



Sveučilište u Zagrebu

Sveučilište u Zagrebu
Geodetski fakultet

Rexhep Nikçi

AUTOMATIZACIJA PROCEDURNOG 3D MODELIRANJA I VIZUALIZACIJE ZGRADA

DOKTORSKI RAD

Zagreb, 2025.



Sveučilište u Zagrebu

Sveučilište u Zagrebu
Geodetski fakultet

Rexhep Nikçi

AUTOMATIZACIJA PROCEDURNOG 3D MODELIRANJA I VIZUALIZACIJE ZGRADA

DOKTORSKI RAD

Mentor:
prof. dr. sc. Robert Župan

Zagreb, 2025.



University of Zagreb

University of Zagreb
Faculty of Geodesy

Rexhep Nikçi

AUTOMATIZATION OF PROCEDURAL 3D MODELING AND VISUALIZATION OF BUILDINGS

DOCTORAL DISSERTATION

Mentor:
prof. dr. sc. Robert Župan

Zagreb, 2025.

SVEUČILIŠTE U ZAGREBU

GEODETSKI FAKULTET



Na temelju članka 19. Etičkog kodeksa Sveučilišta u Zagrebu i Odluke br. 1_349_11 Fakultetskog vijeća Geodetskog fakulteta Sveučilišta u Zagrebu, od 26.10.2017. godine (klasa: 643-03/16-07/03), uređena je obaveza davanja „Izjave o izvornosti“ diplomskog / specijalističkog / doktorskog rada koji se vrednuju na diplomskom, specijalističkom studiju i poslijediplomskom doktorskome studiju geodezije i geoinformatike, a u svrhu potvrđivanja da je rad izvorni rezultat rada studenta te da taj rad ne sadržava druge izvore osim onih koji su u njima navedeni.

IZJAVLJUJEM

Izjavljujem da je moj diplomski / specijalistički / doktorski rad izvorni rezultat mojeg rada te da se u izradi tog rada nisam koristio drugim izvorima osim onih koji su u njemu navedeni.

U Zagrebu, dana _____

Potpis

I. AUTOR	
Ime i prezime:	Rexhep Nikçi
Datum i mjesto rođenja:	04.10.1967., Peć, Republika Kosovo
Sadašnje zaposlenje:	Vermessungsbüro Dipl.-Ing. Rexhep Nikçi
II. DOKTORSKI RAD	
Naslov:	Automatizacija procedurnog modeliranja i vizualizacije 3D zgrada
Broj stranica:	165
Broj slika:	68
Broj tablica:	4
Broj bibliografskih podataka:	152 + 35 URL-a
Ustanova i mjesto gdje je rad izrađen:	Sveučilište u Zagrebu – Geodetski fakultet
Znanstveno područje:	Tehničke znanosti
Znanstveno polje:	Geodezija
Mentor:	prof. dr. sc. Robert Župan
Oznaka i redni broj rada:	
III. OCJENA I OBRANA	
Datum prihvatanja teme Odlukom Senata Sveučilišta u Zagrebu:	21.3.2023.
Datum sjednice Fakultetskog vijeća na kojoj je doktorski rad prihvaćen:	
Sastav povjerenstva koje je ocijenilo doktorski rad:	prof. dr. sc. Đuro Barković
	izv. prof. dr. sc. Mateo Gašparović
	doc. dr. sc. Ivan Medved
Datum obrane doktorskog rada:	
Sastav povjerenstva pred kojim je branjen doktorski rad:	

Zahvala

Sažetak

Istraživanje u disertaciji donosi sveobuhvatan pristup problematici automatizacije procedurnog 3D modeliranja i vizualizacije zgrada, promatrajući ovu temu kroz prizmu geoinformacijskih znanosti, kartografije, arhitekture i informatičke tehnologije. Fokus istraživanja je na razvoju metodologije koja bi omogućila automatizirano generiranje 3D modela zgrada visoke kvalitete, čak i u slučajevima kada su ulazni podaci nepotpuni ili fragmentirani. Postavljeni ciljevi i hipoteze istraživanja usmjereni su na dokazivanje da se procedurnim pristupom, temeljenim na gramatičkim pravilima poput CGA pravila, mogu generirati realistični modeli zgrada te transformirati iz jednog modela u drugi, pritom zadržavajući logičnu strukturu i semantičku dosljednost.

Polazi se od promišljanja da je ključni izazov u 3D modeliranju zgrada pronalaženje načina kako učinkovito reducirati potrebu za ručnim modeliranjem, koje je vremenski zahtjevno i sklono pogreškama, a istovremeno zadržati razinu detalja koja je nužna za primjene poput urbanističkog planiranja, prostorne analize, arhitektonskog dizajna i simulacija okoliša. Stoga se istraživanjem nastoji razviti okvir koji koristi automatizirane metode generiranja modela zgrada putem formalnih i stohastičkih procedurnih gramatika, koje uz odgovarajuće parametre mogu proizvesti brojne varijacije i prilagodbe u kratkom vremenu.

Hipoteze rada potvrđuju da je moguće ostvariti značajan stupanj automatizacije u 3D modeliranju i onda kada nedostaju podaci kao što su oblaci točaka ili LiDAR snimke, posebno korištenjem metoda inverznog procedurnog modeliranja. Takve metode omogućuju rekonstruiranje geometrije objekata i njihovih semantičkih elemenata iz djelomičnih ili neizravnih izvora, poput OpenStreetMap podataka, verbalnih opisa ili fotografija. Pritom se ukazuje na potrebu definiranja granice automatizacije, budući da se pokazalo da semantička nepodudarnost između pojedinih dijelova modela može ozbiljno narušiti kvalitetu krajnjeg rezultata, čak i u prisutnosti formalno korektnih pravila.

Analizom postojećih metoda i alata, autor identificira više smjerova u razvoju procedurnog modeliranja, uključujući primjenu L-sustava, podijeljenih gramatika, parametarskih i hibridnih pristupa. Razrađuju se i algoritmi za spajanje pravila, čime se omogućuje stvaranje prijelaznih modela, koji služe kao most između stilova, struktura i funkcionalnih tipologija zgrada. Poseban doprinos istraživanja ogleda se u razvoju transformacijskih mehanizama koji se mogu koristiti za generiranje novih modela iz postojećih, uz zadržavanje semantičkog integriteta i kontroliranih diskretnih promjena.

U eksperimentalnom dijelu rada prikazano je kako se pomoću CGA pravila mogu generirati različiti tipovi zgrada, od jednostavnih do kompleksnih, uz mogućnost kontrole visine, strukture, broja katova, oblika krova, tipa fasade, pa čak i semantičkih atributa poput namjene prostorija. Nadalje, autor analizira mogućnosti uporabe strojno učenih preferencija korisnika u generativnim modelima, što je omogućeno uvođenjem faktorizacije funkcije preferencija. Time se omogućuje da se gramatička pravila prilagode željama korisnika, čime se unapređuje interaktivnost i personalizacija modela.

Istraživanje također doprinosi primjeni procedurnog modeliranja u geovizualizaciji, posebno kada je cilj poboljšati percepciju urbanih prostora u kontekstu pametnih gradova. Integracijom s IoT sustavima i GIS tehnologijama otvara se mogućnost za stvaranje dinamičkih 3D prikaza koji se mogu prilagođavati u stvarnom vremenu, uzimajući u obzir podatke o prometu, kvaliteti zraka, potrošnji energije i drugim relevantnim varijablama.

U konačnici, rad predstavlja značajan doprinos interdisciplinarnom području automatiziranog 3D modeliranja, nudeći teorijske temelje, metodološke alate i praktične primjene koje mogu unaprijediti kako znanstvene tako i profesionalne prakse u prostornom planiranju, urbanizmu, arhitekturi, informatici i kartografiji. Automatizacija procedurnog modeliranja, kako je prikazana u ovom radu, otvara nove perspektive u razvoju inteligentnih sustava za upravljanje urbanim prostorima, gdje brzina, prilagodljivost i semantička preciznost postaju ključni elementi uspjeha.

Ključne riječi: automatizacija, procedurno modeliranje, 3D vizualizacija, CGA pravila, inverzno modeliranje, geovizualizacija, pametni gradovi, gramatička pravila, generativni dizajn, semantička analiza, oblaci točaka, LiDAR, urbanističko planiranje, GIS, računalna kartografija

Abstract

Dissertation research presents a comprehensive approach to the challenges of automating procedural 3D modeling and visualization of buildings, observing the topic through the lens of geoinformation sciences, cartography, architecture, and information technology. The focus of the research is on developing a methodology that enables automated generation of high-quality 3D building models, even in cases where input data are incomplete or fragmented. The objectives and hypotheses of the research are aimed at proving that a procedural approach, based on grammatical rules such as CGA rules, can generate realistic building models and enable transformations from one model to another, while maintaining logical structure and semantic consistency.

The author starts from the premise that the key challenge in 3D building modeling lies in finding ways to effectively reduce the need for manual modeling, which is time-consuming and prone to errors, while preserving the level of detail necessary for applications such as urban planning, spatial analysis, architectural design, and environmental simulations. Therefore, the research seeks to develop a framework that uses automated methods for generating building models through formal and stochastic procedural grammars, which, with appropriate parameters, can produce numerous variations and adaptations in a short time.

The hypotheses confirm that it is possible to achieve a significant level of automation in 3D modeling even when data such as point clouds or LiDAR scans are missing, especially through the use of inverse procedural modeling methods. These methods enable the reconstruction of object geometry and their semantic elements from partial or indirect sources, such as OpenStreetMap data, verbal descriptions, or photographs. This highlights the need to define the boundaries of automation, as it has been shown that semantic mismatches between model components can seriously compromise the quality of the final result, even in the presence of formally correct rules.

By analyzing existing methods and tools, the author identifies several development directions in procedural modeling, including the use of L-systems, split grammars, parametric and hybrid approaches. Algorithms for rule merging are elaborated, enabling the creation of transitional models that serve as bridges between styles, structures, and functional building typologies. A specific contribution of the research lies in the development of transformation mechanisms that can be used to generate new models from existing ones, while maintaining semantic integrity and controlled discrete changes.

In the experimental part of the study, it is demonstrated how CGA rules can be used to generate various types of buildings, from simple to complex, with control over height, structure, number of floors, roof shape, façade type, and even semantic attributes such as room function. Furthermore, the author analyzes the possibilities of using machine-learned user preferences in generative models, made possible by introducing preference function factorization. This allows grammatical rules to be adapted to user preferences, enhancing interactivity and personalization of the models.

The research also contributes to the application of procedural modeling in geovisualization, especially when the goal is to improve the perception of urban spaces in the context of smart cities. Integration with IoT systems and GIS technologies opens the possibility for creating dynamic 3D visualizations that can adapt in real time, considering data on traffic, air quality, energy consumption, and other relevant variables.

Ultimately, the dissertation makes a significant contribution to the interdisciplinary field of automated 3D modeling, offering theoretical foundations, methodological tools, and practical applications that can enhance both scientific and professional practices in spatial planning, urbanism, architecture, informatics, and cartography. The automation of procedural modeling, as presented in this work, opens new perspectives in the development of intelligent systems for urban space management, where speed, adaptability, and semantic precision become key success factors.

Keywords: automation, procedural modeling, 3D visualization, CGA rules, inverse modeling, geovisualization, smart cities, grammatical rules, generative design, semantic analysis, point clouds, LiDAR, urban planning, GIS, computational cartography.

Sadržaj

1.	Modeliranje i vizualizacija zgrada temeljem procedurnih gramatika	1
1.1.	Uvod.....	1
1.1.1.	Pregled dosadašnjih istraživanja	3
1.1.2.	Svrha istraživanja i motivacija	7
1.1.3.	Struktura doktorskog rada	10
1.1.4.	Tehnički izazovi i pristupi automatizaciji	11
1.2.	Teorijski okvir i pregled literature	13
1.2.1.	Procedurno modeliranje temeljeno na gramatici.....	13
1.2.2.	Lindenmayer sustavi (L-sustavi).....	13
1.2.3.	Klasične gramatike oblika	14
1.2.4.	Gramatike modernog skupa.....	15
1.2.5.	Inverzno procedurno modeliranje temeljeno na gramatici.....	15
1.3.	CGA (Computer Generated Architecture) pravila kroz primjere za unapređenje geovizualizacije 3d modela zgrada	17
1.3.1.	Osnovni koncepti CGA pravila	18
1.3.2.	Struktura CGA pravila za zgrade	19
1.3.3.	Primjeri procedurnog modeliranja zgrada.....	21
1.3.4.	Prednosti i ograničenja CGA pravila	24
1.3.5.	Kategorije inverznih tehnika	25
1.3.6.	Od nule	25
1.3.7.	Prilagodba šablonskih gramatika	26
1.3.8.	Indukcija gramatike.....	27
1.3.9.	Faktorizacija funkcije preferencije.....	27
1.4.	Procedurna gramatika i 3D modeliranje zgrada uz nedostatak podataka.....	30
1.4.1.	Ključni doprinosi dosadašnjih istraživanja.....	30
1.4.2.	Predloženi okvir za 3D modeliranje uz nedostatak podataka	31
1.4.3.	Rasprava i izazovi	31
1.5.	Što je sve potrebno za automatsko procedurno modeliranje zgrada,	33
1.5.1.	Osnove procedurnog modeliranja	33
1.5.2.	Parametarski i pravilski pristupi.....	33
1.5.3.	Procedurno generiranje u video igrama i vizualizacijama	34
1.5.4.	Novi pristup za automatsko generiranje zgrada	34
1.5.5.	Generativni dizajn i održivost	35

2.	Predložena poboljšanja parametrizacije zgrada spajanjem oblika koja su rezultat kombinacija konstrukcijskih oblika koristeći više različitih CGA pravila	37
2.1.	Slična istraživanja	37
2.1.1.	Poboljšanje parametrizacije.....	39
2.1.2.	Predznanje	43
2.2.	Formalne gramatike.....	45
2.3.	Upotreba MLN-a	45
2.3.1.	Strojno učenje težinski pripisane kontekstno-slobodne gramatike za 3D rekonstrukciju fasada.....	49
2.3.2.	Rezultati	53
2.3.3.	Diskusija i prijedlog za nastavak istraživanja	56
2.4.	Skupovi pravila i automatizacija	60
2.4.1.	Transformacija stila.....	69
2.4.2.	Izmjena stilskih pravila	72
2.4.3.	Gotovi primjeri primjene i evaluacija s rezultatima	74
2.5.	Evaluacija sustava za transformaciju grafova	81
3.	Optimizacija 3D modeliranja krovova zgrada korištenjem procedurnih metoda i CGA pravila.....	87
3.1.	Uloga 3D modela u urbanom planiranju i važnost razine detalja (LOD)	87
3.2.	Pregled pristupa modeliranju zgrada i izazovi automatske konstrukcije	89
3.2.1.	Pametni gradovi i potreba za ubrzanim vizualnim modeliranjem.....	90
3.3.	Istraživački okvir: poboljšanje procedurnog modeliranja kroz CGA pravila i urbanu geovizualizaciju.....	91
3.4.	Formalna gramatika definirana četverostrukim skupom.....	93
3.4.1.	Nova metoda oblikovanja jednostavnih i složenih krovnih konstrukcija	94
3.5.	Diskusija.....	97
4.	Procedurni pristup automatskoj izradi 3D modela zgrada u urbanim sredinama	99
4.1.	Uloga 3D modela u kartografiji i prednosti procedurnog pristupa	99
4.2.	Područje istraživanja, podaci i metodologija	100
4.2.1.	Ograničeno područje	101
4.2.2.	Obrada podataka.....	102
4.2.3.	Softverski alati.....	105
4.3.	Primjena u <i>Blenderu</i>	107
4.4.	Izrada i validacija 3D modela zgrada	111
4.5.	Ograničenja modeliranja i analiza podataka	113
4.6.	Diskusija.....	116
5.	Integracija umjetne inteligencije i procedurnog modeliranja u 3D geovizualizaciji zgrada	121

5.1.	Motivacija, hipoteze i istraživački kontekst	121
5.1.1.	Tehnološke mogućnosti i integracijski potencijal suvremenih metoda	122
5.1.2.	Verbalni opisi kao input za 3D modeliranje.....	126
5.1.3.	Vizualna analiza i segmentacija slika zgrada za 3D rekonstrukciju	126
5.1.4.	Interpretacija i izazovi u primjeni	134
5.1.5.	Ključni elementi za izgradnju kvalitetnog tekstualnog inputa u 3D modeliranju	136
5.1.6.	Komparativne analize: točnost, skalabilnost i primjenjivost	138
5.2.	Izazovi, ograničenja i budući smjerovi	140
5.3.	Uloga umjetne inteligencije u budućnosti geovizualizacije i urbanog modeliranja.....	142
6.	Zaključci.....	144
	Pojmovnik	146
	Popis slika i tablica.....	149
	Literatura	153
	Životopis.....	164

1. Modeliranje i vizualizacija zgrada temeljem procedurnih gramatika

Izrada virtualnih modela zgrada od velike je važnosti za istraživačke i poslovne svrhe u geodeziji i kartografiji, ali i u drugim srodnim područjima kao što su npr. arhitektura, arheologija i virtualni prostori videoigara, čiji se zahtjevi mogu kretati između ekspeditivnog stvaranja virtualnih zgrada za opsežno popunjavanje sintetiziranih virtualnih okruženja temeljenih na računalu i testiranja hipoteza putem digitalnih rekonstrukcija. Postoje neki poznati pristupi za postizanje izrade ili rekonstrukcije virtualnih modela, odnosno digitalnih naselja i zgrada.

1.1. Uvod

Ručno modeliranje zahtijeva visokokvalificiranu radnu snagu i znatnu količinu vremena za postizanje željenog digitalnog sadržaja, u procesu koji se sastoji od mnogih faza koje se obično ponavljaju tijekom vremena. Pristup koji se temelji na slici i pristup skeniranju raspona prikladniji su za digitalno očuvanje dobro očuvanih struktura. Međutim, obično zahtijevaju obučene ljudske resurse za pripremu terenskih operacija i rukovanje skupom opremom (npr. 3D skeneri) i naprednim softverskim alatima (npr. fotogrametrijskim aplikacijama). Slično kao i tradicionalni 2D skupovi geopodataka, 3D modeli gradova (koje upotrebljavamo npr. za pametne gradove ili proširenu stvarnost) su aproksimacija stvarnog svijeta, a značajke se modeliraju na određenom stupnju i određenim elementima koji su pojednostavljeni ili izostavljeni. Količina i mješavina sadržaja vođena je uobičajenom uporabom 3D modela grada, porijeklom osnovnih podataka, tehnikom akvizicije, uloženim sredstvima i prostornim odnosima. Ručno stvaranje velikog broja detaljnih modela vrlo je zamorno. Metode procedurnog modeliranja pomažu u smanjenju ručnog napora potrebnog za definiranje modela, dok u isto vrijeme pružaju učinkovit način za opisivanje i pohranu modela. Nakon što se dobije procedurni opis (tj. skup pravila ili gramatika) modela, lako se mogu generirati varijacije modela samo manipuliranjem s nekoliko parametara pravila. Izazov u rekonstrukciji 3D modela gradova je posebice velik broj različitih i složenih oblika građenja. Stoga su u posljednja dva desetljeća predloženi brojni pristupi rekonstrukciji zgrada, ali ova tema i dalje ostaje kao vrlo aktivno područje istraživanja u različitim znanstvenim disciplinama. Osim toga, nedavni razvoj sustava za prikupljanje podataka o ulicama, kao što su mobilni sustavi za kartiranje, otvaraju nove perspektive za poboljšanja u modeliranju zgrada u smislu da se terenski podaci (vrlo gusti i točni) mogu iskoristiti s više performansi (u usporedbi sa podacima dobivenim daljinskim

istraživanjima) za obogaćivanje modela zgrada na razini fasade (npr. geometrija i tekstura). Nakon proučavanja s kojim se sve problemima znanstvenici susreću u različitim pristupima, koji su prikazani u dosadašnjoj literaturi, razvit će se segment učinkovitih tehnika/metoda koje koriste samo najnužnije ili bolje rečeno oskudne podatke a da bi se postigla veća automatizacija cijele izrade od dosadašnje polu-automatske izrade virtualnih okruženja sastavljenih od vanjskih dijelova zgrada isključivo prikazanih vanjskim pročeljima bilo za brzu izradu ili rekonstrukciju postojećih zgrada. Unatoč brojnim postignućima postojećih pristupa procedurnog modeliranja, proizvodnja virtualnih zgrada s interijerima i eksterijerima sastavljenim od nepravokutnih oblika (konveksnih ili konkavnih n-kutova) na razini tlocrta još uvijek se rijetko bavi. Stoga će prva pretpostavka biti da se mogu automatski proizvesti samo zgrade proizvoljno oblikovanih tlocrta.

1.1.1. Pregled dosadašnjih istraživanja

Računalne mogućnosti, kao i dosadašnja istraživanja temeljila su se na raznolikosti gramatika, skupovima pravila i razinama detaljnosti. Nakon što se dobije procedurni opis (tj. skup pravila ili gramatika) modela, lako se mogu generirati varijacije modela samo manipuliranjem s nekoliko parametara pravila. Izazov u rekonstrukciji 3D modela gradova je posebice velik broj različitih i složenih oblika i stilova građenja. Stoga su u posljednja dva desetljeća predloženi brojni pristupi rekonstrukciji zgrada, ali ova tema i dalje ostaje kao vrlo aktivno područje istraživanja u različitim znanstvenim disciplinama. Osim toga, nedavni razvoj sustava za prikupljanje podataka o ulicama, kao što su mobilni sustavi za kartiranje, otvaraju nove perspektive za poboljšanja u modeliranju zgrada u smislu da se terenski podaci (vrlo gusti i točni) mogu iskoristiti s više performansi (u usporedbi sa podacima dobivenim daljinskim istraživanjima) za obogaćivanje modela zgrada na razini fasade (npr. geometrija i tekstura).

Mnogi najsuvremeniji sustavi procedurnog modeliranja proširili su se izvan upotrebe formalnih gramatika, ali se još uvijek oslanjaju na pravila modeliranja. Danas se više ne temelje na gramatici, ali ih znanstvenici koji se bave računalnom grafikom još uvijek tako nazivaju. Njihove prednosti nadmašuju činjenicu da više ne ulaze u formalnu definiciju. Aristid Lindenmayer uveo je 1968 L-sustave (Lindenmayer sustave), koji su tehnika prepisivanja nizova koje je svojevremeno proučavao. Za razliku od formalnih gramatika koje primjenjuju sekvencijalna pravila izrade modela, L-sustavi koriste strategiju paralelne zamjene koja istovremeno primjenjuje pravila modeliranja na sve ne-terminale u nizu. Ovom metodologijom se također mogu izrađivati fraktali. Iako tehnike utemeljene na pravilima i procedurne tehnike općenito imaju mnoge prednosti, postoje i određeni nedostaci. Za pisanje procedura ili pravila za opisivanje sadržaja potrebno je određeno znanje o programiranju CGA pravila oblikovanja. Kartografi ponekad nisu upoznati s programiranjem, dok mnogim programerima nedostaje umjetnički, matematički i kartografski smisao. Nadalje, teško je steći razumijevanje procedurnog modela kada se daju samo pravila, a ne njihov rezultat. Posebno za procedurne modele s takozvanim stohastičkim pravilima koja nude nekoliko različitih rješenja za isti zadatak, može biti teško vizualizirati i kontrolirati prostor mogućih dizajna koji obuhvaćaju. Ti izazovi su općenito komplicirani i ne samo za ljude bez znanja programiranja. Do sada se problem u literaturi i radovima pojavljuje u slučajevima kad dvije jednostavne gramatike modela imaju korespondencije (odnosno semantička podudaranja) za pravila izrade modela koja automatski generiraju masovne 3D modele. Rasponi transformacija diskretnih pravila su za samo vrlo ograničen prostor s četiri oblika, tzv. krupno-zrnate nasuprot fino-zrnatim

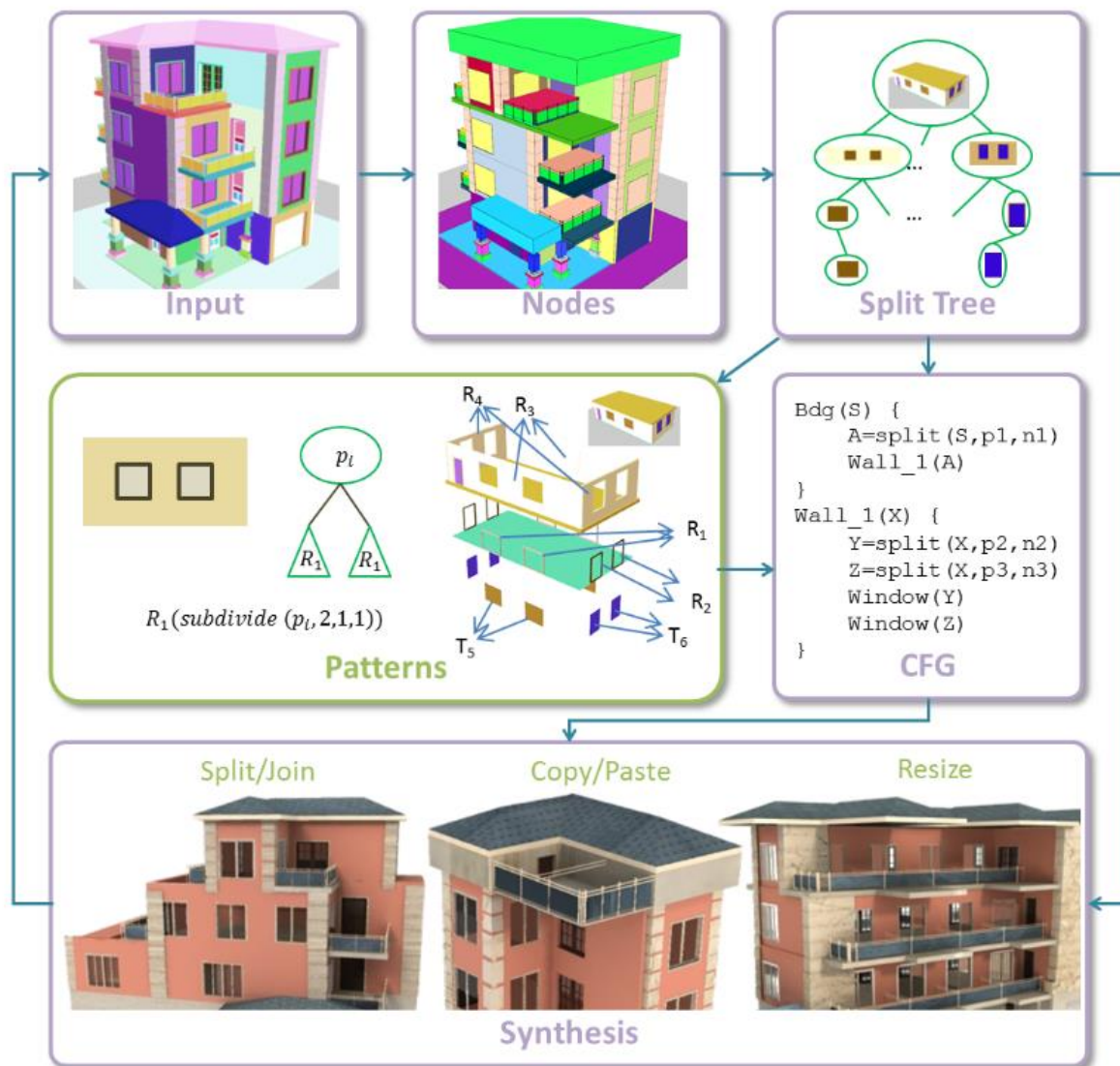
transformacijama. Može se pretpostaviti i u istraživanju još ispitati hoće li spajanje CGA pravila modeliranja za fino-zrnate transformacije obuhvatiti beskonačno puno varijacija oblika.

Dosadašnja istraživanja mogu se grupirati u nekoliko osnovnih pravaca koji se ponavljaju otprilike desetak godina unazad.

U nastavku ću nastojati objasniti pojmove procedurno modeliranje i inverzno procedurno modeliranje. Procedurno modeliranje je jednostavno rečeno stvaranje grafičkog modela (npr. zgrade) pomoću računala i procedurnih pravila (npr. CGA pravila). Cilj je strojnog jezika, razumljivog računalu koji ga može relativno brzo obraditi, dobiti 3D grafički prikaz (koji predstavlja aproksimaciju) što vjerniji stvarnom objektu u prirodi.

Procedurno modeliranje je metodologija koja se koristi za modeliranje najčešće urbanog okoliša (Prusinkiewicz & Lindenmayer, 1990), gradova (Parish & Müller, 2001) i drugih objekata. (Vanegas et al., 2009) i (Smelik et al., 2014a) predstavljaju sveobuhvatan pregled procedurnog modeliranja. Povijest procedurnog modeliranja razvijala se u početku u urbanom dizajnu kroz gramatike oblika (Stiny, 1975), zatim kroz uzorke i poredak, te u novije vrijeme vizualizaciju zgrada procedurnim metodama (Wonka et al., 2003), (Müller, Wonka, Haegler, Ulmer, & Gool, 2006). Dijagram izrade procedurno modelirane zgrade (Slika 1).

Međutim, stvaranje procedurnih pravila zahtijeva poznavanje jezika modeliranja i općeg procesa kodifikacije stila (Schwarz & Müller, 2015a). Dok (Lipp et al., 2008) unutar CityEngine-a opisuju interaktivne sustave za uređivanje procedurnih modela zgrada, pretpostavljaju da su procedurni modeli već osigurani. Za automatizaciju bi bilo najbolje izbjegavati odnosno automatizirati ponavljajuće postupke ručnog pisanja skupa parametriziranih procedurnih pravila pri definiranju određenog stila građevine na temelju ulaznog modela. Za inverzno procedurno modeliranje trebaju nam parametrizirana gramatička pravila i/ili vrijednosti parametara koji daju traženi model. (Vanegas et al., 2010) predlažu inverzni procedurni pristup za rekonstrukciju zgrada na Manhattanu, ali ne daju gramatiku niti prikazuju uređivanje. (Bokeloh et al., 2010a) iskorištavaju ponavljanje djelomično simetričnih struktura i omogućuju sintezu modela zgrade.



Slika 1. Jedan od pristupa kada se uzima unešeni 3D model, izvlači komponente, pronalazi uzorke, konstruira grane stabla, gradi gramatiku bez konteksta i omogućava interaktivnu sintezu operacija (URL 1).

Unatoč tome, njihova osnova simetrije su površine, stoga bilo koji 3D unos segmenata nije primjenjiv kao ulazni parametar. Autori kao (J. Talton et al., 2012a) pronalaze vjerojatnost gramatike koja obuhvaća obrasce grupa segmentiranih i označenih hijerarhijskih oblika kao ulazni parametar. (J. O. Talton et al., 2011) i (Vanegas et al., 2012) koriste optimizaciju Monte Carlo Markovljevog lanca kako bi otkrili kako promijeniti parametre danog procedurnog modela zgrada da bi se dobio rezultat koji zadovoljava željeni skup svojstava, dok Demir uvodi nekoliko tehnika procedurizacije, primjenjujući inverzno procedurno modeliranje na oblake točaka (Demir et al., 2015b) za poboljšanje kvalitete oblaka točaka i mogućnosti procedurnog uređivanja na modele gradova s vanjskom teksturom (Demir et al., 2014) za procedurizaciju

postojećih gradova za sintezu. Također i (Nishida et al., 2016a) primijenili su inverzno procedurno modeliranje za skiciranje ili planiranje, kako bi si olakšali stvaranje novog 3D sadržaja. Suprotno tome, moj automatski pristup proizvodi podijeljenu gramatiku, podržava proizvoljne 3D arhitektonske modele (umjesto samo fasada), stvara hijerarhijska pravila, ne treba označeni unos i podržava razne alate za uređivanje. Štoviše, mnogi od spomenutih dosadašnjih pristupa pretpostavljaju da je procedurni model osiguran i da je potrebno otkriti samo vrijednosti parametara. Nasuprot tome, predstavljam metodu koja ne pretpostavlja nikakvo znanje o procedurnom modelu i generira i gramatiku i vrijednosti parametara. Uređivanje oblika pokriva višestruke metodologije uređivanja s obzirom na strukturu koje su također predložene za poligonalne modele (Mitra et al., 2013). Razlikuju se uglavnom po razini automatizacije, vrstama obrađenih geometrija, podršci za hijerarhijske obrasce, izradi eksplicitne gramatike te fleksibilnosti i kontroli uređivanja. Na primjer, (Pauly et al., 2008) pronalaze uzorke s translatorskim, rotacijskim i cilindričnim mrežnim rasporedom. Međutim, njihov cilj nije generiranje gramatike, niti je omogućena kontrola uređivanja a i hijerarhijski obrasce nisu podržani. Neki prethodni radovi (Bao et al., 2013), (Kalogerakis et al., 2012), (Lin et al., 2011), (Lin et al., 2011) podržavaju hijerarhijske obrasce, ali zahtijevaju pomoć korisnika, dok kod (Lin et al., 2011) podržavaju samo ID uzorke a (Kalogerakis et al., 2012) nude ograničenu kontrolu uređivanja. Pristupi (Bokeloh et al., 2011) i (Bokeloh et al., 2012) pružaju fleksibilnu kontrolu uređivanja oblika, ali nije osigurana podrška za hijerarhijski uzorak. Nova ideja i interaktivni pristup potpuno je automatski, podržava veću grupu uzoraka, a daje se i hijerarhijska organizacija pravila. Nova metoda tu prikazana generira eksplicitnu podijeljenu gramatiku bez konteksta i omogućuje i lokalnu i globalnu sintezu svjesnu strukture. Konfliktne ciljeve uređivanja rješavaju se brzo korištenjem rijetke linearne optimizacije najmanjih kvadrata umjesto zadovoljavanja željenih uređivanja u prethodno određenom redoslijedu (Bao et al., 2013; Kalogerakis et al., 2010; Lin et al., 2011) važnosti kao kod (Zheng et al., 2011) ili rješavanjem matrice kao u (Bokeloh et al., 2012). Odgovarajuća segmentacija može se izvesti pomoću algoritma arhitektonske segmentacije kao što su (Attene et al., 2006; Demir et al., 2015b; Kalogerakis et al., 2010) usmjerili istraživanje na zgrade, ali prikazuju i rezultate koristeći i druge algoritme segmentacije. Inverzno procedurno modeliranje otkriva procedurni prikaz postojećeg geometrijskog modela, a otkriveni procedurni model zatim podržava sintetiziranje novih sličnih modela. Do sada je inverzno procedurno modeliranje struktura zgrada bilo usmjereno na fasade, oblake točaka, zgrade ili cijele gradove. Fasadne metode (F. Wu et al., 2014a; H. Zhang et al., 2013) daju uvjerljive rezultate, ali se ne mogu lako proširiti na geovizualizaciju 3D zgrada. Pristupi modeliranja pomoću oblaka točaka, npr. (Armeni et al.,

2016; Berner et al., 2008; Demir et al., 2015a; Toshev et al., 2010) usredotočeni su na segmentaciju i analizu simetrije za pronalaženje ponavljanja oblika zgrada. U metodama koje se temelje na modelima obično se pretpostavlja postojanje segmentiranog modela sa strukturnim ograničenjima (Lin et al., 2011), višestruki primjerci istog stila gradnje u različitim konfiguracijama (J. Talton et al., 2012a), a segmenti izrezani samo duž krivulja unutar simetričnih područja koja ograničavaju mogućnost obrade proizvoljnih geometrija zgrada (Bokeloh et al., 2010a) i/ili ograničenih geometrijskih oblika (Vanegas et al., 2010).

Rješavanje problema procedurnog modeliranja u literaturi može se razdvojiti na dvije osnovne komponente ili osnovna pitanja. Na koji način zadovoljiti korisnike koji očekuju sve veću kvalitetu digitalnog sadržaja, odnosno računalne i vizualizacijske mogućnosti zbog bolje percepcije kod korisnika, a drugo pitanje odnosi se na dostupne alate za učinkovito (sa što više automatizacije) modeliranje zgrada i u konačnici cijelih gradova.

Trenutni pristupi u istraživanjima se uglavnom bave različitim formulacijama za dijeljenje geometrije na komponente (Slika 3 i Slika 4), a zatim se izdvoje obrasci i pravila (Slika 2). Istraživanja se trebaju usmjeravati ne samo na traženje i izdvajanje uzoraka po sličnosti već nam treba i usmjeravanje i kontrola generiranja modela prema optimalnim parametrima i pravilima procedurnog modeliranja za određeni objekt.

1.1.2. Svrha istraživanja i motivacija

Svrha ovog istraživanja je automatizacija procedurnog 3D modeliranja i vizualizacije zgrada, čime se omogućuje stvaranje točnih, detaljnih i prilagodljivih modela gradova i zgrada uz minimalne ručne intervencije. Ključni cilj je razviti sustave koji koriste skup pravila za brzo generiranje i prilagodbu različitih građevinskih elemenata u urbanim područjima, čime se smanjuje potreba za detaljnim ručnim modeliranjem. Takvi sustavi pomažu u optimizaciji resursa, vremenskih rokova i troškova, omogućujući pouzdaniji i brži proces stvaranja i upravljanja urbanim podacima. Motivacija za ovo istraživanje proizlazi iz sve veće potrebe za brzim generiranjem kvalitetnih 3D modela gradova u kontekstu pametnih gradova i geoinformacijskih sustava (GIS). S obzirom na važnost 3D modela u donošenju urbanističkih odluka, geovizualizaciji, simulacijama okoliša i arhitektonskom dizajnu, automatsko modeliranje predstavlja temeljnu komponentu u unapređenju održivosti i učinkovitosti urbanih sustava. Daljnji razvoj ovog područja olakšat će precizno prostorno planiranje, te omogućiti široku primjenu geovizualizacije za pametne gradove, virtualnu stvarnost i simulacije okoliša.

$F = \{\text{terminalni znakovi, neterminalni znakovi, pravila, aksiom}\}$

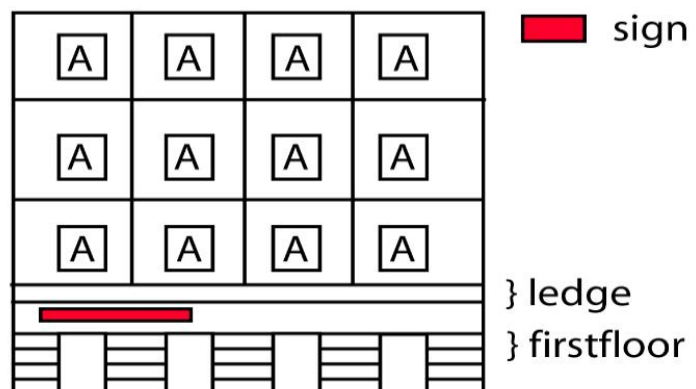


$F = \{\text{terminalni znakovi, neterminalni znakovi, pravila, aksiom}\}$

Slika 2. Izdvajanje pravila i/ili parametara za stvaranje modela iz zadanih podataka zgrada-fotografija, podataka iz oblaka točaka i sl.) (URL 2).



Slika 3. Razdvajanje na komponente (lijevo) i njihovo razdvajanje na uzorke i pravila (desno).



Slika 4. Povratno se može podijeliti ravninu fasade na dijelove (URL 3).

L-sustavi (npr. za vizualizaciju biljaka), dijeljene gramatike (npr. za fasade, zgrade) i gramatike oblika (npr. za modeliranje zgrada, skulpture, prikaz terena, CGA (CityEngine), Slika 5.



Slika 5. Naći gramatiku koja daje željeni 3D model (URL 3).

Automatizacija procedurnog modeliranja nosi veliki potencijal za poboljšanje brzine i točnosti u stvaranju modela urbanih područja, što ima neposredan utjecaj na učinkovitost brojnih aktivnosti. Primjerice, u kontekstu urbanističkog planiranja, 3D modeli omogućuju simulacije koje prikazuju kako zgrade utječu na mikroklimatske uvjete, poput sjena koje bacaju tijekom dana ili širenja buke. Time se osigurava bolje prostorno planiranje i smanjuje potreba za fizičkim izmjenama u prostoru nakon izgradnje. Modeli također olakšavaju procjenu isplativosti projekata poput solarnih panela, jer omogućuju točne procjene izloženosti sunčevom zračenju na zgradama.

Dodatna motivacija proizlazi iz izazova vezanih uz ručne metode modeliranja. Ručno modeliranje zahtijeva značajnu količinu resursa, stručnog znanja i vremena. Procedurno modeliranje, s druge strane, koristi pravila i algoritme koji omogućuju automatsko stvaranje modela cijelih zgrada ili čak dijelova gradova uz minimalnu ručnu intervenciju. To ne samo da smanjuje vrijeme izrade, već omogućuje i brzu prilagodbu modela, što je posebno važno u situacijama gdje je potrebno ažuriranje podataka zbog promjena u urbanom prostoru.

Cilj istraživanja je doprinijeti razvoju sustava koji će unaprijediti postojeće metode procedurnog modeliranja te omogućiti veći stupanj automatizacije u procesima urbanog modeliranja. Očekuje se da će istraživanje rezultirati metodama koje mogu generirati modele zgrada čak i u nedostatku detaljnih podataka, koristeći raspoložive podatke kao što su oblaci točaka ili podaci dobiveni LIDAR tehnologijom. Ova prilagodljivost omogućava lakšu primjenu u različitim okruženjima i urbanim kontekstima. Daljnja motivacija leži u potrebi za standardizacijom podataka i povećanju razine detalja u prikazu (LOD – Level of Detail), što osigurava konzistentnost podataka u 3D GIS sustavima. Razine detalja ključne su u prilagodbi modela specifičnim potrebama, gdje detaljniji prikazi omogućuju složenije analize i simulacije, dok su jednostavniji modeli prikladni za brže vizualizacije i prezentacije na razini cijelog grada.

Procedurno modeliranje s mogućnošću prilagodbe LOD-a također nudi priliku za unaprjeđenje 3D modeliranja u pogledu ekonomičnosti, jer se resursi prilagođavaju potrebama, smanjujući nepotrebne detalje kada nisu potrebni. Predložene metode automatiziranog procedurnog modeliranja mogle bi imati široku primjenu u urbanističkom planiranju, arhitektonskom dizajnu, geoinformatici, te u razvoju pametnih gradova. Primjerice, simulacije u stvarnom vremenu, poput protoka prometa, zagađenja zraka, ili širenja svjetlosti, mogu se brže i preciznije provesti koristeći automatski generirane 3D modele, što je od velike koristi za planiranje i upravljanje gradovima. U konačnici, istraživanje ima za cilj unaprijediti učinkovitost i prilagodljivost procedurnog modeliranja u različitim prostornim aplikacijama, potičući daljnji razvoj i inovacije u području automatizacije geoinformacijskih sustava i 3D vizualizacije zgrada.

1.1.3. Struktura doktorskog rada

Doktorski rad strukturiran je u četiri glavna poglavlja i niz potpoglavlja međusobno povezanih kroz zajednički istraživački okvir. U nastavku se daje koncizan pregled sadržaja pojedinoga poglavlja

Poglavlje 1 sadrži uvod u doktorski rad i prikazuje kontekst istraživanja, motivaciju i znanstvenu opravdanost rada. Obradeni su ključni pojmovi vezani uz procedurno modeliranje i vizualizaciju 3D zgrada te teorijska osnova gramatika u računalnoj grafici. U ovom poglavlju definirani su glavni ciljevi i hipoteze istraživanja, detaljno je opisana korištena metodologija i istraživački plan te su izneseni očekivani znanstveni doprinosi. Dodatno, razrađen je pregled dosadašnjih istraživanja, uključujući parametarske, stohastičke i formalne pristupe modeliranju zgrada.

Poglavlje 2 usmjereno je na prijedloge poboljšanja u automatizaciji procedurnog modeliranja krovova. Prikazani su tehnički izazovi pri parametrizaciji kompleksnih krovnih struktura, a zatim se predlaže nova metoda oblikovanja jednostavnih i složenih krovova temeljem razvijenih CGA pravila. Poglavlje uključuje pregled prethodnih radova, motivaciju za rješavanje ovog problema i eksperimentalne rezultate koji potvrđuju učinkovitost predloženog pristupa.

Poglavlje 3 prikazuje automatsko generiranje 3D modela zgrada korištenjem kombinacije LiDAR podataka i OpenStreetMapa uz primjenu procedurnih modela. Detaljno su opisani korišteni materijali i metode, uključujući obradu prostornih podataka i upotrebu softverskih

alata poput Blender-a i CityEngine-a. U poglavlju se također provodi komparativna analiza učinkovitosti različitih sustava za procedurno modeliranje te se predstavljaju rezultati i evaluacija kvalitete generiranih modela.

Poglavlje 4 bavi se primjenom procedurnog modeliranja u kontekstu geovizualizacije, s naglaskom na usporedbu s metodama temeljenima na umjetnoj inteligenciji. Analiziraju se mogućnosti koje pružaju verbalni opisi, optičko prepoznavanje slika i njihova integracija u 3D modeliranje. Poglavlje uključuje komparativnu analizu rezultata, prikaz ograničenja i izazova, kao i smjernice za buduća istraživanja i razvoj sustava vizualizacije urbanih prostora u pametnim gradovima.

1.1.4. Tehnički izazovi i pristupi automatizaciji

Jedan od glavnih izazova u automatizaciji procedurnog modeliranja leži u razvoju algoritama koji mogu prepoznati, interpretirati i generirati složene arhitektonske oblike. Trenutni procedurni sustavi često koriste pravila bazirana na gramatici (primjerice, CGA pravila u ArcGIS CityEngine-u), koja omogućuju kreiranje zgrada i drugih građevinskih struktura primjenom određenih geometrijskih pravila. Međutim, kako bi se omogućila visoka razina automatizacije i fleksibilnosti, potrebno je unaprijediti sposobnost sustava da analizira i prilagođava modele prema stvarnim podacima, poput oblaka točaka ili fotogrametrijskih snimaka. Razvijanjem metoda koje kombiniraju podatke iz različitih izvora s procedurnim pravilima, omogućava se generiranje modela koji bolje odražavaju specifične značajke pojedinog prostora, što je od ključne važnosti u kontekstu urbanih analiza i prostornog planiranja.

Procedurno modeliranje zahtijeva ne samo generiranje geometrije već i semantičko prepoznavanje – sposobnost sustava da prepozna i razlikuje ključne elemente kao što su fasade, krovovi, prozori i drugi građevinski detalji zgrade. Semantičko prepoznavanje omogućava bolju integraciju modela u pametne gradove, gdje su geoinformacijski sustavi obogaćeni informacijama o strukturi i funkciji objekata. Primjenom metoda strojnog učenja ili dubokih neuronskih mreža, modelima se mogu dodati semantičke oznake koje omogućuju inteligentnije i interaktivnije modele. Na primjer, sustavi bi mogli automatski prepoznati promjene u urbanom okruženju i ažurirati 3D modele u stvarnom vremenu, što doprinosi točnosti i ažurnosti podataka u GIS sustavima.

Jedan od ključnih potencijala automatiziranog procedurnog modeliranja leži u njegovoj povezanosti s Internetom stvari (IoT) i pametnim gradovima. Korištenjem podataka prikupljenih IoT senzorima, modeli se mogu automatski ažurirati u skladu s promjenama u stvarnom vremenu. Na primjer, podaci o prometu, kvaliteti zraka ili potrošnji energije mogu se integrirati u 3D modele, pružajući tako dinamične prikaze koji mogu pomoći u donošenju odluka. Na taj način, procedurno modeliranje postaje važan alat za integraciju i vizualizaciju podataka u kontekstu pametnih gradova, omogućujući bolju analizu, simulaciju i optimizaciju urbanih procesa.

Očekuje se da će rezultati ovog istraživanja doprinijeti razvoju novih metoda za visoko automatizirano modeliranje 3D prostora, što će imati izravnu primjenu u geoinformacijskoj znanosti. Napredak u algoritmima za automatsko prepoznavanje i generiranje geometrijskih i semantičkih podataka ubrzat će procese kartografske vizualizacije i analize prostora, čineći ih preciznijim i učinkovitijim. Na ovaj način, istraživanje ima potencijal transformirati način na koji se koriste 3D modeli u prostornom planiranju, zaštiti okoliša, te mnogim drugim primjenama u kojima su prostorni podaci ključni.

Istraživanje će svakako predstavljati korak naprijed u razvoju metodologija za automatizaciju procedurnog 3D modeliranja i vizualizacije, omogućujući precizniju, bržu i interaktivniju analizu prostora. S obzirom na sve veću potrebu za kvalitetnim 3D modelima u urbanom planiranju i upravljanju gradovima, očekuje se da će ovi alati pomoći u prevladavanju izazova s kojima se suočavaju gradovi u eri digitalne transformacije. Budući smjerovi istraživanja mogli bi uključivati integraciju s umjetnom inteligencijom i IoT tehnologijama, omogućujući stvaranje pametnih, dinamičnih modela gradova koji se mogu prilagoditi i reagirati na promjene u stvarnom vremenu. Ti modeli mogli bi biti posebno korisni u kontekstu pametnih gradova, gdje bi se podaci s terena mogli koristiti za predviđanje i planiranje scenarija kao što su ekološki učinci, prometna opterećenja ili energetska efikasnost. Na ovaj način, procedurno modeliranje postaje ključna tehnologija koja može pomoći urbanistima, planerima i znanstvenicima da razumiju i upravljaju složenim urbanim sustavima na učinkovitiji i održiviji način.

1.2. Teorijski okvir i pregled literature

U ovom poglavlju donosimo kratki pregled povijesti procedurnog modeliranja temeljenog na gramatici, jer će se u daljnjim istraživanjima upravo to upotrijebiti, organizirajući sadržaj kroz ključne metode i razvojne pravce. Poglavlje je strukturirano u nekoliko potpoglavlja, od uvodnih koncepata formalnih gramatika, preko Lindenmayerovih sustava i klasičnih gramatika oblika, pa sve do modernijih pristupa poput inverznog procedurnog modeliranja.

1.2.1. Procedurno modeliranje temeljeno na gramatici

Procedurno modeliranje temeljeno na gramatici odnosi se na metodu generiranja 3D modela pomoću formalnih pravila i sintaktičkih struktura, pri čemu se oblik i struktura objekta opisuju nizom produkcijskih pravila sličnih onima u računalnoj lingvistici ili teoriji jezika. Umjesto ručnog modeliranja, korisnik definira skup gramatičkih pravila kojima se postupno generira složena geometrija – poput zgrada, fasada ili urbanih blokova – iz osnovnih oblika i volumena. Takav pristup omogućuje visoku razinu automatizacije, prilagodljivost i ponovljivost, što ga čini iznimno pogodnim za modeliranje velikih urbanih prostora. Gramatike, poput CGA (Computer Generated Architecture) pravila, omogućuju da se isti niz pravila primijeni na različite ulazne podatke te da se manipuliranjem parametrima brzo dobiju varijacije istog tipa objekta. U kontekstu urbanog modeliranja, proceduralne gramatike omogućuju generiranje zgrada i struktura koje poštuju zadane stilove, proporcije i semantičke uvjete, čime se značajno povećava učinkovitost i fleksibilnost u odnosu na tradicionalne metode 3D modeliranja.

1.2.2. Lindenmayer sustavi (L-sustavi)

Lindenmayer sustavi (L-sustavi) razvijeni su 1968. za modeliranje rasta biljaka. Ovi sustavi uvode metodu paralelne zamjene, koja omogućuje istodobnu primjenu pravila na sve neterminale u nizu, simulirajući biološki rast. Uz pomoć grafičkih prikaza, Przemysław Prusinkiewicz popularizirao je L-sustave u računalnoj grafici. Upotrebom tzv. grafike kornjača, gdje se koriste simboli za pomicanje i rotaciju, omogućio je stvaranje raznih razgranatih struktura. L-sustavi su također prošireni stohastičkim pravilima kako bi omogućili varijacije u modelima, dodajući mogućnost nasumičnog generiranja grana, što dodatno doprinosi realističnosti modela biljaka i prirodnih struktura.

Lindenmayer sustavi (L-sustavi), koje je uveo biolog Aristid Lindenmayer, izvorno su razvijeni za modeliranje biološkog rasta biljaka. Ključna značajka L-sustava je paralelna zamjena

pravila, gdje se pravila istovremeno primjenjuju na sve neterminalne simbole u nizu. Ova tehnika odražava stvarni biološki proces rasta, u kojem se stanice istodobno dijele i razvijaju. Prusinkiewicz je popularizirao L-sustave u računalnoj grafici uvođenjem tzv. grafike kornjača, pristupa gdje "kornjača" linearno prolazi kroz niz simbola, izvršavajući upute kao što su pomicanje naprijed (crtanje linije) ili rotacija (mijenjanje smjera). Ova tehnika omogućila je modeliranje razgranatih struktura poput stabala, čime su L-sustavi postali ključan alat u procedurnom modeliranju prirodnih oblika.

L-sustavi su se proširili uvođenjem stohastičkih pravila koja omogućuju varijacije unutar modela. Na primjer, pravila mogu biti dodijeljena s određenim vjerojatnostima, stvarajući prirodnu varijabilnost u obliku grana ili listova. Također su razvijeni parametarski L-sustavi, koji koriste parametre za prilagodbu oblika, omogućujući kontrolu nad detaljima poput duljine ili kutova grana. Ovi sustavi primijenjeni su i u urbanom planiranju, primjerice za generiranje mreža cesta koje reagiraju na topografiju i okolišne uvjete, pokazujući kako je ovaj pristup postao univerzalan alat u procedurnom generiranju složenih, organskih struktura.

1.2.3. Klasične gramatike oblika

Klasične gramatike oblika, donijele su novi pristup modeliranju koristeći grafičke oblike kao osnovne elemente, za razliku od simboličkih nizova. Pravila u ovim gramatikama nisu tekstualna, već djeluju izravno na grafičke primitivne oblike kao što su linije, pravokutnici i krugovi. Na primjer, Paladijska gramatika omogućuje kreiranje arhitektonskih tlocrta pomoću pravila koja preslikavaju jedan grafički oblik u drugi. Ovaj vizualni i prostorni pristup bio je revolucionaran jer je omogućio modeliranje struktura koje nisu samo apstraktne već i vizualno prepoznatljive.

Međutim, klasične gramatike oblika imaju ograničenja zbog složenosti prepoznavanja podudaranja oblika, koja je računalno zahtjevnija i često zahtijeva korisničku interakciju. Problem se javlja jer se oblici ne tretiraju kao fiksni simboli, već se mogu skalirati, pozicionirati ili transformirati na mnogo načina. Zbog toga pravila proizvodnje u klasičnim gramatičkim sustavima često uključuju kreativni doprinos korisnika koji određuje položaj i proporcije oblika. Ovaj problem podudaranja otežava algoritamsku primjenu ovih gramatika, ali im daje fleksibilnost u dizajnu arhitektonskih i umjetničkih struktura

1.2.4. Gramatike modernog skupa

Gramatike modernog skupa evolucija su klasičnih gramatika oblika, s fokusom na korištenje označenih oblika kao osnovnih jedinica koje se ne razdvajaju dalje. Ova metoda omogućava jednostavnije definiranje pravila zamjene i koristi se za modeliranje složenih 3D struktura. Ključna inovacija unutar ovog pristupa bila je razvoj podijeljenih gramatika, posebno kroz CGA oblik (Computationally Generated Architecture). CGA oblik koristi podjelu oblika duž osi za generiranje arhitektonskih struktura poput fasada i katova, što omogućuje stvaranje zgrada s visokim stupnjem detalja i raznolikosti. Ovaj pristup našao je široku primjenu u industriji, primjerice u softveru CityEngine, gdje se koristi za generiranje gradskih scena i virtualnih gradova.

Kako bi se prevladala ograničenja klasičnih gramatika oblika, razvijene su gramatike modernog skupa koje koriste atome (elemente koji se ne mogu dijeliti) kao osnovne jedinice. Ovaj pristup smanjuje složenost pravila jer pravila djeluju na označene oblike koji se ne razlažu dalje. To omogućuje modeliranje složenih 3D struktura s manje računalnih zahtjeva. Ključna inovacija unutar gramatika modernog skupa bila je razvoj podijeljenih gramatika, koje koriste pravila za dijeljenje oblika duž osi, olakšavajući stvaranje arhitektonskih elemenata kao što su fasade, katovi i prozori.

Najpoznatiji primjer primjene podijeljenih gramatika je CGA oblik (Computationally Generated Architecture), koji je razvio Pascal Müller i njegov tim. CGA omogućuje automatsko generiranje arhitektonskih oblika tako što koristi pravila za podjelu osnovnog oblika na manje dijelove koji se usklađuju s prostornim zahtjevima. CGA se koristi u softveru kao što je CityEngine, koji je namijenjen generiranju gradskih scena. CGA oblik omogućuje preciznu kontrolu nad veličinom i položajem arhitektonskih elemenata, a pravila se mogu parametrizirati kako bi se generirale razne verzije zgrada unutar jednog stila. Dodatne proširene verzije CGA oblika, poput CGA++, uključuju napredne alate za sinkronizaciju izvođenja pravila i bolju kontrolu nad detaljima modela, čime se CGA pozicionira kao standard u procedurnom 3D modeliranju kuća i zgrada.

1.2.5. Inverzno procedurno modeliranje temeljeno na gramatici

Kod inverznog procedurnog modeliranja pokušava se generirati gramatiku iz postojećih 3D modela ili slika uključuje analizu simetrija i ponavljanja unutar modela kako bi se prepoznali dijelovi koje je moguće opisati rekurzivnim pravilima. Inverzno modeliranje zahtijeva složene

algoritme, kao što su metode strojnog učenja, kako bi se omogućilo prepoznavanje obrazaca i generiranje pravila. Na primjer, inverzno modeliranje fasada omogućava generiranje pravila za zgrade s ponavljajućim elementima poput prozora i katova. Iako se ovaj pristup pokazao korisnim, njegova složenost i računalni zahtjevi ograničavaju primjenu na jednostavnije strukture ili specifične domene s pravilnom geometrijom.

Opisujem evoluciju procedurnog modeliranja kroz različite gramatičke pristupe, ističući kako su formalne gramatike postavile temelje za izgradnju složenih i prilagodljivih modela u računalnoj grafici. Svaka metoda, od klasičnih do modernih gramatika, donosi svoje specifične prednosti i ograničenja. Zajedno, ove metode omogućuju stvaranje bogatih i detaljnih 3D modela, čime procedurno modeliranje postaje ključni alat u područjima kao što su urbani dizajn, animacija i simulacija.

Inverzno procedurno modeliranje predstavlja pristup gdje se procedurna pravila generiraju analizom postojećih modela. Ovaj proces pokušava pronaći gramatička pravila koja mogu opisati dati 3D model ili sliku. U inverznom procedurnom modeliranju, izazov leži u činjenici da postoje mnogi mogući načini za kodiranje jednog modela u pravila, i često je nejasno koji je najbolji ili najefikasniji način.

U ovom pristupu koriste se različite tehnike, uključujući prepoznavanje simetrija i sličnosti. Na primjer, fasade zgrada s pravilnim rasporedom prozora i katova omogućuju lakšu identifikaciju pravila koja ih opisuju. Proces obično koristi metode strojnog učenja kako bi prepoznao obrasce i identificirao dijelove koji su prikladni za opis rekurzivnim pravilima. Također, inverzno modeliranje može se koristiti za kompresiju podataka, gdje se složeni modeli mogu pohraniti kao niz pravila umjesto u obliku eksplicitne geometrije, čime se štedi prostor i resursi.

Jedan od čestih pristupa uključuje prilagodbu generičke, parametarske gramatike specifičnim podacima, poput parametarskih oblika ili skica koje zatim predstavljaju pojedine elemente zgrada, prozora ili masa. U slučajevima složenih botaničkih struktura koristi se kombinacija inverznog modeliranja i optimizacije kako bi se dobila gramatika koja opisuje različite, ali međusobno slične strukture. Unatoč potencijalu, inverzno procedurno modeliranje suočeno je s ograničenjima zbog složenosti izvođenja, što ga čini pogodnim uglavnom za specifične vrste ulaznih podataka ili jednostavnije oblike.

Pokazujem kako su se procedure gramatike razvijale od formalnih sustava u lingvistici do ključnih alata u računalnoj grafici i arhitektonskom modeliranju. Svaka vrsta gramatike predstavlja razvojni korak s posebnim doprinosima i ograničenjima. Lindenmayer sustavi

omogućili su modeliranje prirodnih oblika poput biljaka, klasične gramatike oblika uvele su geometrijske transformacije za arhitektonske nacрте, dok su gramatike modernog skupa omogućile stvaranje detaljnih 3D arhitektonskih modela pomoću pravila podjele.

Te metode ne samo da omogućuju stvaranje složenih i estetski privlačnih 3D modela, već su i temelj za razvoj novih tehnika procedurnog modeliranja. Inverzno procedurno modeliranje otvara mogućnosti kompresije podataka i generiranja pravila iz stvarnog svijeta, ali i predstavlja izazov zbog svoje složenosti. Cjelokupni razvoj procedurnog modeliranja temeljenog na gramatici pruža istraživačima i praktičarima širok raspon alata za stvaranje bogatih i raznolikih digitalnih modela, pružajući podršku s primjenama od urbanog planiranja i kartografije, arhitekture do biološkog modeliranja i virtualne stvarnosti.

1.3. CGA (Computer Generated Architecture) pravila kroz primjere za unapređenje geovizualizacije 3d modela zgrada

Procedurno modeliranje uz pomoć CGA (*Computer Generated Architecture*) pravila omogućava stvaranje složenih 3D struktura korištenjem pravila i skripti. Ova tehnika se najčešće koristi u generiranju urbanih pejzaža, arhitektonskih modela i složenih scena, a ističe se svojom fleksibilnošću, brzinom i mogućnošću prilagodbe. U ovom tekstu istražiti ćemo temeljne aspekte CGA pravila, njihovu primjenu, alate poput *Esri CityEngine*-a, te detaljno opisati primjere kako bi se ilustrirale mogućnosti ove tehnike.

Procedurno modeliranje je tehnika generiranja 3D sadržaja kroz algoritme i pravila umjesto ručnog modeliranja. Temelj procedurnog modeliranja su *CGA pravila*, koja omogućuju programiranje generativnih procesa za stvaranje struktura.

CGA pravila su deklarativni jezik specifičan za područje procedurnog modeliranja. Omogućavaju definiciju pravila koja transformiraju osnovne geometrije poput poligona i linija u složene 3D oblike.

Ključne prednosti procedurnog modeliranja su:

- Automatizacija (kao cilj istraživanja): smanjenje vremena potrebnog za stvaranje velikih scena.
- Prilagodljivost: Jednostavne izmjene pravila kako bi se dobili različiti rezultati.
- Efikasnost: Generiranje tisuća objekata s minimalnim ručnim radom.

Primjeri primjene u mom istraživanju:

- Gradovi: Generiranje cijelih urbanih površina.
- Zgrade: Izrada modela zgrada na temelju pravila.

1.3.1. Osnovni koncepti CGA pravila

CGA pravila definiraju način na koji se osnovna geometrija transformira u kompleksne oblike. Ključni elementi CGA pravila uključuju sljedeće:

Početni oblik (*Start Shape*)

Svaki proces započinje osnovnom geometrijom, nazvanom početni oblik (*start shape*). To može biti pravokutnik, poligon ili linija.

Na primjer:

Lot --> Extrude(10)

Ovo pravilo primjenjuje operaciju *Extrude*, koja podiže osnovni poligon na visinu od 10 jedinica.

Pravila i hijerarhija

Pravila su definirana tako da svaki dio zgrade ima svoju ulogu (npr. fasada, krov, prozori). Pravila se mogu međusobno pozivati kako bi se kreirala hijerarhija.

Primjer:

Building -->

Mass -->

split(y) { 0.2: Base | ~2: Body | 0.1: Roof }

Ovaj kod dijeli osnovni volumen na tri dijela: bazu, tijelo zgrade i krov.

Parametri

Parametri omogućavaju prilagodbu modela na temelju ulaznih vrijednosti. Primjeri uključuju visinu, širinu, materijale i boje.

Primjer:

Ovaj kod dijeli osnovni volumen na tri dijela: bazu, tijelo zgrade i krov.

```
attr height = 20
```

```
Building -->
```

```
    Extrude(height)
```

Ovdje je visina zgrade kontrolirana atributom *height*.

Podjela (*Split*)

Geometrija se može dijeliti u različitim smjerovima (*x, y, z*).

Primjer:

```
Facade -->
```

```
    split(x) { 3: Window | 0.5: Frame }*
```

Ovaj kod dijeli fasadu na ponavljajuće prozore i okvire.

Ponavljanje (*Repeat*)

Omogućava beskonačno ponavljanje elemenata dokle god postoji prostor.

Primjer:

```
Wall -->
```

```
    repeat(y) { 2: Brick | 0.2: Mortar }
```

Zid je podijeljen na cigle i fuge.

1.3.2. Struktura CGA pravila za zgrade

Zgrade se modeliraju kroz modularni pristup. Glavni dijelovi uključuju:

Temelj (*Base*)

Temelj je obično prvi dio zgrade koji se definira. Uključuje oblikovanje osnovnog volumena.

Kod za temelj:

Base -->

```
    Extrude(1)
```

Ovaj kod stvara osnovu visine 1 jedinice.

Tijelo Zgrade (*Body*)

Glavni dio zgrade sadrži fasade, prozore i ukrasne elemente.

Kod za tijelo:

Body -->

```
    split(y) { ~2: Floor }*
```

Floor -->

```
    split(x) { 1: Window | 0.2: Frame }*
```

Ovo pravilo dijeli tijelo zgrade na katove, a svaki kat na prozore i okvire.

Fasada (*Facade*)

Fasada je detaljno oblikovana pomoću pravila za prozore, vrata i ukrase.

Kod za fasadu:

Facade -->

```
    split(x) { 3: Window | 0.5: Frame }*
```

Window -->

```
    color("blue")
```

Frame -->

```
color("gray")
```

Ovaj kod definira prozore plave boje i okvire sive boje.

Krov (*Roof*)

Krov se modelira posebnim pravilima koja omogućuju različite stilove (ravni, kosi, mansardni).

Kod za krov:

Roof -->

```
case height > 20: GableRoof
```

```
else: FlatRoof
```

FlatRoof -->

```
Extrude(1)
```

GableRoof -->

```
RoofGable(30)
```

Krov je definiran kao ravan ili dvostrešni ovisno o visini zgrade.

1.3.3. Primjeri procedurnog modeliranja zgrada

U prvom primjeru prikazan je kod za procedurno generiranje jednostavne zgrade.

Kod:

```
attr height = 15
```

Building -->

Base -->

```
Extrude(1)
```

Body -->

```
split(y) { ~height: Floor } *
```

Roof -->

FlatRoof

Floor -->

split(x) { 3: Window | 0.5: Frame }*

To je zgrada s karakteristikama da je:

- Osnovni volumen visine 15 jedinica.
- Katovi su podijeljeni na prozore i okvire.
- Ima ravan krov.

U drugom primjeru prikazan je kod za kompleksnu zgradu s detaljnim fasadama.

Kod:

attr numFloors = 5

attr floorHeight = 3

Building -->

Base -->

Extrude(2)

Body -->

split(y) { ~numFloors*floorHeight: Floor }*

Roof -->

GableRoof

Floor -->

split(x) { 2: Door | 1: Window | 0.3: Frame }*

Door -->

Extrude(3)

Window -->

color("blue")

Frame -->

color("white")

Objašnjenje:

- Pet katova, svaki visine 3 jedinice.
- Svaki kat ima vrata, prozore i okvire.
- Kos dvostrešni krov.

U trećem primjeru prikazano je generiranje urbane strukture bloka zgrada.

Kod:

attr blockWidth = 50

attr blockHeight = 20

Block -->

repeat(x) { ~blockWidth: Building }*

Building -->

Base -->

Extrude(2)

Body -->

split(y) { ~blockHeight: Floor }*

Roof -->

FlatRoof

Floor -->

split(x) { 3: Window | 0.5: Frame }*

Objašnjenje:

- Generira niz zgrada duž osi x.
- Svaka zgrada visine 20 jedinica s ravnim krovovima.

1.3.4. Prednosti i ograničenja CGA pravila

Prednosti:

- Automatsko generiranje složenih modela.
- Brze izmjene dizajna pomoću parametara.
- Pogodno za velike projekte (gradovi, kompleksi).

Ograničenja:

- Potrebna stručnost u pisanju CGA pravila.
- Ograničen realizam kod organskih oblika.

CGA pravila pružaju platformu za procedurno modeliranje zgrada. Koristio sam temeljne koncepte poput podjele, parametara i hijerarhije pravila i pomoću njih generirao složene 3D modele s minimalnim ručnim radom. Integracija ovih tehnika s alatima poput *Esri CityEngine*-a otvara vrata za širok spektar primjena u kartografiji i geovizualizaciji, arhitekturi, urbanizmu i industriji računalnih igara.

Sljedeća dva poglavlja opisuju istraživanje da bi se potvrdila hipoteza da automatizacija procedure gramatike može omogućiti generiranje 3D modela zgrada čak i kada nedostaju podaci poput 3D oblaka točaka ili Lidar podataka. Ovo je posebno obrađeno u kontekstu

inverznog procedurnog modeliranja u kojem se 3D modeli generiraju na temelju ograničenih ulaznih podataka ili potpuno bez njih, koristeći unaprijed definirane generativne gramatike.

Relevantni dijelovi:

- Inveržno procedurno modeliranje temeljeno na gramatikama
Ovo poglavlje istražuje metode generiranja gramatika ili derivacija modela na temelju ograničenih ili nepostojećih ulaznih podataka. Raspravlja se o različitim pristupima, uključujući:
 - Generiranje modela "od nule" koristeći simetrije i sličnosti.
 - Uporabu generičkih šablonskih gramatika za uparivanje s djelomičnim podacima.
 - Induktivne metode strojnog učenja koje generiraju slične modele na temelju nekoliko primjera.
- Faktorizacija funkcije preferencije
Ovo poglavlje objašnjava kako se procedure gramatike mogu prilagoditi korisničkim preferencijama, što uključuje modeliranje zgrada u situacijama kada nedostaju precizni ulazni podaci.

Istraživanje treba pokazati da procedure gramatike mogu djelovati kao alat za "popunjavanje praznina" u podacima, što je korisno za aplikacije poput urbanog planiranja, arhitektonske rekonstrukcije i simulacija.

1.3.5. Kategorije inverznih tehnika

Postojeće tehnike inverznog procedurnog modeliranja mogu se podijeliti u tri glavne kategorije, svaka s jedinstvenim pristupom i metodologijama. Od toga samo prva kategorija pokušava riješiti cjelovit inverzni problem stvaranjem potpuno novih gramatika. Ostale dvije kategorije usredotočene su na prilagodbu generičkih šablonskih gramatika ulaznim podacima ili na izvod stohastičkih gramatika na temelju ograničenog skupa označenih primjera, često zahtijevajući strukturirane i dobro označene ulazne podatke za učinkovit rad.

1.3.6. Od nule

Metode u ovoj kategoriji nastoje generirati nove gramatike bez oslanjanja na postojeće. Algoritmi u ovom području analiziraju ulazni model kako bi identificirali simetrije i sličnosti.

Podskupovi modela koji pokazuju slične transformacije mogu se grupirati i opisati rekurzivnim produkcijskim pravilima. Primjerice, razvijene su tehnike za obradu 2D vektorskih crteža koji se sastoje od osnovnih geometrijskih oblika poput linija, krivulja i trokuta (Št'ava et al., 2010).

U kontekstu 3D modeliranja, Bokeloh i suradnici (Bokeloh et al., 2010b) razvili su metodu za detekciju krutih transformacija i zrcalnih simetrija korištenjem voxel mreža za prepoznavanje ponavljajućih struktura (Aliaga et al., 2007; Bekins & Aliaga, 2005).

Za jednostavnije strukture poput fasada zgrada, zadatak postaje jednostavniji jer se mogu opisati s nekoliko pravila podjele. Rane metode zahtijevale su da korisnici ručno segmentiraju i označe elemente poput katova ili prozora, koji su se zatim kodirali u gramatici (Bekins & Aliaga, 2005). Moderni pristupi automatizirali su ovaj proces, poboljšavajući učinkovitost i upotrebljivost (Müller et al., 2007; F. Wu et al., 2013).

1.3.7. Prilagodba šablonskih gramatika

Ova kategorija uključuje prilagodbu unaprijed definiranih, generičkih, parametarskih gramatika ulaznim podacima. Takve su tehnike uspješno primijenjene na strukturirane fasade, gdje pravila podjele mogu predstavljati arhitektonske uzorke (Simon et al., 2011; Teboul et al., 2010, 2011). U 3D modeliranju ovaj pristup proširen je na specijalizirane primjene, poput rekonstrukcije dorskih hramova (Mathias et al., 2011).

Iako ove metode učinkovito rade za specifične i dobro definirane tipove ulaznih podataka, teško ih je generalizirati na šire kategorije, poput svih stilova zgrada. Nedavni napreci u strojnom učenju omogućili su usklađivanje skica s parametriziranim dijelovima gramatičkih pravila, poput prozora ili krovova, koji se zatim mogu kombinirati za stvaranje cjelovitih procedurnih gramatika (Nishida et al., 2016b).

Osim toga, tehnike koje koriste stohastičke gramatike pokušavaju pronaći najprikladniji derivat za specifične kriterije. Na primjer, Markovljevi chain Monte Carlo (MCMC) korišteni su za uzorkovanje prostora oblika L-sustava kako bi se pronašao derivat koji najbolje ispunjava zadani 3D volumen. Te napredne metode pretraživanja i isprobavanja različitih mogućnosti dodatno su poboljšane u novijim istraživanjima, čime je znatno povećana njihova učinkovitost – posebno kada se želi napraviti više varijanti složenih 3D oblika, poput zgrada s različitim volumenima i dimenzijama (Ritchie et al., 2015).

1.3.8. Indukcija gramatike

Posljednja kategorija crpi inspiraciju iz obrade prirodnih jezika. Na primjer, Bayesov model spajanja može naučiti jednostavnu stohastičku gramatiku iz skupa ulaznih primjera. Ova gramatika ne samo da generira zadane primjere, već i nove izlaze sa sličnim strukturnim značajkama (Stolcke & Omohundro, 1994).

U području 3D modeliranja, Talton i suradnici (J. Talton et al., 2012b) prvi su primijenili ovu tehniku, izvlačeći determinističke gramatike iz označenih grafova scena. Te gramatike zatim se generaliziraju spajanjem produkcijskih pravila ili se čine specifičnijima razdvajanjem pravila. Integracija MCMC uzorkovanja omogućava temeljito istraživanje prostora gramatika. Načelo minimalne duljine opisa osigurava da rezultatna gramatika balansira kompaktnost i sposobnost rekreiranja ulaznih primjera.

Inverzno procedurno modeliranje pruža moćne alate za rekonstrukciju, kompresiju i proširenje 3D modela koristeći pristupe temeljene na gramatikama. Podjela metoda na kreiranje gramatika *od nule*, prilagodbu šablona i indukciju gramatike omogućuje istraživačima odabir najprikladnije strategije za svoju specifičnu primjenu. Napredak u automatizaciji i strojnome učenju nastavlja širiti granice onoga što se može postići ovim tehnikama. Koristim operaciju dijeljenja pravila sličnu Taltonovoj za promjenu distribucije vjerojatnosti stohastičkih gramatika (Martinovic & Van Gool, 2013). Bayesova gramatička indukcija je naravno također primijenjena na fasadama.

1.3.9. Faktorizacija funkcije preferencije

Istražujem metodu faktorizacije funkcije preferencije, koja omogućava prilagodbu generativnih modela specifičnim željama korisnika. Ključni izazov u radu sa stohastičkim gramatičkim sustavima jest njihova sklonost generiranju modela na temelju unaprijed definiranih pravila, koja ne odgovaraju uvijek korisničkim očekivanjima. Faktorizacija preferencija pruža način da se ovi sustavi prilagode, omogućujući stvaranje modela koji bolje odražavaju želje korisnika bez potrebe za ručnom izmjenom složenih pravila. Predložena formula omogućuje modeliranje ili pojedinačne funkcije preferencije ili se može modelirati cjelokupna funkcija preferencija kao proizvod K čimbenici, tj. K funkcije pojedinačnih preferencija pomnožene zajedno:

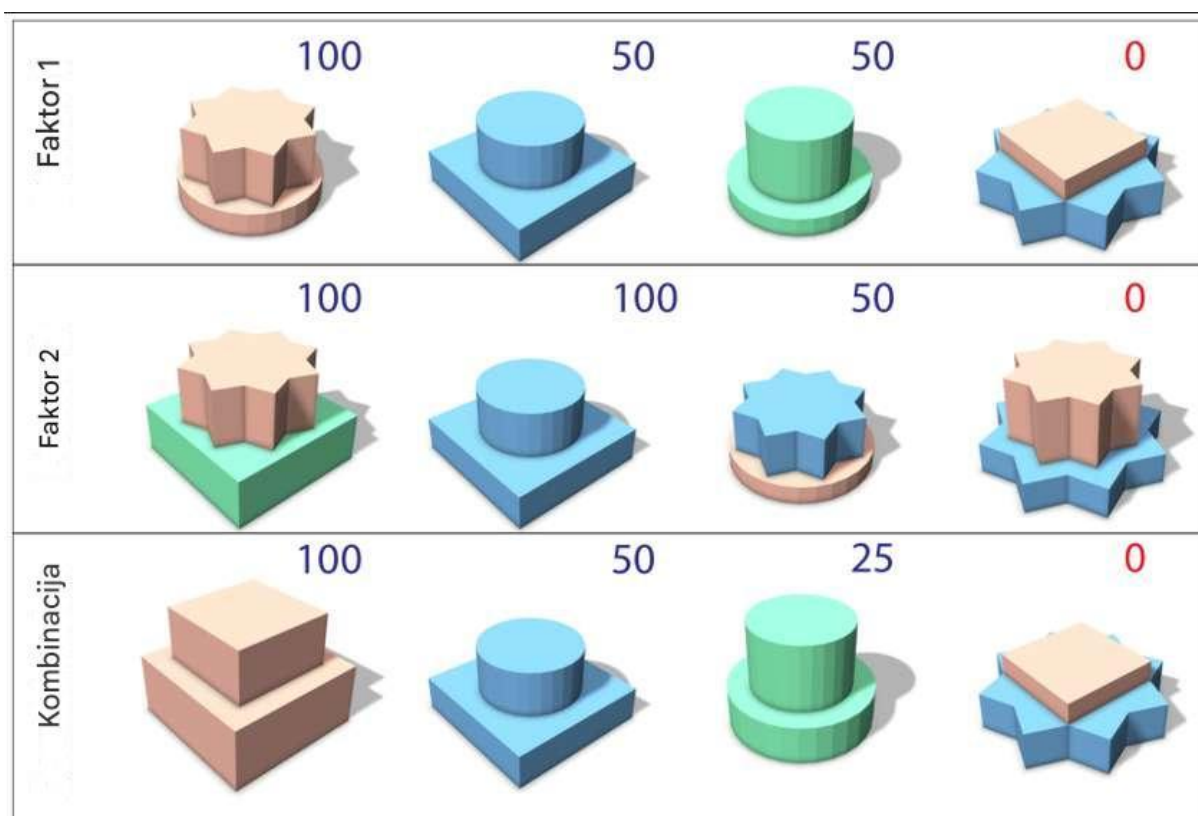
$$u(m) = \prod_{i=1}^K u^i(m)$$

gdje K odabire korisnik. Posjedovanje faktora omogućuje određivanje preferencija za pojedine aspekte pojedinačno, npr. za materijale, vrste prozora ili oblike krova. Često je lakše postaviti ocjene preferencija za pojedinačne aspekte nego za mnoge aspekte odjednom. U toj metodi koristi se Gaussova regresija procesa kako bi se ocjene korisnika na uzorcima modela interpolirale na cijeli prostor mogućih modela. Na taj način sustav može predvidjeti preferencije i za one modele koji nisu izravno ocijenjeni, čime se povećava preciznost i efikasnost prilagodbe. Naučene preferencije omogućuju izmjenu gramatičkih pravila kako bi se modeli usmjerili prema stilovima ili atributima koji su korisniku važni.

Funkcija preferencije razlaže se na više komponenti, pri čemu svaka od njih odražava određenu karakteristiku modela, poput oblika, proporcija ili estetskih značajki. Ovo razdvajanje olakšava preciznu prilagodbu, jer omogućuje detaljno razumijevanje kako različiti aspekti modela utječu na korisničke preferencije. Ovaj pristup nije samo tehnički koristan, već i intuitivan, jer omogućuje korisnicima da jednostavnije oblikuju rezultate prema svojim potrebama.

Faktorizacija funkcije preferencije pronalazi primjenu u mnogim područjima, uključujući urbanističko planiranje i arhitektonsko dizajniranje. Primjerice, može pomoći u generiranju različitih vrsta zgrada za određene gradske zone, osiguravajući da svaki model odgovara specifičnom kontekstu, poput stambenih, poslovnih ili industrijskih područja.

Ova metoda predstavlja važan korak prema personaliziranijem i preciznijem procedurnom modeliranju, omogućujući sustavima veću fleksibilnost i bolje zadovoljavanje potreba korisnika. Učenje preferencija kroz statističke metode čini ovaj proces bržim i intuitivnijim, pružajući snažan alat za prilagodbu složenih modela.



Slika 6. Preferencije korisnika kroz ocjenjivanje nekoliko uzoraka

Faktorizacija funkcije preferencije – Funkcija preferencije može se naučiti (AI) kao kombinacija čimbenika ili stupnjevita primjena Gaussove regresije procesa (GPR) na prostor oblika gramatičke fasade. Sustav uči preferencije korisnika kroz ocjenjivanje nekoliko uzoraka (prikazani kao obojeni krugovi) i interpolira ih na cijeli prostor oblika. Najpoželjniji oblici (crvena područja) sustav generira češće, dok se manje poželjni oblici (plava područja) generiraju rjeđe (Slika 6). Ovaj pristup omogućuje preciznu kontrolu nad distribucijom modela unutar stohastičke gramatike.

1.4. Procedurna gramatika i 3D modeliranje zgrada uz nedostatak podataka

Procedurna gramatika postala je moćan alat za generiranje složenih 3D modela zgrada definiranjem sustava pravila. Ovaj pristup osobito obećava u urbanom modeliranju i arhitektonskom dizajnu, gdje izvori podataka poput oblaka točaka ili Lidar skeniranja mogu biti nepotpuni ili nedostupni. Ovaj dokument sintetizira relevantna istraživanja o automatizaciji procedure gramatike, naglašavajući kako ona može omogućiti 3D modeliranje u uvjetima oskudnih podataka te predlaže metode za rješavanje povezanih izazova. Dodatno, istražuje se integracija procedure gramatike s naprednim algoritmima radi povećanja mogućnosti modeliranja.

1.4.1. Ključni doprinosi dosadašnjih istraživanja

Prilagodljivost procedure gramatike u svom radu (Simon, 2011) naglašava u stvaranju 3D urbanih modela unatoč ograničenjima u podacima. Predložene metode temelje se na pravilima koja interpoliraju između poznatih geometrija zgrada i generaliziraju prema neviđenim strukturama. Ovo omogućuje popunjavanje praznina u podacima korištenjem kontekstualnih urbanih obrazaca. Njegovo istraživanje pokazuje kako strukturirani pristup procedure gramatike pruža izvedive aproksimacije za nedostajuće podatke, otvarajući mogućnosti za automatizaciju u uvjetima ograničenih resursa.

Uporabu oblika gramatike predstavljaju (Mathias et al., 2011) u procedurnom modeliranju zgrada kroz prepoznavanje arhitektonskih uzoraka u postojećim podacima. Njihov rad uključuje mehanizme za automatizirano primjenjivanje pravila čak i kada ključni ulazni podaci, poput oblaka točaka, nisu dostupni. To omogućuje fleksibilnost u definiranju arhitektonskih pravila i generiranje modela temeljenih na detektiranim obrascima, čime se osigurava strukturna vjerodostojnost modela.

Na optimizaciju primjene procedurnih pravila izvlačenjem geometrije iz nepotpunih skupova podataka fokusiraju se (Demir & Aliaga, 2018). Njihova metoda vodi korisnički definirane radne procese kako bi se prilagodili ograničenjima ulaza i osigurali uvjerljivi rezultati u urbanim sredinama. Ovaj rad ističe važnost definiranja prilagodljivih pravila za suočavanje s nepotpunim podacima, nudeći praktično rješenje za arhitektonsko modeliranje.

Povezanost procedure gramatike i statističkog učenja za generiranje 3D modela zgrada istražuju (Dehbi et al., 2017). Predloženi algoritmi uče pravila iz postojećih podataka i ekstrapoliraju ih na nove modele, nadoknađujući nedostatke u informacijama. Statistički

pristupi obogaćuju procedurnu gramatiku omogućujući stvaranje uzoraka iz djelomičnih podataka, smanjujući potrebu za ručnim definiranjem pravila.

Prepravljanje pravila procedure gramatike kako bi poboljšali prilagodljivost i riješili nepravilnosti u podacima istražuju (Martin & Patow, 2019). Njihov rad naglašava kako procedurna pravila mogu evoluirati da bi odgovorila na arhitektonske varijacije i nedostatke u podacima. Pristup prepravljanja pravila proširuje primjenjivost procedure gramatike na raznolike skupove podataka, osiguravajući robusne rezultate.

1.4.2. Predloženi okvir za 3D modeliranje uz nedostatak podataka

Na temelju ovih istraživanja predlaže se sveobuhvatan okvir za procedurno modeliranje 3D zgrada koji uzima u obzir nedostatak podataka. Proces počinje analizom kvalitete i potpunosti dostupnih podataka te integracijom kontekstualnih izvora poput satelitskih snimaka. Definiranje procedurnih pravila prilagođava se specifičnostima arhitektonskog stila i urbanog konteksta. Primjena hijerarhijskih pravila omogućuje strukturiranje modela, dok se statistički modeli koriste za popunjavanje nedostajućih pravila iz postojećih skupova podataka koji su dostupni na internetu (poput OpenStreetMap-a, Google Open Buildings, CityGML modela u 3D Warehouseu, te dostupnih LiDAR i fotogrametrijskih baza podataka). Nadalje, iterativni proces provjere i ručnog ispravljanja osigurava točnost generiranih modela. Završni korak uključuje automatsku geovizualizaciju i generiranje 3D prikaza urbanih sredina, čime se omogućuje njihova daljnja analiza i korištenje. Kompletna dokumentacija je u dodatku B, zajedno s kodom i priložima uz detaljna objašnjenja. Ukoliko nema dostupnih ulaznih LIDAR podataka (oblak točaka) može se koristiti skriptnu datoteku „generate_test_ply.py“. Svi kodovi su na URL 4.

1.4.3. Rasprava i izazovi

Procedurna gramatika pokazuje značajan potencijal u generaliziranju podataka i generiranju vjerodostojnih modela čak i u uvjetima oskudnih podataka. No, suočavanje s izazovima nedostatka podataka zahtijeva dodatne prilagodbe pravila. Ključni izazov je definiranje dovoljno fleksibilnih pravila koja omogućuju pokrivanje raznih arhitektonskih stilova i prilagodbu nepredvidivim situacijama. Pri tome se često susreće problem prekomjerne generalizacije, koja može rezultirati modelima koji odstupaju od stvarnosti.

Jedan od najvećih izazova je integracija kontekstualnih podataka poput zračnih snimaka ili kartografskih podataka. Iako ti izvori mogu značajno poboljšati rezultate, oni uvode tehničke izazove poput usklađivanja različitih razmjera i formata podataka. Dodatno, semantička neslaganja između elemenata modela, primjerice geometrijskih i strukturnih nesklada, često zahtijevaju ručnu intervenciju. Ovdje se automatizacija suočava s ograničenjima koja zahtijevaju inovativne metode za premošćivanje tih neusklađenosti, odnosno nedovoljne standardizacije.

Skalabilnost je još jedan važan izazov, osobito kada se primjenjuje na velike urbane sredine. Procedurni sustavi često zahtijevaju značajne računalne resurse kako bi obradili kompleksne podatke i generirali detaljne modele. Metode poput statističkog učenja i prepravljanja pravila pružaju obećavajuće smjerove, no njihova primjena zahtijeva pažljivo balansiranje između točnosti i učinkovitosti.

Dodatni izazov leži u percepciji i primjeni generiranih modela. Automatski generirani modeli često se koriste u urbanom planiranju, arhitektonskom dizajnu ili vizualizacijama. Međutim, njihova uporabljivost ovisi o kvaliteti generacije i mogućnosti korisnika da prilagode modele specifičnim potrebama. Time je potrebno razviti interaktivne alate koji omogućuju jednostavnu manipulaciju i validaciju modela.

Istraživanja naglašavaju kako procedurna gramatika ima transformativni potencijal u modeliranju 3D zgrada, osobito u uvjetima oskudnih podataka. Njena sposobnost generaliziranja podataka i primjene pravila pruža uvjerljive mogućnosti za stvaranje vjerodostojnih modela urbanih sredina. No, kako bi se postigla potpuna automatizacija, potrebno je riješiti izazove povezane s nedostatkom podataka, semantičkim neskladima i skalabilnošću.

Predloženi okvir za procedurno modeliranje pruža smjernice za budući razvoj, uključujući prilagodbu pravila specifičnim arhitektonskim kontekstima, integraciju kontekstualnih podataka i razvoj interaktivnih alata za validaciju modela. Buduće istraživanje treba se usmjeriti na hibridne sustave koji kombiniraju automatizaciju i stručni nadzor, čime bi se osigurala primjenjivost procedurne gramatike u raznim disciplinama, od urbanog planiranja do virtualnih simulacija.

Dodatno, napredak u geovizualizaciji omogućuje bolje razumijevanje generiranih modela i njihovu primjenu u stvarnim scenarijima. Integracija ovih metoda s procedurnim pristupima

obećava revoluciju u načinu na koji modeliramo, analiziramo i vizualiziramo urbane prostore, čineći procedurnu gramatiku ključnim alatom za budućnost arhitekture i urbanizma.

1.5. Što je sve potrebno za automatsko procedurno modeliranje zgrada,

Automatsko procedurno modeliranje zgrada predstavlja proces generiranja 3D modela zgrada korištenjem algoritamskih tehnika, često temeljenih na parametrima ili pravilima koja definiraju izgled, strukturu i funkcionalnost zgrade. Ovaj pristup koristi procedurne metode za stvaranje zgrada u virtualnim okruženjima, kao što su video igre, simulacije ili arhitektonske vizualizacije. Takvo modeliranje omogućava stvaranje varijanti objekata na temelju unaprijed postavljenih pravila, što može značajno ubrzati dizajnerski proces i omogućiti generiranje kompleksnih struktura.

1.5.1. Osnove procedurnog modeliranja

Prvi pokušaji procedurnog modeliranja objekata, uključujući zgrade, datiraju iz 1960-ih i 1970-ih godina, kada je procedurno generiranje koristilo jednostavne algoritme za stvaranje osnovnih objekata u računalnim slikama i arhitekturi. U tim ranim fazama, razvoj je bio ograničen mogućnostima računala, ali je postavljen temelj za razvoj složenijih sistema. Jedan od pionira u ovom području bio je Ivan Sutherland, koji je 1963. godine razvio Sketchpad, prvi računalni alat za grafički dizajn. Iako se Sketchpad nije specifično bavio modeliranjem zgrada, njegov princip interaktivnog crtanja na ekranu bio je inspiracija za kasniji razvoj računalne grafike i modeliranja objekata (Scheler, n.d.).

1.5.2. Parametarski i pravilski pristupi

Postupno su se razvijali parametarski i pravilski sustavi, koji omogućuju generiranje različitih varijacija objekata na temelju postavljenih parametara. Ovi pristupi su se pokazali korisnima u arhitekturi, jer su omogućavali vizualnim dizajnerima postavke za osnovne smjernice u oblikovanje zgrada, dok su istovremeno ostavljali prostor za varijacije i personalizaciju (Gürsel Dino, 2012).

Jedan od ključnih radova u tom smjeru bio je tzv. Generativni dizajn koji se temelji na algoritmima evolucije i stohastičkim procesima. U ovom kontekstu, parametarski dizajn postao

je ključni alat za arhitekte i urbaniste, jer omogućava generiranje različitih oblika zgrada kroz variranje parametara poput visine, širine, proporcija i sl (Correia et al., 2023).

1.5.3. Procedurno generiranje u video igrama i vizualizacijama

Jedan od najpoznatijih područja primjene automatskog procedurnog modeliranja zgrada je industrija video igara, gdje se koristi za generiranje velikih gradova i kompleksnih urbanih područja. Minecraft, jedan od najpoznatijih naslova koji koristi procedurno modeliranje, omogućava igračima da grade zgrade u virtualnim svjetovima koristeći jednostavne blokove, dok je sam proces generiranja okoliša temeljen na algoritamskom pristupu. Slično tome, u video igri SimCity i njenim kasnijim verzijama, procedurno modeliranje koristi se za generiranje urbanih područja i simulisanje razvoja gradova. Ovdje se koristi algoritmička generacija kako bi se stvorili dinamični gradovi sa svim potrebnim infrastrukturnim elementima, poput cesta, zgrada i zona (Patterson & Ward, 2022).

1.5.4. Novi pristup za automatsko generiranje zgrada

U novije vrijeme, različite tehnike procedurnog modeliranja integriraju se s naprednim tehnologijama poput umjetničke inteligencije (AI) i strojno učenje za još sofisticiranije generiranje zgrada.

Deep Learning i Generative Adversarial Networks (GANs): Neke od novijih istraživanja koriste Generative Adversarial Networks (GANs) za stvaranje realističnih zgrada i arhitektonskih dizajna. GAN-ovi omogućavaju stvaranje novih objekata na temelju postojećih podataka, što omogućava "naučene" generativne modele koji mogu stvarati inovativne i kompleksne arhitektonske oblike (A. N. Wu et al., 2022).

Procedurna generacija u BIM (Building Information Modeling): Procedurno modeliranje se sve više integrira u BIM sustave, koji omogućuju stvaranje zgrada s velikim detaljima u pogledu građevinskih specifikacija, struktura i materijala. Ovaj pristup omogućava arhitektama, inženjerima i izvođačima da generiraju 3D modele zgrada koji su funkcionalni i kompatibilni sa stvarnim građevinskim zahtjevima (Panteli et al., 2020).

1.5.5. Generativni dizajn i održivost

U kontekstu održivog dizajna, procedurno modeliranje omogućava stvaranje zgrada koje optimiziraju prostor, energiju i resurse. Korištenjem algoritama, moguće je generirati arhitektonske strukture koje smanjuju potrošnju energije ili koriste obnovljive izvore energije, čime se povećava ekološka održivost zgrada.

Korištenje algoritama poput genetskih algoritama, simulacija evolucije ili topološke optimizacije za stvaranje energetski učinkovitih zgrada koje automatski prilagođavaju svoj dizajn ovisno o uvjetima okoline, smanjuje potrebu za ručnim prilagodbama i može dovesti do većih ušteda u građevinskoj industriji.

Da bi se pokazalo da automatizacija u modeliranju zgrada ima ograničenja i da je ručna dorada često potrebna zbog semantičkih nesklada, preporuča se pristup koji kombinira teorijski, praktični i eksperimentalni dio.

Automatizacija u modeliranju zgrada može doseći samo određeni stupanj preciznosti jer se oslanja na pravila i algoritme koji prepoznaju obrasce, simetrije i geometrijske odnose. No, izazov se javlja kada automatizirani proces naiđe na složene arhitektonske detalje ili nepredvidive elemente dizajna koji zahtijevaju semantičko razumijevanje. Na primjer, tijekom automatskog generiranja fasade, sustav može pravilno identificirati prozore i vrata, ali neće prepoznati kontekstualne odnose poput povijesne važnosti određenog elementa ili njegove povezanosti s okolnim strukturama.

Jedan od najučinkovitijih načina da se demonstrira ovo ograničenje je putem praktične studije slučaja. Korištenjem automatiziranih alata za izradu modela, poput procedurnog generiranja, može se generirati model zgrade s elementima koji su geometrijski i strukturno pravilni. No, već pri pokušaju obrade zgrada s kompleksnim stilovima ili nepravilnostima, rezultati postaju nepouzdana. Primjerice, nepravilnosti u geometriji krova ili odstupanja u veličini prozora unutar istog pročelja često uzrokuju da automatizirani alati primjene univerzalna pravila, što rezultira neskladom.

Semantički nesklad može se posebno jasno uočiti u situacijama kada su dijelovi modela međusobno strukturno različiti. Automatizacija možda neće prepoznati da specifičan uzorak na fasadi nosi estetsku ili funkcionalnu vrijednost. Samo jedna takva neusklađenost može narušiti cjelokupni model, stvarajući rezultat koji je tehnički ispravan, ali kontekstualno pogrešan. Ovo ukazuje na ključnu ulogu ljudske intervencije, gdje dizajner mora prilagoditi parametre ili ručno korigirati dijelove modela kako bi uskladio geometrijsku točnost s dizajnerskom namjerom.

Kombinacija automatiziranih alata i ručne dorade također se može testirati uspoređivanjem modela generiranih isključivo automatiziranim putem s onima koji su prošli kroz ljudsku reviziju. Takva analiza jasno pokazuje kako ručno prilagođavanje ispravlja nesklad između dijelova modela i omogućuje postizanje veće semantičke dosljednosti. Upravo ovaj proces ručne intervencije ističe granicu do koje automatizacija može dosegnuti i naglašava potrebu za ljudskom stručnošću u 3D modeliranju.

2. Predložena poboljšanja parametrizacije zgrada spajanjem oblika koja su rezultat kombinacija konstrukcijskih oblika koristeći više različitih CGA pravila

Automatsko kreiranje 3D modela iz slika ili oblaka točaka predstavlja izazov zbog raznolikosti i složenosti građevinskih objekata.

Kako bi se uspješno riješili ovi izazovi, važno je koristiti unaprijed poznate informacije koje pomažu u razumijevanju strukture građevina. Formalne gramatike omogućuju opisivanje odnosa i struktura dijelova zgrada, a standard poput CityGML-a koristi se za definiranje geometrije, topologije i izgleda urbanih objekata u različitim razinama detalja.

U ovom poglavlju ispituje se novi pristup obradi problema uvođenjem automatskog učenja pravila gramatike koja se koriste za rekonstrukciju fasada 3D modela građevina. Za razliku od tradicionalnih metoda koje zahtijevaju ručno definiranje pravila, predloženi pristup koristi strojno učenje kako bi automatizirao taj proces, čime se smanjuje potreba za stručnim znanjem i značajno ubrzava proces. Pristup je primjenjiv na razne stilove fasada, uključujući složene 3D strukture. Nadam se da ću pokazati kako se strojno učenje, uključujući statističke metode i logičke mreže, može koristiti za rekonstrukciju zgrada. Takav pristup omogućuje modeliranje građevinskih struktura, njihovih odnosa i svojstava, u slučajevima kada postoje nesigurnosti u podacima uzrokovane nedostacima ili preprekama u dostupnim informacijama.

Oblikovne gramatike su formalni sustavi koji definiraju skup pravila za generiranje složenih oblika ili struktura kroz sukcesivne primjene tih pravila. U kontekstu 3D modeliranja zgrada, oblikovne gramatike omogućuju procedurno stvaranje modela zgrada primjenom definiranih pravila koja određuju kako se osnovni elementi mogu kombinirati i modificirati kako bi se dobile složene građevine, čime se omogućuje dosljednu i učinkovitu kartografsku vizualizaciju velikog broja različitih zgrada, što je posebno korisno u rekonstrukciji 3D objekata i urbanih scena.

2.1. Slična istraživanja

Izrada 3D modela zgrada pomoću gramatičkih pravila ima veliku ulogu u rekonstrukciji 3D objekata. Pravila gramatike u procedurnom modeliranju omogućuju generiranje velikog broja zgrada. Na primjer, (Müller Gang; Wonka Peter; Van Gool Luc, 2007) predstavili su sustav temeljen na CGA oblikovnoj gramatici za dosljedno generiranje modela zgrada, ali gramatička

pravila u njihovom pristupu ručno su definirana, što zahtijeva stručno znanje i dosta vremena. S druge strane, obrnuto procedurno modeliranje koristi formalne gramatike za rekonstrukciju stvarnih zgrada. (Ripperda & Brenner, 2009) kombinirali su unaprijed definirana pravila gramatike s metodom Markovljevih lanaca za analizu fasada. (Martinovic & Van Gool, 2013) uveli su pristup učenju Bayesovih gramatičkih pravila iz fotografija za generiranje i rekonstrukciju 2D fasada. Iako je ovaj pristup inovativan, ograničen je na ravne fasade.

Oblikovne gramatike za analizu fasada temeljenih na uzorcima rešetki koristili su (Teboul et al., 2011), dok je (Becker, 2009) kombinirao podatke iz fotografija i laserskih skenera za identifikaciju prozora i vrata. Ipak, većina ovih metoda pretpostavlja da su fasade jednostavne 2D plohe, što otežava njihovu primjenu na složenije strukture.

Znamo da pravila (treebanks) igraju ključnu ulogu u modeliranju jezičnih struktura (Charniak, 1996; Tsochantaridis et al., 2004).

Za početak treba definirati Markovljeve logičke mreže (MLN) kombiniraju Markovljeve mreže i prvorednu logiku kako bi omogućile modeliranje nesigurnosti u složenim sustavima. One predstavljaju skup prvorednih logičkih formula, pri čemu svaka formula ima pridruženu težinu koja odražava njenu važnost ili vjerojatnost. Glavna prednost MLN-a je njihova sposobnost da integriraju simboličko zaključivanje s vjerojatnosnim modeliranjem, što ih čini korisnima u područjima poput obrade prirodnog jezika, računalnog vida, analize podataka i umjetne inteligencije. U kontekstu analize fasada i 3D modeliranja zgrada, MLN se mogu koristiti za prepoznavanje obrazaca i kombiniranje različitih izvora podataka (poput fotografija i laserskih skenera) radi preciznije rekonstrukcije arhitektonskih elemenata.

Općenita formula za Markovljeve logičke mreže (MLN) temelji se na definiranju skupa prvorednih logičkih formula F_i s pridruženim težinama w_i . Vjerojatnost dodjeljivanja istinitosnih vrijednosti skupu predikata određuje se pomoću Markovljeve distribucije:

$$P(X = x) = \frac{1}{Z} \exp \left(\sum_i w_i n_i(x) \right)$$

gdje je:

- X - skup svih mogućih dodjeljivanja istinitosnih vrijednosti predikatima,
- x - konkretna instanca dodjele,

- w_i - težina formule F_i
- $n_i(x)$ - broj instanci formule F_i koje su istinite u dodjeli x ,
- Z - normalizacijska konstanta (tzv. particijska funkcija), definirana kao

$$Z = \sum_{x'} \exp \left(\sum_i w_i n_i(x') \right)$$

Ova formula pokazuje kako MLN kombiniraju prvorednu logiku i Markovljeve mreže tako da težine pravila utječu na vjerojatnosnu distribuciju mogućih svjetova (instanci). Što je veća težina formule, to je veća vjerojatnost da će se i izvršiti u konačnoj interpretaciji podataka.

Markovljeve logičke mreže (MLN) koriste se za kombiniranje logike i vjerojatnosti, omogućujući modeliranje nesigurnosti u podacima (Richardson Pedro, 2006). Najbolje su pokazali primjenu MLN-a u zadacima poput razlikovanja entiteta (Singla & Domingos, 2006), što je slično problemu analize fasada. Ovaj pregled ukazuje na potrebu za metodama koje mogu modelirati složenost i nesigurnost u stvarnim podacima, što je glavni cilj u istraživanju u ovom poglavlju.

2.1.1. Poboljšanje parametrizacije

Istražit ću mogućnosti unapređenja oblikovanja i parametrizacije, kao i primjenu automatizacije u praksi. Također će se analizirati u kojoj je mjeri ljudska intervencija i dalje ključna za postizanje najboljih rezultata. Poseban naglasak bit će na praktičnim primjerima i studijama slučaja koje prikazuju prednosti i izazove kombiniranja automatiziranih i ručnih metoda.

U kontekstu 3D rekonstrukcije zgrada, formalne gramatike dobivaju sve veću pozornost (Musialski Peter; Aliaga Daniel G.; Wimmer Michael; Van Gool Luc; Purgathofer Werner, 2012). Formalne gramatike adekvatno opisuju agregaciju zgrada, kao i odnose koji postoje između njihovih dijelova. Koriste se za generiranje sintetičkih modela grada (procedurno modeliranje) (Vanegas Daniel G.; Wonka Peter; Müller Pascal; Waddell Paul; Watson Benjamin, 2010), kao i za rekonstrukciju zgrada i njihovih dijelova. Do sada, osim nekoliko pristupa (Ripperda & Brenner, 2009) i (Becker, 2009) gdje su pokušali izvući gramatička pravila iz danih podataka (npr. fotografije), pravila su uglavnom ručno izvedena.

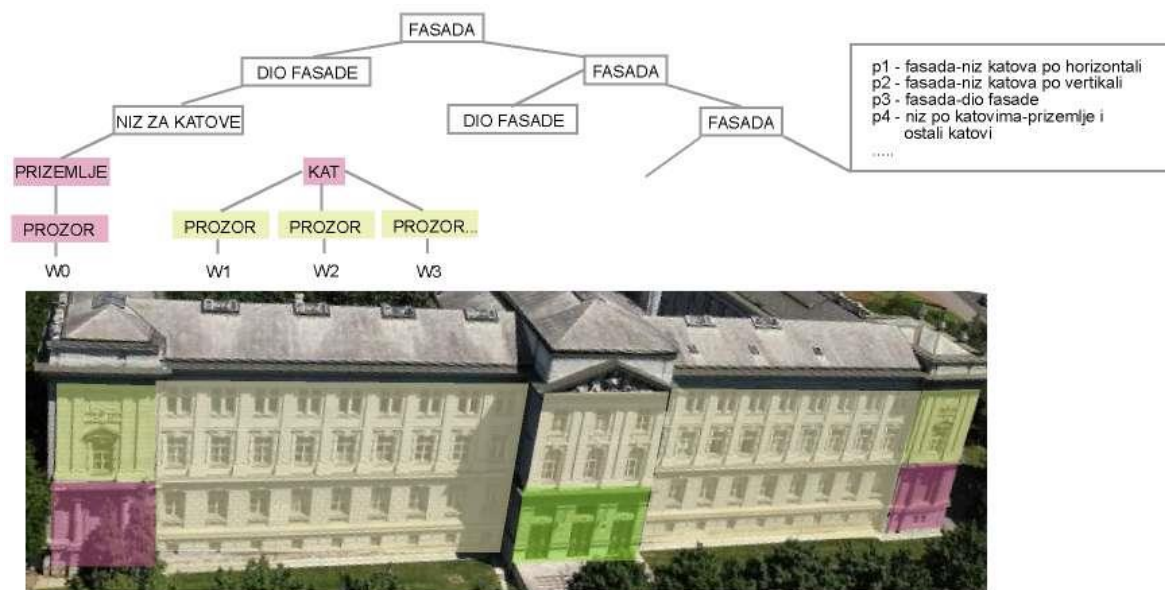
Novi pristup za automatsko učenje ponderirane atribuirane kontekstno-slobodne gramatike (WACFG – *Weighted Attributed Context-Free Grammar*) za identifikaciju i rekonstrukciju fasada u 3D modelima zgrada. Naša metoda bavi se ne samo ravnim fasadama, već i fasadama

sa sofisticiranim 3D strukturama. Gramatike atributa – ključni koncept u našem pristupu – proširuju gramatike bez konteksta atributima i semantičkim pravilima. U kontekstu 3D modeliranja ovo daje veću izražajnu komponentu u vizualizacijskom smislu.

To bi bio novi i drugačiji pristup automatskom učenju ponderirane atribuirane kontekstno-slobodne gramatike (WACFG) za 3D rekonstrukciju zgrada. Za razliku od procedurnih metoda, predloženi pristup je deklarativan, što znači da odjeljuje način prikaza zgrada i njihovih elemenata od samog procesa rekonstrukcije. WACFG je korišten i za modeliranje i za rekonstrukciju, pri čemu opisuje taksonomsku i partonomsku strukturu zgrada koristeći ponderiranu kontekstno-slobodnu gramatiku (WCFG). Dodatno, na sustav su primijenjena značajna ograničenja, definirana metodama statističkog relacijskog učenja (SRL), konkretno pomoću Markovljevih logičkih mreža (MLN).

Ključni pojmovi i njihova objašnjenja:

- Ponderirana atribuirana kontekstno-slobodna gramatika (WACFG- Weighted Attributed Context-Free Grammar) – proširenje kontekstno-slobodnih gramatika (CFG), gdje se svakom simbolu dodjeljuju dodatni atributi i pravila imaju pridružene težine koje određuju njihovu vjerojatnost primjene.
- Taksonomska struktura – hijerarhijska organizacija elemenata zgrade prema vrstama i kategorijama (npr. zgrada → katovi → prozori).
- Partonomska struktura – opisuje odnose dijela i cjeline u modeliranju (npr. fasada je dio zgrade, a prozori su dio fasade).
- Ponderirana kontekstno-slobodna gramatika (WCFG) – verzija kontekstno-slobodnih gramatika gdje pravila imaju pridružene težine koje utječu na način generiranja modela.
- Statističko relacijsko učenje (SRL) – skup metoda strojnog učenja koje rade s podacima koji su organizirani u mreže odnosa, čime se omogućuje analiza složenih struktura kao što su zgrade.
- Markovljeve logičke mreže (MLN) – kombinacija Markovljevih mreža i prvredne logike, koja omogućuje modeliranje nesigurnosti i veza između različitih elemenata zgrada pomoću pravila i vjerojatnosti.



Slika 7. Pročelje ili prednja fasada muzeja Mimara modelirana težinskom atributnom gramatikom.

Struktura pročelja (Slika 7) opisana je izvedenim stablom uz rasčlambu (vrh u sredini) iz ponderiranog dijela gramatike bez konteksta (gore desno). Ograničenja i atributi dijelova zgrade modelirani su na relacijski način pomoću Markovljevih logičkih mreža (dolje).

Po principu:

A₁: ista Širina (a,b) – slična (a,b)

A₂: ista Visina (a,b) – slična (a,b)

A₃: isto Horizontalno poravnanje (a,b) – slična (a,b)

A₄: – slična (a,b), slična (b,c) – slična (a,c)

.....itd



Slika 8. Pojedinačni dijelovi središnjeg dijela zgrade Mimare.

Slika 8 prikazuje prednju fasadu muzeja Mimara, zajedno s njezinim stablom raščlanjivanja, koje vizualizira taksonomsku i partonomsku strukturu fasade. Partonomska struktura prikazuje način na koji se fasada sastoji od pet dijelova, koji su rekurzivno kombinirani prema pravilima dobivenim iz inducirane ponderirane kontekstno-slobodne gramatike (WCFG). Izvadak ovih pravila prikazan je u okviru na vrhu slike, pri čemu P_i označava vjerojatnost primjene pojedinog pravila u skladu s definiranim distribucijama unutar gramatike. U donjem lijevom okviru prikazane su logičke formule Markovljevih logičkih mreža (MLN). Težine k_i , koje su automatski naučene, odražavaju važnost pojedinih formula. Ove formule definiraju strukturna ograničenja – primjerice, pravila vertikalnog i horizontalnog poravnanja – kao i pozadinsko znanje, poput informacija o susjedstvu i visini katova unutar modeliranih 3D objekata. Za razliku od klasičnih metoda statističkog učenja, statističko relacijsko učenje (SRL) omogućuje simbolički prikaz stanja, umjesto eksplicitnog nabiranja svih mogućih konfiguracija. Prednost ovog pristupa leži u sposobnosti da sažeto predstavi vjerojatnosne ovisnosti između atributa povezanih objekata, čime se omogućuje učinkovito dijeljenje parametara između sličnih elemenata. Također, SRL metode omogućuju kombinaciju nesigurnosti opažanja i strukturnih modela, što ih čini pogodnim za složene zadatke rekonstrukcije. Proces učenja WCFG-a omogućuje modeliranje fasadnih struktura, s posebnim naglaskom na njihov način spajanja i organizacije. Osim toga, omogućuje analizu distribucije i važnosti različitih strukturnih obrazaca, čime se proširuju klasična gramatička pravila koja ne uzimaju u obzir kontekst. WCFG-ovi omogućuju modeliranje objekata s nepoznatim brojem parametara, poput broja katova i prozora, što je ključno za fleksibilnu rekonstrukciju arhitektonskih elemenata. Naš

pristup eksplicitno uzima u obzir nesigurnost opažanja, kombinirajući projektivnu geometriju, probabilistička pravila i Markovljeve logičke mreže (MLN). Sve komponente WACFG-a automatski se uče iz primjera, pri čemu se gramatička pravila i njihove vjerojatnosti uče pomoću SVM-ova (podrškom vektorskih strojeva), dok se MLN modeli treniraju pomoću statističkih relacijskih metoda učenja. Naučeni WACFG primjenjuje se za rekonstrukciju zgrada na temelju opažanja, koristeći SVM klasifikaciju i MLN zaključivanje (dodatak B, URL 5). Koliko mi je poznato, ovo je prva demonstracija primjene naprednih tehnika strojnog učenja – posebno statističkog relacijskog učenja – za 3D rekonstrukciju zgrada.

Novi pristup automatskom učenju kontekstno-slobodne gramatike s ponderiranom atribucijom za analizu 3D oblaka točaka prikazan je na (Slika 9). Kako bi se učinkovito primijenilo strojno učenje, primjenjujemo dva glavna koraka: prvo se uči struktura fasada, a zatim se određuju parametri građevinskih elemenata i pridružena ograničenja. U početnoj fazi generira se ponderirana kontekstno-slobodna gramatika (WCFG) temeljena na relacijskoj bazi podataka građevina (RBDB). Ova baza obuhvaća 1300 označenih fasada iz različitih regija u Njemačkoj, pri čemu se proces učenja vodi pomoću pseudo-logaritamske mjere vjerojatnosti. Koristim Markovljeve logičke mreže (MLN), implementirane kroz Alchemy—sustav otvorenog koda (alchemy.cs.washington.edu/). Po zadanim postavkama, Alchemy primjenjuje algoritam MaxWalkSat (MWS) za MAP zaključivanje, dok alternativno može koristiti i napredne metode poput max-product Belief Propagation. Kako bi se dodatno iskoristilo probabilističke sposobnosti MLN-a, oslanjamo se na pristup Kok i Domingosa (2005.), koji koriste pretraživanje za optimizaciju strukture i parametara modela. Na taj način naš sustav omogućuje detaljnu analizu fasadnih struktura i njihovu adaptaciju na raznolike građevinske stilove.

2.1.2. Predznanje

Potreba za trodimenzionalnim (3D) modelima zgrada visoke rezolucije dramatično je porasla u posljednjem desetljeću. Koriste se u različitim zadacima i aplikacijama kao što je e-planiranje za učinkovitu komunikaciju i upravljanje sofisticiranim urbanim prostornim okruženjima (Engel Jürgen, 2012). Automatsko izdvajanje takvih modela iz slika ili 3D oblaka točaka izazovan je zadatak. To se pripisuje raznolikosti i složenosti predmeta koje je napravio čovjek. Kako bi se nosilo s ovom poteškoćom, ključno je uključivanje prethodnog znanja predstavljenog kroz 3D modele. U tu je svrhu eksplicitno modeliranje semantike ključni zadatak. U tom kontekstu uveden je CityGML (Henn Christoph; Gröger Gerhard; Plümer Lutz,

2011) koji osim semantike definira 3D geometriju, topologiju i izgled urbanih objekata u različitim razinama detalja.

U kontekstu 3D rekonstrukcije zgrada, formalne gramatike dobivaju sve veću pozornost (Musialski Peter; Aliaga Daniel G.; Wimmer Michael; Van Gool Luc; Purgathofer Werner, 2012). Formalne gramatike adekvatno opisuju agregaciju zgrada, kao i odnose koji postoje između njihovih dijelova. Koriste se za generiranje sintetičkih modela grada (procedurno modeliranje) (Vanegas Daniel G.; Wonka Peter; Müller Pascal; Waddell Paul; Watson Benjamin, 2010), kao i za rekonstrukciju zgrada i njihovih dijelova. Do sada, osim nekoliko pristupa (Becker, 2009; Ripperda & Brenner, 2009) koji su pokušali izvući gramatička pravila iz danih podataka (npr. slike) pravila su uglavnom ručno izvedena. Međutim, to je skup i naporan posao koji zahtijeva stručno znanje.

Cilj u ovom dijelu istraživanja je napraviti korak prema automatskom strojnom učenju takvih pravila. Predlažemo novi pristup za automatsko učenje ponderirane atribuirane gramatike za identifikaciju i rekonstrukciju fasada u 3D modelima zgrada. Ova metoda bavi se ne samo ravnim fasadama, već i fasadama sa sofisticiranim 3D strukturama. Gramatike atributa – ključni koncept u našem pristupu – proširuju gramatike bez konteksta atributima i semantičkim pravilima. U kontekstu 3D modeliranja ovo daje veću izražajnu snagu u cilju modeliranja ograničenja između primitiva modeliranih 3D objekata. Na taj se način mogu adekvatno modelirati geometrijska, topološka i semantička ograničenja koja karakteriziraju objekte koje je čovjek izgradio.

Glavni doprinos je automatsko strojno učenje ponderirane atribuirane kontekstno-slobodne gramatike (WACFG) za 3D rekonstrukciju zgrada. Za razliku od procedurnih metoda, predlažemo deklarativni pristup koji odvaja prikaz zgrada i njihovih dijelova od zadatka rekonstrukcije. Koristio sam WACFG za modeliranje kao i za zadatke rekonstrukcije. WACFG opisuje taksonomsku i partonomsku strukturu zgrada ponderiranom kontekstno-slobodnom gramatikom (WCFG) i značajnim ograničenjima koja su opisana pomoću metoda statističkog relacijskog učenja (SRL) i Markovljevih logičkih mreža (MLN).

Težine W_i su automatski naučene i označavaju važnost povezane formule F_i . Formule pokazuju ograničenja, npr. vertikalna i horizontalna poravnanja i nedostatak podataka o pozadinskoj fasadi zgrade, npr. informacije o susjedstvu i katu, temeljnih 3D objekata i sl.

SRL modeli, za razliku od onoga što se tradicionalno radi u statističkom učenju, nastoje izbjeći eksplicitno nabrojanje stanja, koristeći simbolički prikaz stanja. Prednost ovih modela leži u

sposobnosti sažetog predstavljanja vjerojatnosnih ovisnosti među atributima različitih povezanih objekata. To omogućuje kompaktni prikaz naučenih modela koji omogućuju dijeljenje parametara sličnih objekata. Osim toga, SRL metode omogućuju kombiniranje nesigurnosti opažanja kao i strukturnih modela. Učenje WCFG-a omogućuje modeliranje fasadnih struktura, posebice njihovo spajanje u različitim dijelovima. Nadalje, ovo daje uvid u distribuciju i važnost različitih strukturnih obrazaca prema težinama koje proširuju klasična gramatička pravila bez konteksta. WACFG-ovi omogućuju modeliranje objekata s *a priori* nepoznatim brojem parametara kao što su broj katova i prozora. Tu se bavim nesigurnošću opažanja neizvjesnom projektivnom geometrijom, probabilističkim pravilima i MLN-ovima. Sve u svemu, WACFG nam omogućuje da se nosimo sa složenošću i raznolikošću zgrada iz stvarnog svijeta. Sve komponente WACFG-a se automatski uče iz primjera (Slika 9). Gramatička pravila i njihove vjerojatnosti uče SVM-ovi, MLN se uči pomoću statističkih relacijskih metoda učenja. Naučeni WACFG primjenjuje se za rekonstrukciju zgrada iz opažanja korištenjem klasifikacije u cilju modeliranja ograničenja između primitiva modeliranih 3D objekata. Na taj se način mogu adekvatno modelirati geometrijska, topološka i semantička ograničenja koja karakteriziraju objekte koje je izradio čovjek.

2.2. Formalne gramatike

Formalna gramatika GGG definira se kao četvorka: početni simbol SSS, skup neterminala NNN, skup terminala TTT i skup pravila produkcije RRR (Chomsky, 1956, 1959). Pravila produkcije omogućuju zamjenu neterminalnog simbola s odgovarajućim nizom terminalnih i neterminalnih simbola. Kontekstno slobodne gramatike (CFG) specifične su po tome što lijeva strana pravila može sadržavati samo jedan neterminalni simbol. Ovo svojstvo olakšava modeliranje hijerarhijskih struktura, poput dijelova zgrada. Za proširenje mogućnosti CFG-a, (Knuth, 1968, 1971) uveo je atributne gramatike, gdje se neterminalima i terminalima dodaju atributi, dok semantička pravila definiraju odnose između tih atributa. Ovaj pristup omogućuje modeliranje geometrijskih i topoloških ograničenja.

2.3. Upotreba MLN-a

Markovljeve logičke mreže (MLN) omogućuju kombinaciju logike i probabilističkih grafičkih modela. Za razliku od klasične logike, svakoj formuli u MLN-u pridružuje se težina. MLN se, kada je u potpunosti uključen, predstavlja kao Markovljevo slučajno polje (MRF), što

omogućuje primjenu tehnika probabilističkog zaključivanja poput Gibbsovog uzorkovanja ili vjerojatnosti. MLN se definiraju pomoću formulacija u logici prvog reda. Formule se konstruiraju pomoću konstanti, varijabli, funkcija i predikata. Da bi se ograničilo složenost, ovdje koristim logiku prvog reda bez funkcija. Konstante predstavljaju objekte u domeni, poput Prozora (Window1) ili vrata (Door1), dok varijable xxx pokrivaju sve objekte u obuhvatu. Predikati vraćaju istinitu vrijednost i definiraju attribute objekata ili odnose između njih. Na primjer, unarni predikat $isDoor(A)$ vrijedi ako je objekt AAA vrata, dok binarni predikat $similar(A,B)$ vrijedi ako su objekti AAA i BBB istog tipa.

MLN se definira kao skup parova (F_i, k_i) , gdje je F_i formula u logici prvog reda, a k_i težina formule. Konstrukcija MRF-a temelji se na uzemljenim predikatima MLN-a, pri čemu svaki uzemljeni predikat odgovara binarnoj varijabli. Težina k_i pridružuje se svakoj potencijalnoj funkciji povezanoj s formulom F_i . Za automatsko učenje težina MLN-a često se maksimizira pseudo-vjerojatnost umjesto log-vjerojatnosti $p(x,w)p(x,w)$ za dane podatke.

Za konstrukciju strukture MLN-a, tj. formula prvog reda, iz skupa podataka, (Kok & Domingos, 2005) koristili su pretragu vođenu pseudo-log-likelihood mjerom. Referentna implementacija MLN-a, uključujući učenje strukture i parametara, dostupna je kroz otvoreni alat Alchemy 2.0 (<https://alchemy.cs.washington.edu/>). Alchemy koristi MaxWalkSat (MWS) za MAP zaključivanje, ali omogućuje i uporabu algoritama npr za vjerojatnost.

U biti, MLN-ovi su definirani formulama prvog reda. Formula prvog reda je konstruirana pomoću konstanti, varijabli, funkcija i predikata. Da bi ipak ograničio složenost, odnosno pojednostavio strukturu, može se ograničiti na logiku prvog reda (odnosno logičke varijable) bez funkcija. Konstante predstavljaju objekte u našoj domeni, npr. Prozor1, Prozor2, Vrata1 i tako dalje. Logička varijabla x definirana je pomoću domena objekata. Predikati vraćaju istinitu vrijednost i definiraju attribute objekata ili odnosa između objekata. Na primjer, unarni predikat $isDoor(A)$ je istinit ako je objekt A vrata, dok je binarni predikat $sličan(A, B)$ istinit ako su objekti A i B istog tipa, možda uključuju topološka i geometrijska svojstva. Atom je predikat primijenjen na skup termina, a termin može biti konstanta ili varijabla. Osnovni pojam je pojam koji ne sadrži nikakve varijable i sukladno tome, atom je predikat čiji argumenti su svi temeljni pojmovi. Dakle, x i "Prozor1" su termini i $sličan(x, Prozor1)$.

Ovdje je F_i formula u logici prvog reda, a k_i je težina formule F_i .

Osim toga imamo i skup konstanti $C5fC1$ za koji je potreban $CjCjg$. Sada se MLN može koristiti za konstruiranje MRF-a. Za svaki osnovni predikat u MLN binarna varijabla se dodaje u MRF, npr. sličan (Prozor1, Prozor2) odgovara jednoj binarnoj varijabli u MRF-u. Vrijednost slučajne varijabla modelira istinito stanje osnovnog predikata. Zbog konačnog skupa konstanti, ovo rezultira konačnim skupom slučajnih varijabli (vidi dodatak A, URL 6).

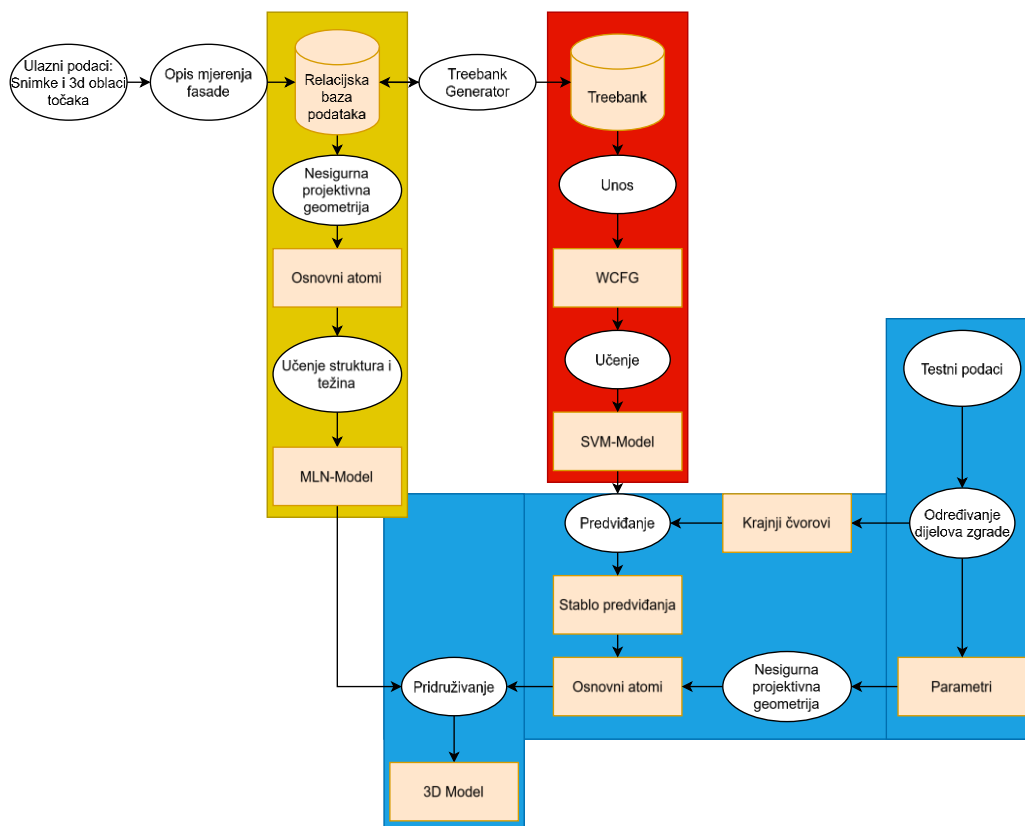
Na primjer, ako jedan dio formule F_i je $\text{sameWidth} \delta \text{Window1; Window2} \neq \text{sličan} \delta \text{Prozor1; Window2}$, onda se MRF-u dodaje potencijal $\exp \delta k_i f_k$. Značajka f_k definirana je nad svim osnovnim predikatima koji se pojavljuju u formuli i daje 1 ako je formula zadovoljena i 0 u suprotnom. U biti, MLN-ovi služe kao predložak za MRF-ove koji nam omogućuju jednostavno modeliranje neovisnosti na temelju logike pravila. To se može bolje objasniti i pomoću formula:

$$\varphi_i(x) = \exp(w_i \cdot f_i(x))$$

W_i - težina formule

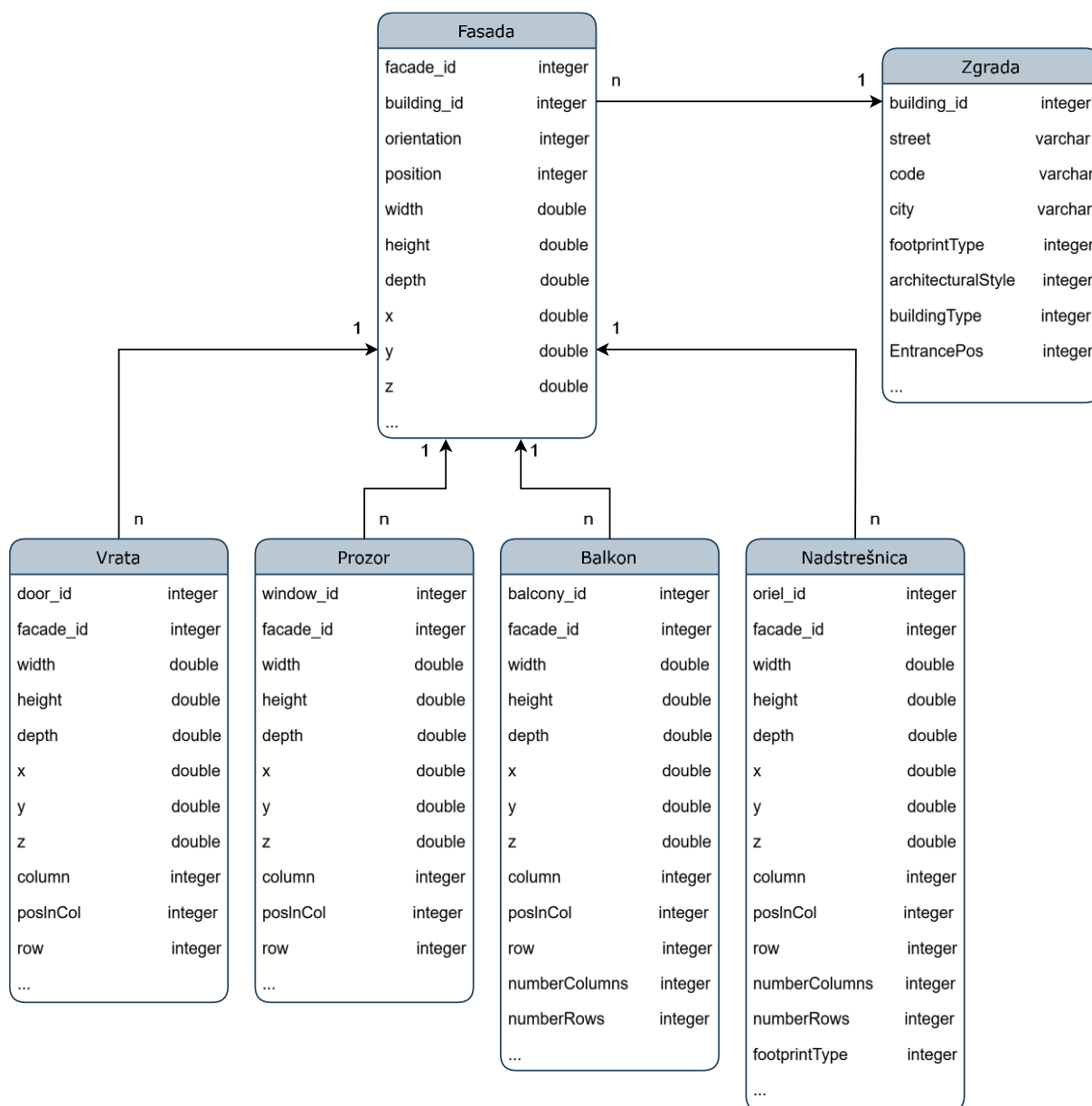
$f_i(x)$ - funkcija koja vraća 1 ako je formula zadovoljena, inače 0

Dijagram toka (Slika 9) prikazuje model strojnog učenja.



Slika 9. Dijagram toka WACFG modela strojnog učenja za fasade zgrada. Rezultati faze obuke su SVM-model (za klasifikaciju fasadne strukture-označeno crvenom) i MLN-model

(procjena parametara fasadnih dijelova-označeno žutom). Ti se koraci (označeno plavom) primjenjuju u fazi predviđanja kako bi se na kraju izveo model fasade.



Slika 10. Shema korištene relacijske baze podataka (RBDB).

Za interaktivno mjerenje slika kao i 3D oblaka točaka koristio sam open-source measureFacade (URL 7) alat. Građevni dijelovi od interesa su fasade, prozori, vrata, balkoni i strehe. Prepoznaju se oblici kao i lokacijski parametri svakog dijela zgrade. Za razliku od (Ripperda, 2008), korišteni su oblaci točaka kao i fotografije fasada i zadržana je relativna lokacija svakog dijela, npr. lokacija podova ili stupova. Slika 10 prikazuje shemu korištene baze podataka RBDB. Sve zgrade karakterizira njihov arhitektonski stil kao što je "poslijeratno doba" u

Njemačkoj ili wilhelminijski stil (Slika 11). Svaka zgrada sastoji se od jedne ili više fasada s njihovim međusobnim položajem u zgradi. Fasada se sastoji od nekoliko dijelova kao što su prozori ili erkeri. Svaki je dio povezan s određenim stupovima i podovima kako bi se opisale strukturne informacije. Svi se podaci uzimaju ručno ili iz izvornih, ispravljenih i skaliranih slika ili iz LiDAR 3D oblaka točaka, odnosno upotrebljena je baza od 70 zgrada koja je već snimljena LIDAR-om, URL 8). Kako bi se naučilo fasadne strukture, slijedimo nadzirani pristup učenja koristeći Support Vector Machines za strukturirane podatke. Ovo se izvodi na temelju ulazno-izlaznih parova. U fazi predviđanja, vidi dijagram toka (Slika 9, plava pozadina), za danu instancu fasade, predviđa se najvjerojatnije stablo koje predstavlja njen strukturni opis, taksonomiju i partonomiju. Ulaz je slijed znakovnih nizova, koji identificiraju tip (prozor, vrata, balkon i orijentir) dijelova fasade. Ulazna sekvenca se dobiva primjenom fasadnog objekta.

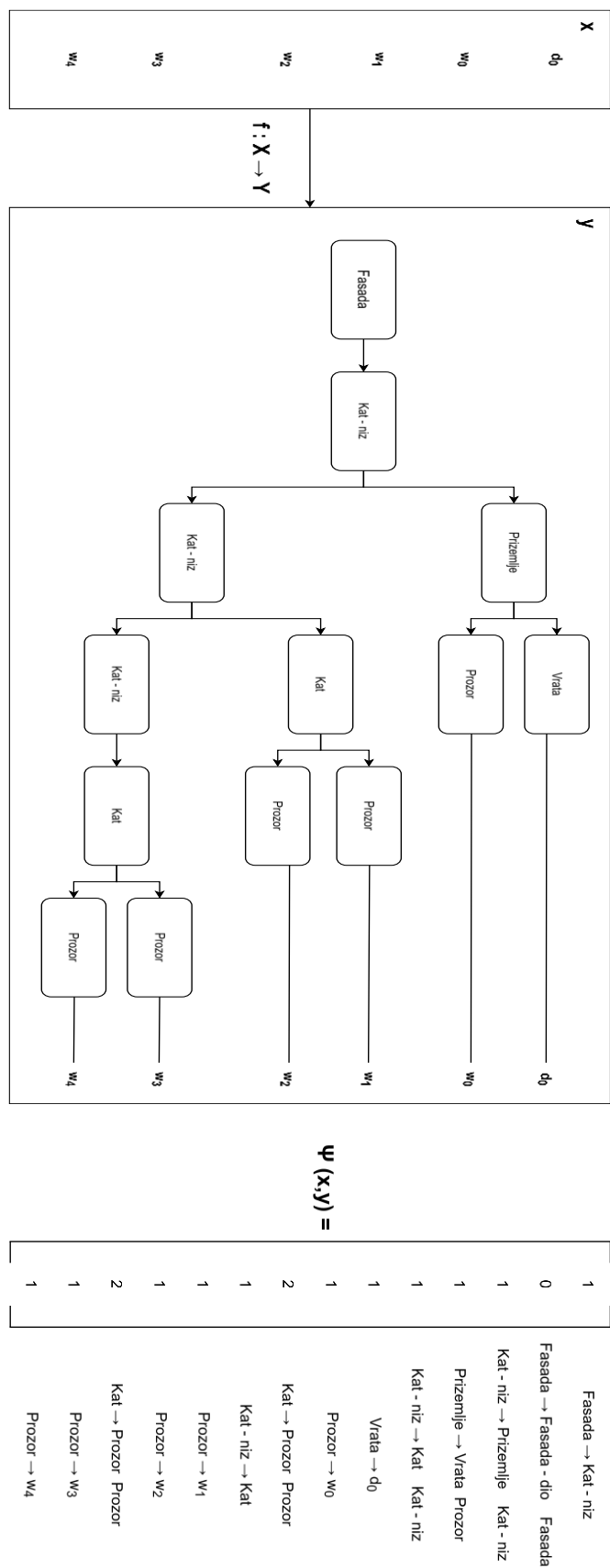


Slika 11. Poslijeratne fasade zgrada u Njemačkoj.

2.3.1. Strojno učenje težinski pripisane kontekstno-slobodne gramatike za 3D rekonstrukciju fasada

Shema nove metode za automatsko učenje težinski pripisane kontekstno-slobodne gramatike (WACFG) za analizu 3D oblaka točaka. Prikaz te metode (Slika 12) i proces učenja podijeljen je u faze: prvo se gradi struktura fasada, zatim se uče parametri i ograničenja za pojedine dijelove zgrada. U prvoj fazi razvija se WACFG koristeći relacijsku bazu podataka zgrada (RBDB) koja sadrži 533 zgrada iz Njemačke. Ove fasade pokrivaju različite stilove, čime se osigurava raznolikost i složenost modela. Za prikupljanje podataka o dimenzijama fasada upotrijebljen je alat *measureFacade* koji omogućava mjerenja na slikama i 3D oblacima točaka, a dimenzije su preuzete iz baze. Glavni dijelovi fasada uključuju prozore, vrata, balkone i erker-prozore. Ovi elementi su zabilježeni s detaljima o obliku i položaju. Za razliku od ranijih metoda, koje su se oslanjale samo na slike, ovaj pristup koristi i slike i oblake točaka kako bi

zadržao prostorne odnose dijelova po vrstama zgrada (npr. "obiteljske kuće"). Svaka zgrada ima jednu ili više fasada, a svaka fasada uključuje dijelove koji su raspoređeni prema stupcima i katovima kako bi se prikazala struktura.



Slika 12. Ilustrativni model analize primjera fasade. Izvedeno je izlazno stablo analize y, dana sekvenca dijelova građevine x pomoću vektora frekvencije pravila $W\ddot{o}x; y\dot{p}$.

Za kontrolu, neki su podaci prikupljeni ručno, korištenjem visokokvalitetnih slika ili LiDAR 3D oblaka točaka (zahvaljujući njemačkim geodetskim tvrtkama koje spominjem u zahvali).

Za strojno učenje strukture fasada koristim nadzirano učenje s potporno-vektorskim strojevima (SVM) prilagođenim strukturiranim podacima. Tijekom predviđanja (slika 3, plava pozadina), za svaku fasadu generira se najvjerojatnije parsersko stablo, koje prikazuje njezinu organizaciju. Ulazni podaci su nizovi elemenata fasade, poput prozora i vrata, koji se detektiraju pomoću metode jezgrenih procjena gustoće (KDE). KDE omogućava neparametarsku procjenu gustoće, što pomaže u određivanju oblika i položaja fasadnih elemenata. Za modeliranje fasadnih struktura koristim WACFG (ponderiranu atributnu kontekstno-slobodnu gramatiku), koja omogućava agregaciju dijelova fasade i prepoznavanje strukturnih obrazaca. Pravila gramatike dobivaju težine koje označavaju njihovu važnost. Za učenje gramatike kreiran je treebank (u nedostatku odgovarajuće riječi na hrvatskom) iz RBDB baze podataka. Kod fasada koje ne slijede pravilan mrežni uzorak, koristim hibridnu vizualizaciju, kombinirajući stupce (columnArray) i katove (floorArray).

Tablica 1. Popis korištenih predikata

sameFloor (x;y) true	ako i samo ako su x i y na istom katu
shareHorAlignment (x;y) true	ako i samo ako su x i y horizontalno poravnati
sameColumn (x;y) true	ako i samo ako x i y pripadaju istom stupcu
neighborHor (x;y) true	ako i samo ako su x i y horizontalni susjedi
shareVerAlignment (x;y) true	ako i samo ako su x i y vertikalno poravnati
neighborVer (x;y) true	ako i samo ako su x i y vertikalni susjedi
sameHeight (x;y) true	ako i samo ako su x i y vertikalni susjedi
sameHeight (x;y) true	ako i samo ako su x i y istu visinu.
sameWidth (x;y) true	ako i samo ako su x i y istu širinu.

Struktura fasade određuje se temeljem RBDB podataka, uz primjenu kriterija minimalne duljine opisa. Nakon generiranja stabala koristim SVM za derivaciju i parsiranje WACFG-a (Slika 9). Ova gramatika uključuje pravila s težinama, gdje svako pravilo ima određenu važnost. Za razliku od klasičnih SVM-ova, gdje vektori imaju fiksnu duljinu, ovdje vektori mogu biti različitih duljina jer predstavljaju struktuirana stabla parsiranja (Slika 12). Trenutno WACFG opisuje samo taksonomsku i strukturnu organizaciju fasada, uključujući topološke odnose između dijelova. Da bi se obuhvatilo dodatne faktore poput poravnjanja, geometrijske sličnosti i atributa (oblik, položaj), proširio sam gramatiku koristeći Markovljeve logičke mreže (MLN). Ovaj pristup omogućava rješavanje neizvjesnosti i nedostatka podataka. Za generiranje MLN-a koristimo podatke iz RBDB-a (Slika 9, svijetlosmeđa pozadina). Statistička geometrijska analiza temelji se na neizvjesnoj projektivnoj geometriji, što omogućava usporedbu geometrijskih elemenata fasade, primjerice prozora. Model uključuje testove poput provjere paralelnosti ili identiteta 3D točaka, pri čemu se uzima u obzir mogućnost pogrešaka u mjerenju. Tablica 1. prikazuje ključne predikate koji se koriste u modelu, uključujući horizontalno poravnanje, susjedstvo, visine i širine. Tijekom predviđanja, MLN procjenjuje geometrijske i topološke značajke fasade, omogućavajući rekonstrukciju 3D modela čak i kad su podaci djelomično nedostupni. Na kraju, rekonstrukcija fasade koristi naučene MLN i WACFG modele kako bi predvidjela najvjerojatniju geometriju i strukturu fasade.

2.3.2. Rezultati

Razvio sam tri različita modela SVM-a, prilagođena specifičnim stilovima zgrada.

- Prvi model temelji se na uzorcima poslijeratnih zgrada u Njemačkoj.
- Drugi model koristi Wilhelmski arhitektonski stil.
- Treći model primjenjuje se kada stil zgrade nije poznat.

Model za nepoznate stilove sadrži 450 pravila, Wilhelmski model ima 160 pravila, dok model za poslijeratne zgrade uključuje 135 pravila.

Tablica 2 prikazuje F1 rezultate dobivene tijekom učenja i testiranja pomoću softvera svmcfg (URL 9). Predviđanje strukture fasade temelji se samo na redoslijedu fasadnih elemenata (prozori, vrata itd.), bez dodatnih geometrijskih podataka ili informacija o katovima. Unatoč tome, F1 rezultat je uvijek najmanje 0,8.

Tijekom strojnog učenja koristio sam 90% podataka, dok je preostalih 10% korišteno za testiranje. Ovaj pristup omogućava i klasifikaciju zgrada nepoznatog stila, što je korisno u slučajevima kada tlocrti ili dodatni podaci nisu dostupni.

Također sam testirao MLN model na skupu podataka s 70 zgrada iz Njemačke. Svaka zgrada ima različit broj elemenata, zbog čega veličina MLN-a varira. Kako bi se provjerilo otpornost modela na nepotpune podatke, nasumično su uklonjeni dijelovi *sameWidth* i *sameHeight* predikata sa svake fasade.

Eksperimenti su provedeni metodom desetostruke unakrsne validacije – 90% zgrada korišteno je za učenje, a ostatak za testiranje. Rezultati su mjereni F1 rezultatom, koji je standardna metrika za procjenu točnosti modela.

Korišten je F1 rezultat kako bi izmjerili točnost predikcija pri rekonstrukciji fasada. Evaluacija se sastojala od:

1. Učenja modela – trenirali su različite modele gramatike koristeći nadzirano učenje.
2. Testiranja modela – model su testirali na novim podacima iz baze RBDB.
3. Izračuna F1 rezultata – usporedili su predviđene strukture fasada s stvarnim strukturama iz baze podataka.

F1 rezultat se računa pomoću formule:

$$F1 = 2 \times \frac{\text{Preciznost} \times \text{Odziv}}{\text{Preciznost} + \text{Odziv}}$$

Gdje su:

- Preciznost (Precision) = $\frac{TP}{TP+FP}$ (koliko su predikcije modela točne)
- Odziv (Recall) = $\frac{TP}{TP+FN}$ (koliko točnih primjera model uspješno prepozna)
- TP (True Positives) – ispravno prepoznati elementi fasade
- FP (False Positives) – pogrešno prepoznati elementi fasade
- FN (False Negatives) – elementi fasade koje model nije prepoznao

Korištena je *5-fold cross-validation* metoda, što znači da su se podaci podijelili u 5 dijelova, od kojih su 4 koristili za učenje, a 1 za testiranje i ponavlja se postupak 5 puta. F1 rezultat se smanjivao kad su podaci bili nepotpuni (npr. ako su neki elementi fasade bili zaklonjeni ili nisu

detektirani), najbolji F1 rezultat (~0.99) postignut je kada su svi podaci bili dostupni i kod 50% izgubljenih podataka, F1 rezultat je pao na 0.62, što pokazuje da model može raditi i s nepotpunim podacima, ali s manjom točnošću, što potvrđuje hipotezu.

Najprije se procjenjuju performanse ručno izrađenog MLN-a, gdje su samo težine naučene koristeći učenje Alchemy sustava. Time je omogućena usporedba s potpuno automatski naučenim MLN-ovima. Za početak se konstruira jednostavni MLN, gdje su sličnosti objekata zaključivane na temelju logičkih pravila. Na primjer, formula $\text{sameWidth}(x, y) \rightarrow \text{similar}(x, y)$ implicira da ista širina snažno (s većom težinom) sugerira sličnost. Također su dodana pravila koja izražavaju tranzitivnost među nepromatranim predikatima. Za sve zgrade provedena je MAP inferencija i usporedio sam predikcije. Prvi stupac u (Tablica 2) prikazuje prosječne rezultate za MaxWalkSat (MWS), dok drugi prikazuje rezultate dobivene pomoću Belief Propagation (BP). Kada nije bilo nedostatka podataka, problem je bio jednostavan, što rezultira F1 rezultatom od 0,977 za MWS. S povećanjem broja uklonjenih promatranja, F1 rezultati padaju, ali i dalje ostaju visoki, npr. 0,851 s 25% uklonjenih podataka.

BP rezultati su znatno lošiji (ispod 0,5) zbog problema s ručno postavljenim parametrima za pravila tranzicije. BP često ne funkcionira dobro s gotovo determinističkim težinama pravila. Eksperimentirano je i s uklanjanjem pravila tranzicije, što je dalo uravnoteženije rezultate, ali ukupna izvedba je bila s lošijim rezultatima.

Tablica 2. Za predviđene slične predikate na temelju MWS i BP, F1 rezultati.

	Ručno		Auto. naučeno		Poluautomatski	
	MWS	BP	MWS	BP	MWS	BP
0% podataka nedostaje	0,977	0,483	0,993	0,992	0,997	0,972
25% podataka nedostaje	0,851	0,466	0,841	0,876	0,789	0,877
50% podataka nedostaje	0,724	0,431	0,612	0,713	0,662	0,703

Zatim sam testirao automatski naučene MLN-ove putem sustava Alchemy. Treći i četvrti stupac (Tablica 2) prikazuju rezultate za MLN-ove s maksimalno dvije varijable po formuli. F1 rezultati su uravnoteženi za MWS i BP, pri čemu BP daje bolje rezultate u situacijama s većim

brojem nedostajućih podataka. Automatski naučeni MLN-ovi pokazali su se učinkovitima bez potrebe za ručnim unosom pravila.

Na primjer, jedno od naučenih pravila je $\neg \text{sameHeight}(a1, a2) \wedge \neg \text{sameWidth}(a1, a2) \rightarrow \neg \text{similar}(a1, a2)$. Iako ovo može izgledati intuitivno, model prepoznaje iznimke i dodjeljuje nisku težinu ovoj formuli, čime penalizira neprimjerene zaključke.

U posljednjem eksperimentu kombinirana su ručno izrađena pravila s automatski naučenim MLN-ovima, uključujući pravila za tranzitivnost, simetriju i refleksivnost. No, rezultati su pokazali nešto slabiju izvedbu. To znači da automatski naučeni modeli bolje rade samostalno, jer ručno dodana pravila mogu nepotrebno povećati složenost bez poboljšanja rezultata.

Može se zaključiti da automatski naučeni MLN modeli omogućuju učinkovito prepoznavanje obrazaca u fasadama bez potrebe za stručnim znanjem. Oni mogu samostalno učiti iz podataka, prepoznajući strukturne obrasce poput poravnavanja prozora ili položaja vrata, bez ručnog definiranja pravila od strane stručnjaka kao što su kartografi, geodeti, arhitekti ili građevinski inženjeri. Budući da su automatski naučeni MLN-ovi često manji, omogućuju bržu obradu podataka. U (dodatak A, URL 6) slika prikazuje sintaktičko stablo fasada s ukupno 5 rekonstruiranih fasada generiranih pomoću WACFG-a. Iako su prisutni šum, procjene gustoće i zaklonjeni dijelovi, vidi se da MLN pomaže ispraviti pogrešne parametre i nadopuniti nedostajuće dijelove fasade.

2.3.3. Diskusija i prijedlog za nastavak istraživanja

Predstavljena nova metoda koristi strojno učenje za učenje težinski pripisanih kontekstno-slobodnih pravila gramatike za 3D rekonstrukciju zgrada. Pravila služe kao moćan alat za unaprijed definiranje modela i pružanje semantičkih informacija koje pomažu procesu rekonstrukcije zgrada i njihovih dijelova. Ovaj pristup značajno smanjuje potrebu za ručnim projektiranjem gramatičkih pravila, što je često zahtjevan i dugotrajan zadatak. Kako bi se pojednostavilo i strukturiralo proces učenja, koristio sam dvostupanjsku strategiju. Prvo se razvija osnovna kontekstno-slobodna gramatika, a zatim su pravila proširena atributima i ograničenjima za dijelove zgrade, poput prozora i vrata. Ovaj prošireni model može opisati geometrijske i topološke odnose među elementima fasade. Za učenje gramatike koristio sam nadgledani pristup strojnog učenja temeljen na podršci SVM-a. Kao ulazne podatke koristio sam stabla koja predstavljaju strukture fasada i dobivena su iz relacijske baze podataka zgrada iz Njemačke. Ova baza sadrži širok raspon stilova zgrada, omogućujući modelu da se prilagodi

različitim strukturama. Proširenje osnovne gramatike atributima i pravilima ograničenja postignuto je korištenjem Markovljevih logičkih mreža (MLN). Ovo je prvi put, koliko mi je poznato, da su MLN-ovi primijenjeni u području 3D rekonstrukcije zgrada. MLN omogućavaju probabilističko zaključivanje koje pomaže pri donošenju odluka o geometriji i topologiji dijelova zgrada čak i u slučajevima kada su podaci nepotpuni ili neprecizni. Ispitivanja su pokazala su da predstavljeni model uspješno rješava probleme složenosti, nesigurnosti i nekompletnosti u stvarnim aplikacijama. Rezultati pokazuju visoku stopu točnosti u prepoznavanju struktura fasada, s F1 rezultatima između 0,8 i 0,84. Kada su svi podaci dostupni, točnost dostiže 0,99, dok je u slučaju gubitka 25% podataka točnost 0,83, a s 50% podataka točnost pada na 0,61. Svi navedeni rezultati su bolji od sličnih, prethodno spomenutih istraživanja. Osim toga, pokazano je da ručno kreirana pravila ne nadmašuju automatski naučena. Metoda omogućava automatsko generiranje pravila, težina, modela klasifikacije za strukture fasada i parametara MLN-a, čime se eliminira potreba za ručnim radom stručnjaka. Ova sposobnost učenja oslanja se na opsežnu bazu podataka koja pokriva različite stilove i strukture zgrada. Iako je takav pristup učinkovit, još uvijek ima prostora za napredak. Korištene su diskretne varijable u MLN-ovima, a svakako u nekim daljnjim istraživanjima treba upotrijebiti hibridne MLN-ove kako bi se ispitao rad s kontinuiranim varijablama, poput oblika i dimenzija dijelova zgrade. Hibridni MLN-ovi mogu integrirati distribucije poput multivarijatnih Gaussovih distribucija za bolje modeliranje. Osim toga, istražiti ćemo metode za iskorištavanje simetrije u strukturama fasada, što bi moglo dodatno smanjiti složenost problema. Simetrija se često pojavljuje u tlocrtima zgrada i može biti korisna za optimizaciju procesa rekonstrukcije. Primjena naprednih algoritama za zaključivanje, poput podignutog zaključivanja, također može povećati učinkovitost. I na kraju, prednost trenutnog modela u odnosu na slične je da podržava nerektonične (nepravokutne) oblike. Nerektonični oblici su geometrijski oblici koji nisu pravokutni, tj. čiji rubovi nisu ravne linije koje tvore pravokutne kutove od 90° (stupnjeva). To uključuje oblike s zakrivljenim rubovima, poput kružnih ili eliptičnih dijelova, kao i poligone s nepravilnim kutovima, poput trapeza, paralelograma, ili oblika s organskim konturama. U kontekstu 3D modeliranja i rekonstrukcije zgrada, podrška za nerektonične oblike znači mogućnost modeliranja fasada koje uključuju zakrivljene elemente (npr. kupole, zaobljeni prozori, lučni prolazi) ili kompleksne geometrije poput ukrasnih elemenata. Razlog zašto prethodni modeli koje možemo naći u dostupnoj literaturi nisu podržavali nerektonične oblike proizlazi iz same prirode korištene težinski pripisane kontekstno-slobodne gramatike (WACFG). Tada je gramatika razvijena za rad s pravokutnim strukturama, jer većina zgrada ima fasade sastavljene od pravilnih elemenata poput prozora,

vrata, ili ravnih zidova. Pravila gramatike i njihovi atributi stoga su definirani na način da podržavaju jednostavne pravokutne forme, gdje su granice dijelova opisane ravnim linijama i pravokutnim odnosima. Podrška za nerektonične oblike zahtijevala je proširenje modela kako bi uključio pravila koja uzimaju u obzir zakrivljenosti i kompleksne geometrijske odnose. To bi uključivalo prilagodbu metode za prepoznavanje i klasifikaciju oblika, kao i dodavanje dodatnih atributa i pravila za opisivanje zakrivljenih i nepravilnih elemenata koja su također uključena i prikazana u rezultatima. Tako da su uvedeni parametri za radijus zakrivljenosti, kuteve između linija koji nisu 90°, te topološka pravila za prepoznavanje kontura. Dakle, prvobitna hipoteza koja glasi “Automatizacija se može provesti samo do određenog stupnja, a koji se najvjerojatnije poklapa sa semantičkim podudaranjima modeliranja zgrada (odnosno podudaranja između dijelova oblikovanja ili dizajniranja koji su geometrijski i strukturno različiti), tako da samo jedna neusklađenost može narušiti rezultat, pa je potrebna ručna dorada” sada se može potvrditi i objasniti na sljedeći način:

1. Automatizacija i ograničenja:

- U istraživanju se predlaže automatsko generiranje pravila za rekonstrukciju fasada koristeći WACFG i MLN. Međutim, rezultati pokazuju da metoda može imati ograničenja u slučajevima kada su podaci nesigurni ili kada model nailazi na strukturalno složene oblike (npr. nerektonične oblike, fasade s nepredvidivim detaljima). To znači da automatizacija uspijeva do određene granice, a izvan te granice ručna intervencija može biti nužna kako bi se osigurala preciznost rezultata.

2. Utjecaj neusklađenosti:

- U istraživanju se ističe kako modeli poput MLN-a omogućavaju zaključivanje i rješavanje nesigurnosti, no i dalje postoji rizik od grešaka zbog neusklađenosti između semantičkih i geometrijskih karakteristika zgrada. Na primjer, ako fasada sadrži asimetrične ili složene uzorke koji nisu obuhvaćeni pravilima gramatike, automatska rekonstrukcija može generirati nepravilne ili nepotpune rezultate.

3. Potvrda hipoteze kroz eksperiment:

- Eksperimenti su pokazali da čak i s 25%-50% nedostajućih podataka, model može izvesti rekonstrukciju, ali s ograničenom preciznošću (F1 rezultat pada na

0,62 pri velikom gubitku podataka). Ovo implicira da se automatska metoda ne može u potpunosti osloniti na strogo definirana pravila kada su ulazni podaci nesavršeni.

- Prikazane fasade demonstriraju kako MLN može popuniti nedostajuće podatke ili ispraviti geometrijske pogreške, ali u slučajevima značajnih odstupanja od vizualno prepoznatljive fasade ili nepredviđenih obrazaca, ručna dorada je svakako potrebna kako bi se uklonile nepravilnosti.

Kako istraživanje potvrđuje hipotezu?

1. Potrebna ručna dorada kod složenih ili nesigurnih podataka:

- U istraživanju se zaključuje da trenutni modeli ne podržavaju nereketične oblike i asimetrične elemente, što znači da automatizacija ne može u potpunosti obuhvatiti sve semantičke i geometrijske razlike u modeliranju zgrada.
- Ručna intervencija bila bi potrebna za proširenje modela ili prilagodbu rezultata u takvim situacijama.

2. Prag automatizacije:

- Rezultati pokazuju da se automatizacija uspješno primjenjuje na pravokutne i pravilne fasade s dovoljno podataka. Međutim, kada se suoči s složenim dizajnom ili značajnim neskladom između ulaznih podataka i pravila, automatski proces daje nedovoljno dobre rezultate. Ovo potvrđuje hipotezu da postoji određeni prag nakon kojeg ručna intervencija postaje nužna.

3. Praktična primjena rezultata:

- Metoda pruža pouzdane rezultate za većinu urbanih fasada, ali kako bi se riješili specifični problemi nesemantičkog podudaranja (npr. geometrijska ili strukturna neusklađenost), sustav zahtijeva ručnu nadogradnju pravila ili korekciju rezultata.

Istraživanje jasno pokazuje da automatizacija može doseći samo određeni stupanj, posebno kod složenih oblika i nepotpunih podataka. Ovo podržava hipotezu da nesemantičke ili geometrijske neusklađenosti mogu zahtijevati ručnu doradu kako bi se osigurala točnost i funkcionalnost modela. Dakle, dok se metoda ističe u velikom broju slučajeva, stručna intervencija i ručna

dorada 3D modela ostaje ključna za osiguravanje kvalitete u rubnim slučajevima i jedino tako se osigurava identičan model kao što izgleda zgrada i u prirodi.

2.4. Skupovi pravila i automatizacija

Kako bih ispitao i treću hipotezu, razvijam novi pristup koji smatram prirodnim nastavkom dosadašnjih istraživanja u ovom području. U posljednjih nekoliko godina zabilježen je značajan napredak u tehnikama procedurnog modeliranja, pri čemu se kao glavna prednost ističe mogućnost generiranja velikih urbanih scena uz korištenje ograničenog skupa pravila. Na temelju toga predlažem uporabu generičkih tehnika prepisivanja grafova za transformaciju ulaznih skupova pravila u nove, čime se omogućuje njihova automatska ponovna uporaba, modifikacija i generiranje. Sustav koji razvijam temelji se na procedurnom jeziku, inspiriranom dobro poznatim CGA gramatikama, a primjenjuje se na modeliranje fasada.

Predlažem formalizam ponovnog pisanja skupa pravila koji dopušta automatsku izmjenu postojećih zgrada ili čak generiranje novih počevši od početnog skupa pravila, bez potrebe za daljnjom intervencijom korisnika. Na ovaj način, korisnik samo treba specificirati smjernice transformacije i sustav će ih automatski primijeniti na bilo koji skup pravila koji je dat kao unos, što rezultira novim skupom pravila spremnih za korištenje za generiranje novih zgrada.

Neki od autora ranije su se bavili tehnikama temeljenim na gramatici, kao što su pokazali (Wonka Michael; Sillion François; Ribarsky William, 2003) i kasnije koje su poboljšali (Müller Peter; Haegler Simon; Ulmer Andreas; Van Gool Luc, 2006). Ovi pristupi dopuštaju samo neizravnu kontrolu nad konačnim modelom promjenom tekstualne reprezentacije temeljne gramatike i pružaju interaktivnu kontrolu samo promjenom nekih konstruktivnih parametara na konačnom modelu. Također primjećujem da se kako CGA gramatika, koju su detaljno opisali Müller i suradnici (2006) (Müller Peter; Haegler Simon; Ulmer Andreas; Van Gool Luc, 2006), tako i sustav koji razvijam u ovom radu, mogu implementirati unutar okvira GML-a (Generative Modeling Language). Ova mogućnost implementacije ukazuje na tehničku kompatibilnost i fleksibilnost pristupa, što dodatno potvrđuje potencijal za proširenje postojećih modela procedure generacije unutar standardiziranih alata. i sustav koji opisujemo u ovom radu može se implementirati na GML. Kasnije su (Lipp Peter; Wimmer Michael, 2008) predstavili su paradigmu interaktivnog vizualnog uređivanja u stvarnom vremenu za gramatike oblika, koja omogućuje kreiranje skupova pravila od početka bez potrebe za ručnim pisanjem ili izmjenom tekstualnih datoteka.. U novije vrijeme, (Patow, 2012), Esrijev CityEngine i Epic Games' UDK

predstavili su vizualni jezik za rad s oblicima gramatičkog modeliranja, u kojem je svaki čvor vizualne reprezentacije jedna naredba ili operacija, dok veza između dva čvora označava smjer i redoslijed protoka geometrijske obrade – tj. tok transformacija koje se primjenjuju na ulazne podatke. Oblik (u ovom kontekstu) – odnosi se na geometrijski ili logički entitet unutar procedurnog modela, poput fasade, kata, prozora itd., koji se opisuje i transformira pomoću pravila. Tok – predstavlja smjer protoka podataka ili transformacija između tih oblika. U vizualnom jeziku to izgleda kao poveznica između čvorova (npr. strelica od naredbe "Split" prema naredbi "Insert") i označava kojim redom se operacije izvode. (Krecklau Darko; Kobbelt Leif, 2010) je proširio način upotrebe neterminalnih simbola u procedurnom modeliranju tako da se oni mogu koristiti kao ulazni parametri drugim simbolima, na način sličan funkcijama u programskim jezicima. Iako ovaj pristup omogućuje veliku fleksibilnost i izražajnost, glavni nedostatak je što zahtijeva poznavanje programiranja kako bi se učinkovito koristio i upravljalo njegovom složenosti.

Neterminalni simboli su elementi gramatike koji ne predstavljaju konkretne dijelove konačnog modela (npr. prozor, vrata, kat), već služe kao "pravila" ili "kontejneri" koji se dalje razrađuju u druge simbole – terminalne ili druge neterminalne. Oni opisuju strukturu ili logiku modela, a ne njegovu završnu formu. U kontekstu procedurnog modeliranja, neterminalni simbol može označavati npr. zgradu, koja se dalje može razložiti na katove, a svaki kat može sadržavati prozore i vrata – pri čemu su prozori i vrata već konkretni ili tzv. terminalni elementi (Slika 13).

Ako zamislimo jednostavno pravilo zapisano u CGA stilu:

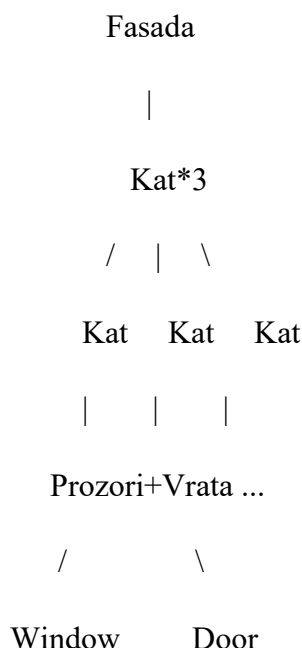
Fasada --> Kat*3

Kat --> Prozori + Vrata

Prozori --> Window("modern")

Vrata --> Door("classic")

A uz vizualizaciju hijerarhije:



Slika 13. Stablo gramatike. Svaki kat se dalje razlaže na prozore i vrata, a ti se simboli na kraju pretvaraju u konkretne objekte.

Jedan od glavnih nedostataka ranijih pristupa je što zahtijevaju znanje programiranja kako bi se korisnici mogli nositi sa složenošću njihove primjene. (Krecklau Leif, 2012) predstavili su napredne građevne blokove, tzv. primitive visoke razine, koji uključuju parametre – vrijednosti koje se mogu mijenjati i omogućuju interaktivno uređivanje modela pomoću vizualnih alata (manipulatora) koji su ugrađeni u sam model. S druge strane, (LinJinjie ; ZhangHao ; LiangCheng ; SharfAndrei ; DeussenOliver ; ChenBaoquan, 2011), razvili su tehniku koja omogućuje prilagodbu nepravilnih 3D arhitektonskih oblika novim zadanim ciljevima, a pritom se zadržava izvorni vizualni stil objekta. Ta metoda ima suprotan cilj u odnosu na pristup prikazan u ovom radu: dok njihov sustav prenosi postojeći stil s jedne zgrade na drugu, moj pristup omogućuje promjenu samog stila gradnje. (Schwarz Pascal, 2015) predložio je koncept proširenih kontekstualnih upita, u kojem se tzv. primitivci (osnovni građevni elementi) tretiraju kao „građevine ili elementi prve klase“, odnosno kao ravnopravni nositelji značenja unutar sustava. Nekoliko novijih istraživanja usmjereno je na pojednostavljivanje načina na koji korisnici interaktivno oblikuju zgrade. To se postiže tako da se općenito definira način na koji korisnik odabire i mijenja elemente na fasadi. (Nishida Ignacio; Aliaga Daniel G.; Benes Bedrich; Bousseau Adrien, 2016) predstavili su tehniku za izradu urbanih modela korištenjem jednostavnog interaktivnog skiciranja, gdje se modeli automatski generiraju prema

procedurnim pravilima (Jesus Gustavo; Coelho António; Sousa A. Augusto, 2018) i (Jiang Dong-Ming; Zhang Xiaopeng; Wonka Peter, 2018) razvili su metode za „generalizirane odabire“, gdje se složeniji elementi mogu birati i uređivati putem upita i primjera – bez potrebe za ručnim kodiranjem pravila. Time se omogućuje definiranje cijelih skupova pravila unutar sučelja koje je razumljivo i vizualno orijentirano. (Lipp Matthias; Lau Cheryl; Wonka Peter; Müller Pascal, 2019) predstavili su sustav koji omogućuje dodavanje čitavog sloja lokalnih izmjena na procedurno generirane zgrade. Na taj način moguće je brzo i fleksibilno proizvesti velik broj detaljno razrađenih nacрта zgrada. (Ritchie Ben; Goodman Noah D.; Hanrahan Pat, 2015) dizajnirali su pristup koji omogućuje korisniku da kontrolira rezultate visoko varijabilnih procedurnih modela korištenjem tzv. stohastički uređenih sekvencijalnih Monte Carlo algoritama. Ovaj pristup omogućuje da se, unatoč nasumično generiranim varijacijama, postigne rezultat koji odgovara ciljanom obliku – primjerice, po volumetriji. (Lienhard Cheryl; Müller Pascal; Wonka Peter; Pauly Mark, 2017) predložili su pristup u kojem se dva procedurna modela povezuju kroz niz međukoraka. Umjesto izravnog spajanja geometrije – kao kod klasičnih mesh-algoritama (Botsch Leif; Pauly Mark; Alliez Pierre; Levy Bruno, 2010) kombiniraju generativna pravila (stabla oblika), što omogućuje postupnu transformaciju jednog modela u drugi. Za razliku od navedenih metoda, moj cilj je razviti pristup koji omogućuje transformaciju postojećih procedurnih modela zgrada, uz kontrolu korisnika i uz očuvanje gramatičke strukture modela. Korisnik definira pravila transformacije koja se primjenjuju na ulazni model. Rezultat je novi model, koji se i dalje temelji na pravilima, ali s jasno određenim i željenim promjenama.

U ovom poglavlju predlažem primjenu tehnika prepisivanja grafova nad skupovima pravila, s ciljem automatske izmjene stilova procedurnih modela zgrada. Za razliku od pristupa opisanog u (Patow, 2012), gdje su bile moguće samo jednostavne operacije, moj prijedlog omogućuje izvođenje i složenijih promjena bez potrebe za dodatnom intervencijom korisnika – osim u fazi definiranja početnih pravila za prepisivanje.

Ovaj pristup omogućuje potpuno automatsku transformaciju stila postojeće zgrade u novi stil i to kroz modificiranje njezinih gramatičkih pravila pomoću strukture grafa.

U tehničkom smislu, svaka takva transformacija može se prikazati kao graf u kojem:

- čvorovi predstavljaju pojedine operacije (ili skupine operacija) koje se primjenjuju na model, a

- rubovi (poveznice između čvorova) opisuju tijek geometrije, odnosno način na koji se podaci prenose i obrađuju.

Korištenjem takvih grafova moguće je strukturirano mijenjati pravila generiranja zgrade, što za rezultat ima potpuno novi vizualni i funkcionalni izgled modela. Na primjer, jednostavna zgrada (Slika 14), može se definirati pomoću niza pravila kako slijedi:

Lot Extrude(height) : InitialShape

InitialShape Comp(f){front : FrFacade | strana : fasada}

FrFacade Split(y){height1st : FirstFFr | ? 1 : Fasada}

Ponavljanje fasade (y){? visinaKat : Kat}

Ponavljanje poda (x){? širina prozora : Prozor}

Umetanje prozora (“modernWindow.obj”)

FirstFFr Insert(“door.obj”)

i grafički prikazano na (Slika 14). Ovdje su korištene neke unaprijed definirane konstante, kao što je visina1 za visinu prvog kata, fheight za visinu ostalih katova i windowwidth za širinu prozora.



Slika 14. Početna jednostavna zgrada za geovizualizaciju i početak transformacije u druge stilove.

Transformacije grafova temelje se na primjeni posebnih pravila, koja se u stručnoj literaturi nazivaju produkcijama (Rozenberg, 1997). Svako pravilo zamjenjuje određeni dio postojećeg grafa s novim, unaprijed definiranim dijelom.

Pravilo transformacije može se zapisati kao:

$r = (p, L, R, \text{ljepilo})$

gdje:

- p je uvjet koji određuje treba li se pravilo primijeniti,
- L je izvorni (lijevi) dio grafa koji se traži,
- R je novi (desni) dio koji se umeće umjesto L ,
- ljepilo definira način povezivanja novog dijela s ostatkom grafa.

Ako je uvjet p ispunjen, sustav će ukloniti dio L iz glavnog grafa G , te umetnuti R i povezati ga s ostatkom grafa pomoću "ljepila". To uključuje spajanje čvorova i dodavanje veza (rubova) između postojećih i novih dijelova.

To izgleda ovako:

$p(L)? \text{ljepilo}(L, R)$

– što znači: ako uvjet $p(L)$ vrijedi, primijeni povezivanje L i R prema definiranom mehanizmu.

Uvjet p vraća jednostavnu istinitu vrijednost (True/False), a ljepilo uključuje skup naredbi, tzv. upute za prepisivanje, koje se izvršavaju redom, odvojene točka-zarezom.

Važno je napomenuti da će se pravilo automatski primijeniti na sve dijelove grafa G koji odgovaraju L .

Na primjer, ako je uvjet:

$(n.type = \text{Split})$

$i (n \rightarrow m)$

$uz (m.type = \text{Insert})$

– sustav će pronaći sve čvorove n tipa „Split“ koji imaju vezu prema čvorovima m tipa „Insert“. Te se veze zatim označavaju i pripremaju za zamjenu.

Uvodim i dvije glavne vrste uvjeta:

1. Parametarski uvjeti – npr. $n.parm = vrijednost$ – koji uspoređuju vrijednosti parametara čvorova (uz podršku operatora kao što su $<$, $>$, $!=$, itd.).
2. Uvjeti veze – npr. $n \rightarrow m$ – koji provjeravaju postoji li veza između čvorova. Također postoji skraćenica poput $n \rightarrow f$ vrijednost m , što znači: čvor n je povezan s čvorom m , a m ima svojstvo $filter = vrijednost$. Suprotno tome, izraz $n \nrightarrow m$ označava da čvor n nije povezan s m .

Nakon definiranja uvjeta, sustav može izvršiti ponovno spajanje čvorova. To uključuje umetanje novih čvorova odmah nakon nekog postojećeg čvora n , zadržavanje njegovih ulaznih veza i preusmjeravanje izlaznih veza prema novoj konfiguraciji (npr. pomoću uvjeta poput: $p(n \rightarrow_s b) \Rightarrow n \rightarrow_s a \rightarrow_s b$).

Sustav dodatno omogućuje korištenje Python funkcija – korisnik može pozvati funkciju iz vanjske biblioteke i predati joj parametre.

Na primjer:

```
n.size = compNewProportions(n.size)
```

– gdje je `compNewProportions` Python funkcija koja prima trenutne dimenzije i vraća nove u skladu s drugačijim stilom. Ta se funkcija automatski učitava i izvršava u pozadini.

Kako bi se pravilo primijenilo, prvo moramo pronaći sve podgrafove L u glavnom grafu G koji zadovoljavaju uvjet p . To rezultira skupom kandidata $\{L_i\}$, na koje se pravilo zatim primjenjuje. Pravila se ne ponavljaju na istom dijelu grafa, ali se mogu primjenjivati u proizvoljnom redoslijedu – sve dok se ne preklapaju.

Nakon što se jedno pravilo primijeni, označeni čvorovi se „zaključaju“ za to pravilo. Kad se krene s drugim pravilom, ta se ograničenja poništavaju.

Sustav prvo pronalazi sve čvorove tipa “Repeat”. Zatim provjerava imaju li vezu prema čvoru b u smjeru istoka. Ako obje komponente uvjeta vrijede, izdvajaju se svi parovi čvorova koji ih zadovoljavaju. Time se graf može postupno mijenjati prema zadanim pravilima, uz strogo definiran mehanizam.

U sljedećem primjeru je pravilo koje transformira početnu fasadu u novu. Svako se pravilo primjenjuje samo jednom, ali se odnosi na sve dijelove grafa koji zadovoljavaju uvjet.

```
( n.product = vrata ) ? $\Rightarrow$ 
```

```
a = new ( Insert )
```

a.object = “ ulaznavrata.obj”

$n \rightarrow a$

U pravilu prepisivanja grafova, svaka jednostavna naredba odgovorna je za pretvorbu jednog dijela grafa L_i u njegov odgovarajući novi oblik R_i . Tijekom tog procesa, sustav ponovno stvara skup povezanih parova čvorova označenih kao $\{(n_i, b_i)\}$, ali sada dodatno filtrira rezultate – izbacuju se svi čvorovi n_i koji nisu tipa “Repeat”.

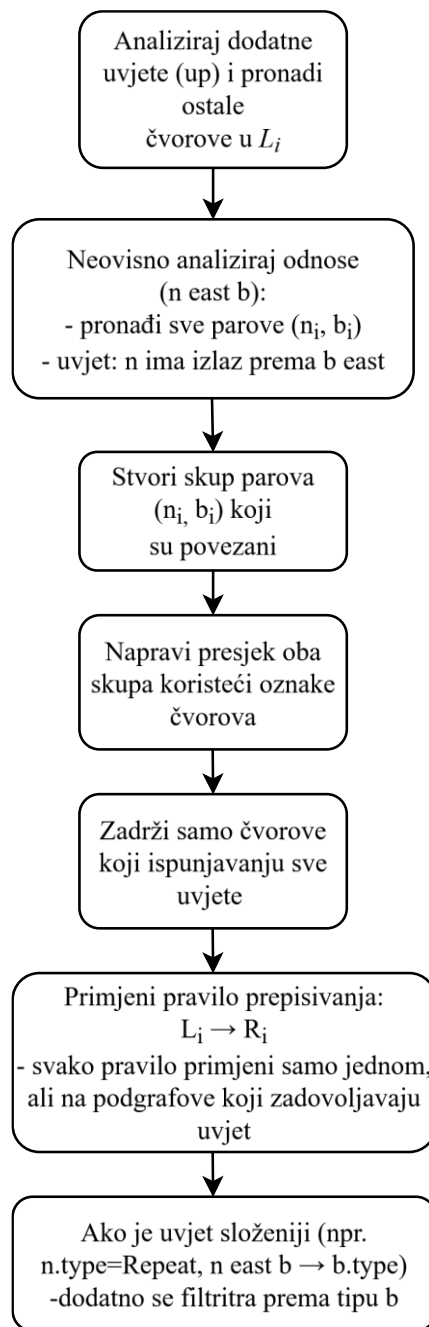
Može se napraviti i da sustav ne traži samo da čvor n bude tipa “Repeat” i povezan s čvorom b u smjeru istoka (east), nego i da čvor b bude tipa “Split”.

Ako se umjesto operatora I koristi ILI , tada se pravilo primjenjuje na sve čvorove koji zadovoljavaju bilo koji od navedenih uvjeta, a ne sve istovremeno. Drugim riječima, u tom se slučaju pravi unija skupova, a ne presjek kao kod operatora I (&).

Cijeli postupak može se dodatno vizualizirati dijagramom (Slika 15).

Nakon što sustav prepozna sve čvorove u grafu koji odgovaraju lijevom dijelu nekog pravila preoblikovanja (tzv. L_i), dobiva se skup parova ili višestrukih skupova čvorova – zapisanih kao torke (npr. $\{(n_i, m_i, \dots, z_i)\}$). Svaka takva torka predstavlja jedan primjer (ili pojavu) tog uzorka L_i unutar glavnog grafa G , tj. točno određenu kombinaciju čvorova koja odgovara uvjetima iz pravila. Drugim riječima, to su svi dijelovi grafa koji odgovaraju obrascu definiranom u pravilu.

Svaki čvor unutar torki povezan je s odgovarajućim elementima definiranim u pravilu i koristi se u sljedećem koraku – tijekom primjene pravila. Tada ti čvorovi služe kao točke povezivanja za novu strukturu s desne strane pravila (R_i). Drugim riječima, oni omogućuju “ljepljenje” novih dijelova grafa na postojeću mrežu, bilo dodavanjem novih čvorova, bilo stvaranjem novih veza između starih i novih elemenata.



Slika 15. Shema koja prikazuje cijeli proces prepisivanja grafa korak po korak.

U sljedećem dijelu predstavljeni su rezultati, odnosno primjene skupa pravila na složeniji primjer.

Ovdje su korištene i dvije dodatne funkcije (napisane u Pythonu):

- `getenv` – standardna funkcija za dohvat varijable iz okruženja.

- `graphutils.get_style` – funkcija iz pomoćne biblioteke koja vraća stil određenog elementa grafa.

Na primjer, operacija `InitQuad(height)` koristi se za izradu jednostavne vertikalne fasade – što je početni oblik.

2.4.1. Transformacija stila

U ovom koraku, svi parametri se redom obrađuju i primjenjuju se potrebne promjene. Ako treba, čvorovi se ponovno povezuju ili spajaju.

Kad se obradi cijeli podgraf (tj. svi čvorovi koji su odgovarali obrascu L), pravilo završava s izvršavanjem.

Ako više nema ponavljanja obrasca L u grafu, to znači da je pravilo u potpunosti završilo svoju primjenu.

U ovom dijelu prikazujem kako se pravila prepisivanja mogu koristiti za preoblikovanje fasade. Pravila koja koristimo slična su onima ranije opisanim, ali sada ih primjenjujemo u složenijoj situaciji.

Koristimo pravila definirana u jeziku koji omogućuje prepoznavanje određenih uzoraka u grafu i njihovu zamjenu novim strukturama. Svako pravilo opisuje što se traži u grafu (tzv. lijeva strana pravila, L) i što se s tim dijelom grafa treba napraviti (desna strana, R).

Na primjer, pravilom se može reći:

„Pronađi čvor koji je tipa *Repeat* i koji ima vezu prema čvoru tipa *Split*, i zatim to zamijeni određenim novim elementima.“

U ovom primjeru također se koriste dvije vanjske funkcije (napisane u Pythonu):

- `getenv` – funkcija koja dohvaća vrijednosti iz okruženja (npr. visinu ili širinu elemenata).
- `graphutils.get_style` – funkcija koja određuje stil elemenata u grafu (npr. boju, debljinu linije i sl.).

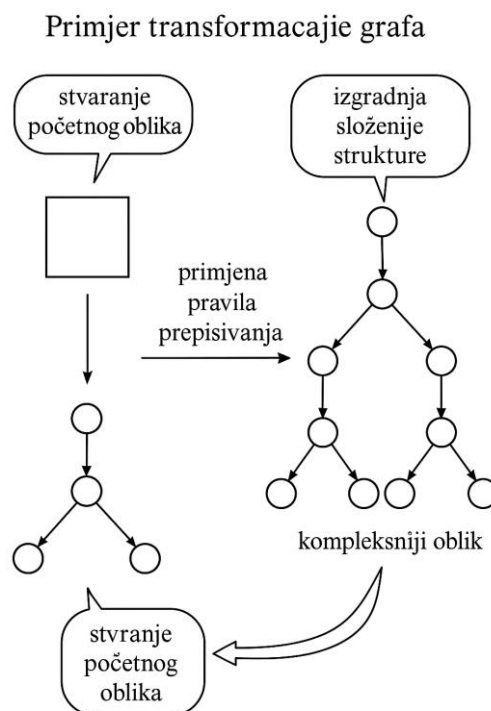
Te funkcije omogućuju dodatnu fleksibilnost, jer pravila mogu reagirati na dinamičke podatke iz okruženja. Proces transformacije počinje stvaranjem osnovne fasade pomoću naredbe poput `InitQuad(height)`. Ta naredba stvara početni grafički oblik – jednostavnu uspravnu fasadu,

koja se može vidjeti na (Slika 17, lijeva strana). Primjenom pravila R1 transformira se graf tako da se dodaju novi elementi ili zamijene postojeći.

U ovom konkretnom primjeru pojavljuju se i:

- rekurzivne primjene – pravilo se primjenjuje više puta, dok god postoje dijelovi grafa koji mu odgovaraju,
- kompleksne situacije – gdje se, npr., više čvorova mora povezati na specifičan način ili postoje posebni uvjeti za primjenu pravila.

Takve transformacije omogućuju izgradnju sve kompleksnijih grafova koji predstavljaju novu verziju fasade (Slika 16 i Slika 17).



Slika 16. Dijagram koji objašnjava svaki korak transformacije grafa – od početne fasade, preko primjene pravila, do složenije strukture.



Slika 17. Transformacija na realističnoj fasadi. Primjećuje se kako se različita pravila mogu kombinirati bilo kojim redoslijedom i kako se mogu koristiti s različitim stilovima (gornji red: ciglena fasada, donji red: betonska fasada).

Prikazan je primjer prepisivanja grafa koji demonstrira kako se definiran skup pravila može primijeniti na postojeću strukturu kako bi se izgradio kompleksniji arhitektonski prikaz fasade. Nakon što se lociraju svi čvorovi koji se nalaze u lijevom dijelu pravila transformacije, dobivamo skupove torki oblika $\{(n_i, m_i, \dots, z_i)\} \setminus \{(n_i, m_i, \dots, z_i)\} \setminus \{(n_i, m_i, \dots, z_i)\}$. Svaka torka predstavlja jednu instancu lijeve strane pravila $L_i L_i L_i$ pronađenu unutar grafa GGG.

Ti čvorovi u svakoj točki odgovaraju označenim elementima unutar svakog pravila, te će se koristiti kao osnova za izvođenje naredbi lijepljenja definiranim u desnoj strani pravila $R_i R_i R_i$. Drugim riječima, oni omogućuju identifikaciju preciznih pozicija u grafu gdje će se izvršiti promjene.

U ovom primjeru koristimo dvije vanjske funkcije, koje su implementirane u Pythonu. Prva je standardna funkcija `getenv`, koja dohvaća varijablu iz okruženja. Druga je `graphutils.get_style`, funkcija koja je dio postojećeg pomoćnog modula za stilizaciju grafičkih elemenata. Ove funkcije omogućuju da pravila ne budu statična, već da u obzir uzimaju parametre iz konteksta izvođenja.

Početna fasada se stvara operacijom `InitQuad(height)`, koja generira jednostavnu uspravnu geometriju fasade i predstavlja osnovu daljnje konstrukcije. Taj se oblik može vidjeti na početku primjera i upravo on služi kao početni graf za prepisivanje.

U nastavku, transformacija R1 (koja se ranije prikazivala slikovno) uključuje primjenu pravila koje uvodi nove elemente i odnose među njima, ovisno o obrascu prepoznatom u grafu. Ova transformacija može uključivati i rekurzivne aplikacije — što znači da se pravilo može primijeniti na dijelove grafa koji su već jednom transformirani, ukoliko ponovno zadovolje uvjete obrasca LLL. Time se otvara mogućnost stvaranja hijerarhijskih i kompleksnijih struktura. Također se spominju i „čudne situacije“ koje se mogu pojaviti, što upućuje na nelinearno ponašanje grafa prilikom ponovljenih transformacija.

Treba razmotriti na koji način pravila ne mijenjaju samo strukturu grafa, već i njegov stil. Taj se proces naziva **transformacija stila**. Prilikom izvođenja pravila, redom se procjenjuju svi parametri, uključujući pozive funkcija poput *getenv* ili vlastitih izraza definiranih u pravilima. Nakon što se svi parametri evaluiraju, izvršavaju se promjene: bilo da se mijenjaju atributi postojećih čvorova, ili da se oni ponovno povezuju (eng. „reattached“) na drugi način unutar grafa. Nakon što se evaluacija ulaza završi, pravilo je završilo svoje djelovanje nad trenutnim podgrafom. Ako se uvjeti grafa LLL više ne pojavljuju unutar grafa, tada se može smatrati da je pravilo u potpunosti izvršeno.

2.4.2. Izmjena stilskih pravila

Ovaj dio objašnjava kako sustav ne omogućuje samo prepoznavanje i transformaciju strukture grafa, nego i dinamičku promjenu stila prikaza elemenata. Na primjer, pravilo može mijenjati vizualna svojstva poput boje ili debljine linije na temelju definiranih parametara.

Primjer izraza:

```
style = graphutils.get_style(n.height)
```

U ovom slučaju funkcija *get_style* koristi visinu čvora *n* kako bi odredila odgovarajući stil prikaza. Takvi stilovi mogu se primijeniti na različite čvorove u grafu, čime se vizualno razlikuju, primjerice, katovi u zgradi ili različiti arhitektonski elementi.

Ako se pravilo primijeni na čvor koji već ima definiran stil, sustav omogućuje:

- nadogradnju stila (dodavanje novih karakteristika),
- zamjenu postojećeg stila,

- ili interpolaciju (kombiniranje starog i novog stila).

Također, sustav podržava nasljeđivanje stilova s roditeljskih čvorova, čime se osigurava konzistentno stiliziranje složenijih dijelova grafa bez dodatnog ručnog podešavanja.

Ovdje se ističe važnost funkcija i izraza koji dodatno proširuju mogućnosti prepisivanja grafa. Pravila ne moraju biti statična; mogu reagirati na dinamične parametre, poput ulaznih podataka korisnika ili promjenjivih uvjeta u okruženju. Primjer uvjeta ovisnog o okruženju:

```
if getenv("ENV_VAR") = "urban":
```

Ovakav uvjet omogućuje da se pravila ponašaju drugačije ovisno o trenutačnom kontekstu izvršavanja (npr. razlikovanje urbanih i ruralnih područja). Funkcije unutar sustava mogu uključivati:

- matematičke operacije,
- pozive drugim funkcijama,
- logičke izraze.

Na taj način, ista pravila mogu generirati različite rezultate ovisno o trenutnim uvjetima, čime sustav dobiva visoku razinu prilagodljivosti.

Ovo poglavlje reflektira cjelokupni pristup i naglašava da prepisivanje grafova predstavlja značajan potencijal za automatizaciju i kreativnu kontrolu u generiranju arhitektonskih struktura. Predloženi sustav omogućuje:

- precizno procedurno modeliranje,
- fleksibilnu upotrebu pravila, izraza i stilova,
- te automatizirano upravljanje velikim skupovima podataka.

Navedene metode mogu se primijeniti na stvarne projekte, poput generiranja fasada za cijele blokove zgrada. Osim toga, postoje mogućnosti integracije s vanjskim alatima (npr. BIM sustavi, sustavi za simulaciju osvjetljenja), pri čemu bi dobiveni grafovi mogli biti izvezeni i

daljnje obrađeni u specijaliziranim programima. To bi mogla biti nova tema ili nastavak istraživanja.

Predstavljen je novi pristup procedurnom modeliranju struktura zgrada, temeljen na tehnici prepisivanja grafova. Predloženi sustav omogućuje integraciju geometrije, stilskih pravila i logičkih odnosa unutar jedinstvenog, fleksibilnog okvira. Ključna prednost sustava je njegova proširivost: pravila su definirana jasno i precizno, ali ostavljaju dovoljno slobode za prilagodbu različitim vrstama dizajna – ne samo za modeliranje fasada, nego i za širi spektar arhitektonskih i urbanih struktura. Sustav koristi generičke tehnike prepisivanja grafova za automatsku transformaciju skupa pravila, čime omogućuje ponovnu upotrebu, nadogradnju i generiranje novih pravila bez potrebe za ručnom intervencijom. Time se značajno smanjuje kompleksnost modeliranja, osobito u slučajevima kada je potrebno primijeniti identične izmjene na velikom broju objekata. Praktičnost i primjena sustava demonstrirane su u kontekstu procedurnog jezika visoke razine temeljenog na CGA gramatikama, s posebnim naglaskom na automatsku transformaciju stilova fasada. Vizualizacija rezultata pokazuje da sustav omogućuje transformacije koje su automatske, transparentne za krajnjeg korisnika i ne zahtijevaju dodatne napore u modeliranju. Ovaj rad time predstavlja novu fazu i ideju unutar procedurnog modeliranja, otvarajući već spomenute mogućnosti za buduća istraživanja. Potencijalna proširenja uključuju podršku za više tipova veza između čvorova, hijerarhijske definicije pravila, te učenje pravila iz primjera korištenjem strojnih algoritama. Konačno, sustav postavlja temelje za razvoj inteligentnih, prilagodljivih alata za dizajn i 3D modeliranje zgrada ubuduće, kao i njihove geovizualizacije u kartografiji.

2.4.3. Gotovi primjeri primjene i evaluacija s rezultatima

Za primjer transformacije stilova, uzeo sam centar Münchena (München) koji odražava bogatu povijest kroz različite arhitektonske stilove, od srednjovjekovne gotike do moderne arhitekture. Evo glavnih stilova koje možete pronaći u Altstadtu (Starom gradu) i okolici:

1. Gotička arhitektura (1100. – 1500.) Npr. Frauenkirche (Katedrala Blažene Djevice Marije) – Kasnogotička crkva s prepoznatljivim dvostrukim tornjevima i crvenom ciglenom fasadom. Altes Rathaus (Stara gradska vijećnica) – Prekrasna gotička zgrada s visokim tornjem i šiljastim lukovima.
2. Barok i rokoko (1600. – 1700.) Npr. Theatinerkirche (Crkva sv. Kajetana) – Talijanski barok, s upečatljivom žutom fasadom i bogato ukrašenom unutrašnjosti, Asamkirche (Asamova crkva) – Jedna od najspektakularnijih rokoko crkava u Njemačkoj, puna zlatnih detalja i fresaka.

3. Klasicizam i neoklasicizam (1700. – 1800.). Npr. Glyptothek & Alte Pinakothek – Muzejske zgrade inspirirane grčko-rimskom arhitekturom. Königsplatz – Trg s impozantnim neoklasičnim građevinama, oblikovan po uzoru na antičku Atenu.

4. Historicizam i neogotika (1800. – 1900.). Npr. Neues Rathaus (Nova gradska vijećnica) – Neogotička građevina s bogato ukrašenom fasadom i poznatim Glockenspielom (satni mehanizam s figurama). Justizpalast (Palača pravde) – Impozantna neobarokna zgrada s velikom kupolom.

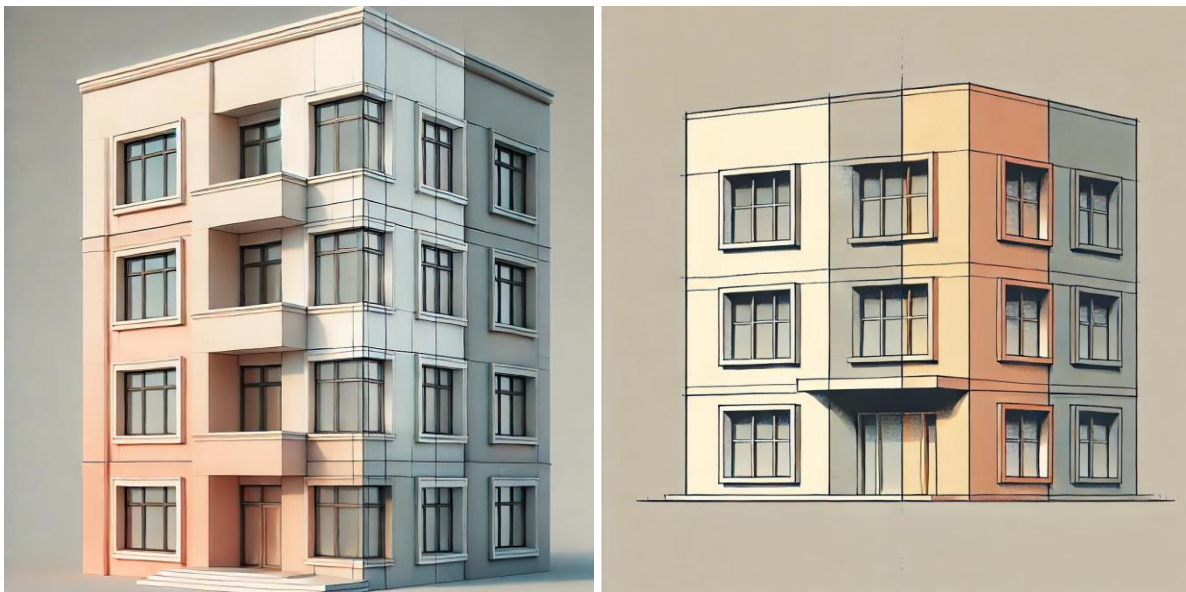
5. Moderna i suvremena arhitektura (1900. – danas). Npr. BMW Welt & BMWov muzej – Futuristički i modernistički stil s puno stakla i čelika. Allianz Arena – Visokotehnološki stadion, poznat po vanjskoj LED rasvjeti koja mijenja boje. München je kombinacija srednjovjekovne, barokne i moderne arhitekture, što ga čini jednim od najraznolikijih gradova u Njemačkoj.

Naravno da postoje i drugi stilovi (Slika 18), i primjećuje se različite dijelove zgrada prema tim stilovima, ali smatram da će biti dovoljno prikazati osnovne koji su nabrojani.



Slika 18. Pojednostavljeni različiti stilovi fasada i gradnje do danas.

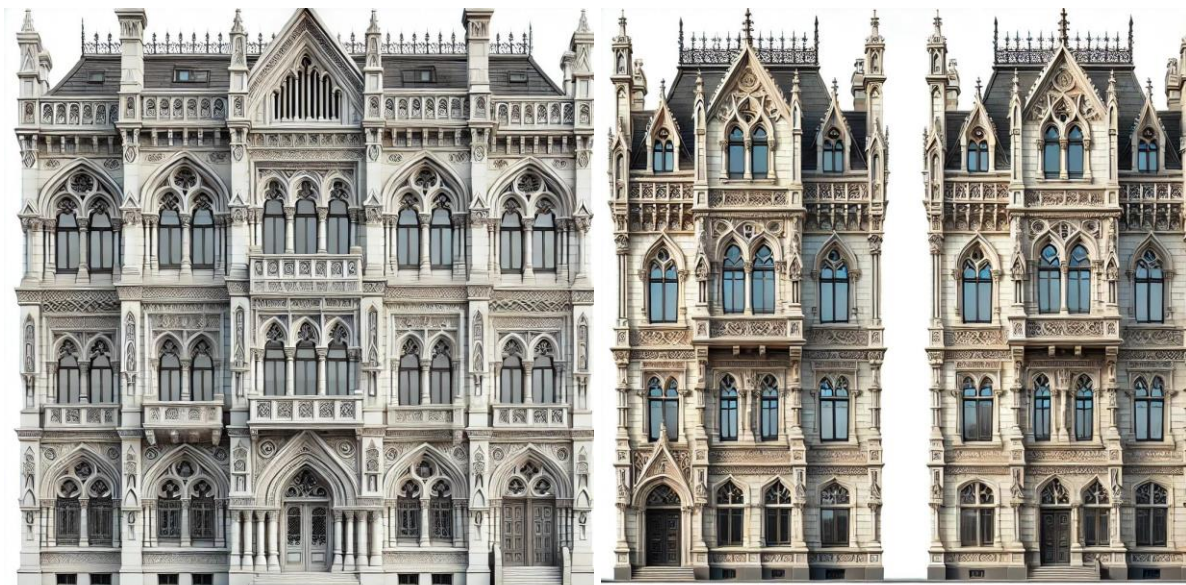
Prethodni primjeri su prikazani da se dobije bolji dojam i percepcija o svakom od nabrojanih stilova, a potom transformacijama i promjenom parametara stilova dolazimo do vizualnih rezultata za početnu jednostavnu zgradu (Slika 19) koja je nadalje prikazana kroz različite stilove (Slika 20, Slika 21, Slika 22, Slika 23, Slika 24, Slika 25, Slika 26), uz primjenu prethodno opisanih rješenja u predstavljenoj novoj metodi automatizacije procedurnog modeliranja zgrada. Počinje se od jednostavne zgrade koja je u napravljena u modernom stilu i nadalje prikazujem rezultate ili vizualizaciju te iste zgrade kako ju automatski dobijemo s predloženom metodom (dakle bez ručnih popravaka bilo kojeg dijela zgrade).



Slika 19. Dva primjera jednostavne zgrade koja je napravljena s osnovnim elementima prozora, vrata, i jednostavnim krovom zbog lakše analize i nakon toga transformacije u različite stilove.



Slika 20. Gotički stil. Vidljive su varijacije koje se događaju i sve razlike za lijevu i desnu zgradu sa (Slika 19) i njihove varijacije u transformacijama za gotički stil.



Slika 21. Također gotički stil za početne zgrade iz (Slika 19).



Slika 22. Slike modela (Slika 20) u perspektivnom prikazu. Ovdje su slike građevine u gotičkom stilu iz perspektive zraka koje prikazuju njezine šiljaste lukove, strmi krov, i zamršene arhitektonske detalje.

Gotički modeli (Slika 20) mogu se prikazati i u perspektivi, pri čemu se može primijetiti da ukoliko nemamo 3D podatke već samo ulazne podatke (Slika 19), za transformaciju dobijemo rezultate (Slika 22) koje odražavaju nedostatak podataka i tzv. izmišljeni ili nestvarni dubinski dio zgrade.



Slika 23. Barokni i rokoko stil.



Slika 24. Klasicizam i neoklasicizam.



Slika 25. Historicizam i neogotika.



Slika 26. Moderna i suvremena arhitektura.



Slika 27. Dijelovi za npr. gotički stil zgrade (URL 10).



Slika 28. Dijelovi za vrata za zgrade u rimskom stilu (URL 10).

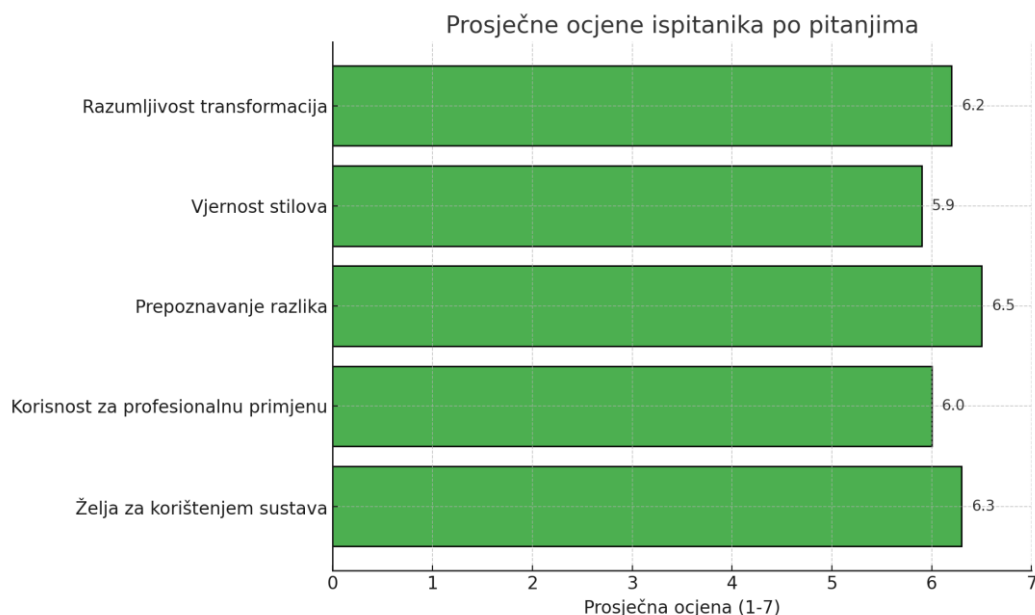
Prikazani su primjeri dijelova zgrade (Slika 27 i Slika 28) koji su korišteni za procedurno modeliranje u različitim stilovima.

Korištene biblioteke za strojno učenje bile su dostupne online i sve su upotrijebljene za formiranje predloženih stilova zgrada i poslije za provjeru ili evaluaciju s korisnicima. Upotrijebljene biblioteke su:

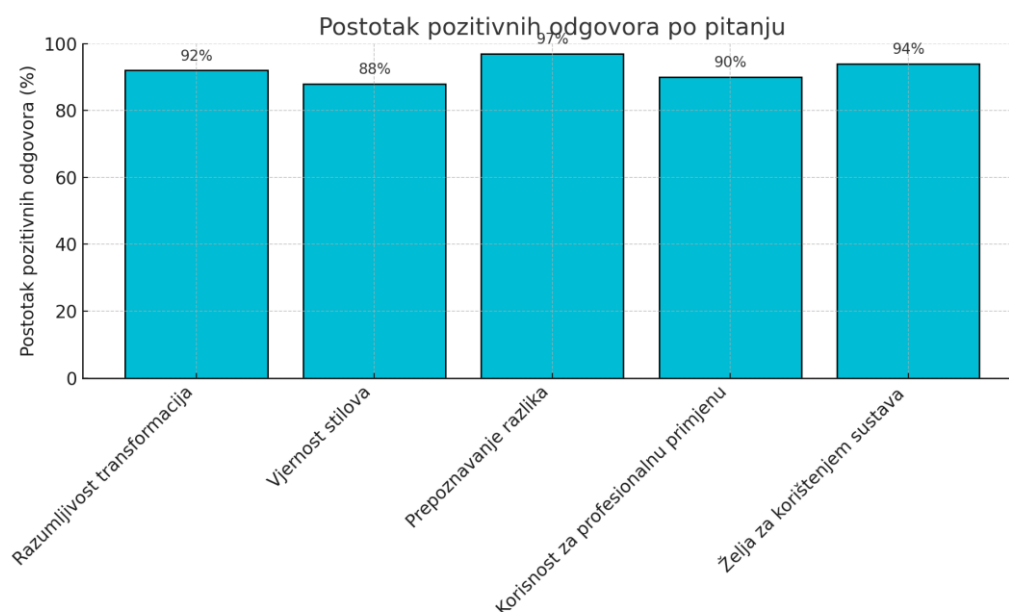
- Sketchfab (sketchfab.com)
- TurboSquid (turbosquid.com)
- CGTrader (cgtrader.com)
- 3D Warehouse (SketchUp) (3dwarehouse.sketchup.com)
- Evermotion (evermotion.org)
- BlenderKit (blenderkit.com)
- Quixel Megascans (quixel.com)
- Open3DModel (open3dmodel.com)
- KitBash3D (kitbash3d.com)
- Free3D (free3d.com)

2.5. Evaluacija sustava za transformaciju grafova

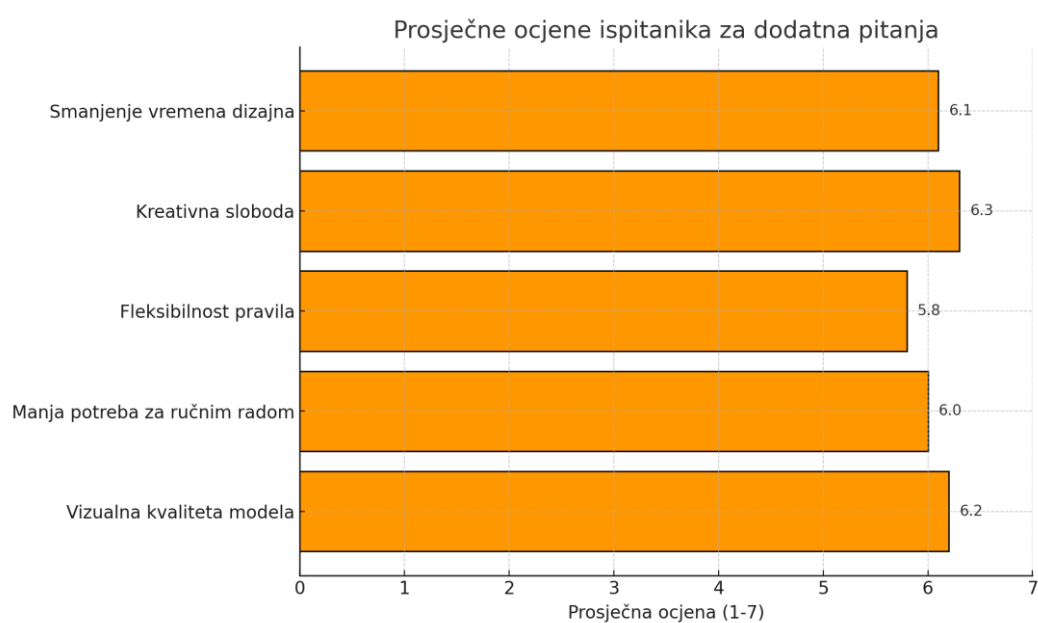
Nadalje se prikazuju rezultate evaluacije sustava za transformaciju grafova provedenog na uzorku od 59 stručnjaka iz područja kartografije, geodezije i arhitekture iz TUM i iz privatnih njemačkih tvrtki. Prikazani su dijagrami rezultata (Slika 29, Slika 30, Slika 31, Slika 32, Slika 33, Slika 34), detaljna diskusija i zaključci istraživanja u poglavlju.



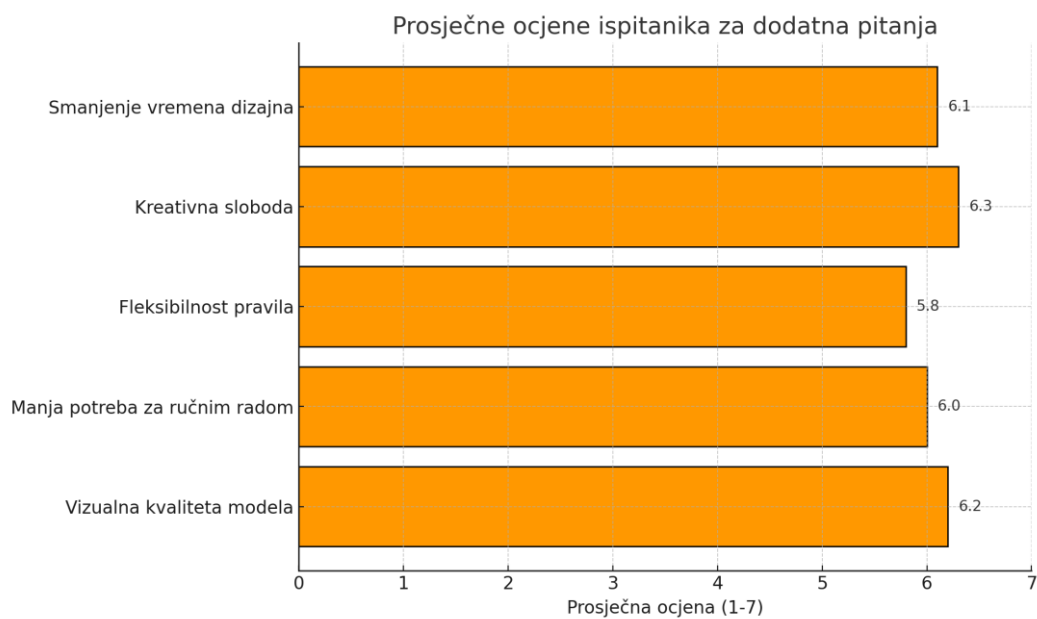
Slika 29. Prosječne ocjene ispitanika po pitanjima na Likertovoj skali (1-7).



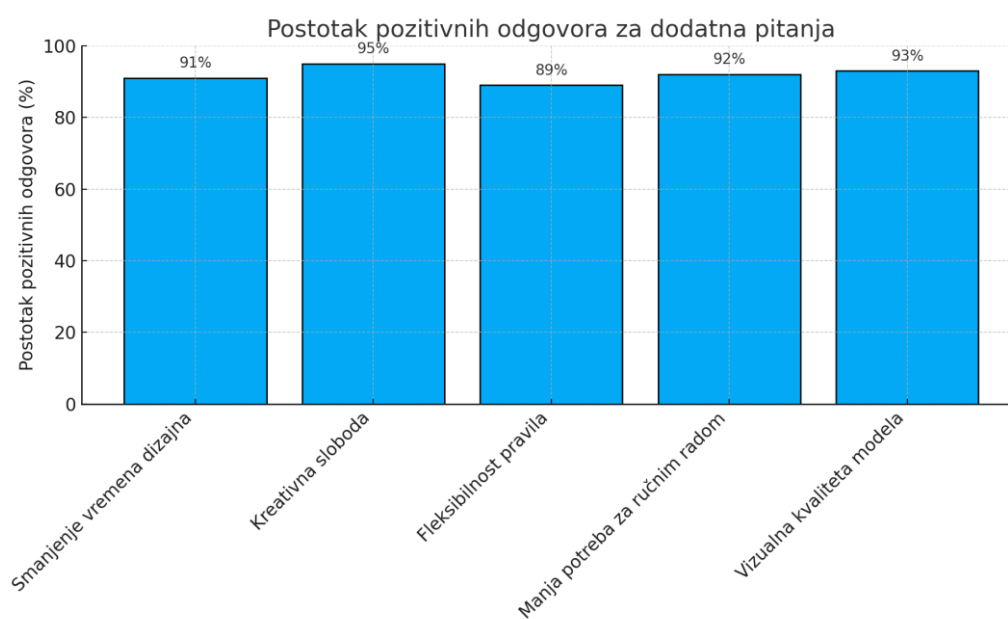
Slika 30. Postotak visoko pozitivnih odgovora (ocjene 5, 6, ili 7) za svako postavljeno pitanje.



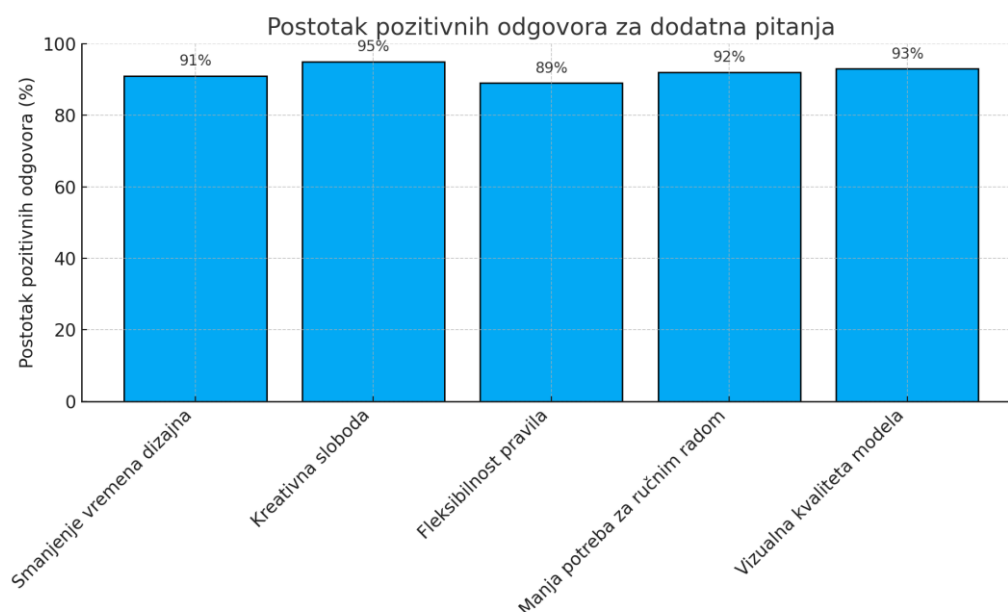
Slika 31. Prosječne ocjene za dodatna pitanja vezana uz funkcionalnost i korisnost sustava.



Slika 32. Prosječne ocjene za dodatna pitanja vezana uz funkcionalnost i korisnost sustava.



Slika 33. Postotak pozitivnih odgovora za dodatna pitanja.



Slika 34. Postotak pozitivnih odgovora na dodatna pitanja.

Evaluacija potvrđuje visoku razinu prihvaćenosti i zadovoljstva korisnika u ključnim aspektima kao što su brzina rada, fleksibilnost primjene pravila i kreativna sloboda u oblikovanju modela. Sudionici su u velikoj mjeri prepoznali da sustav smanjuje ukupno vrijeme potrebno za dizajn, istovremeno omogućujući veću kreativnost. Posebno su istaknuli kako pravila omogućuju jednostavne prilagodbe različitim arhitektonskim stilovima bez gubitka vizualne kvalitete. Iako su rezultati izrazito pozitivni, naglašena je potreba za dodatnim edukativnim sadržajem i vizualnim alatima za praćenje transformacija kako bi se dodatno olakšalo korištenje sustava korisnicima bez programerskog predznanja. Sveukupno, rezultati dodatnih pitanja snažno podržavaju hipotezu o efikasnosti i primjenjivosti razvijenog sustava za procedurno modeliranje u profesionalnim i edukativnim kontekstima.

Može se zaključiti da rezultati evaluacije ukazuju na vrlo visok stupanj prihvaćenosti sustava među ispitanicima. Najveće zadovoljstvo izraženo je kod prepoznavanja razlika između originalnih i transformiranih modela, dok su razumljivost transformacija i korisnost sustava također visoko ocijenjeni. Sudionici su prepoznali potencijal sustava za brzu i fleksibilnu modifikaciju stilskih karakteristika 3D modela, osobito u projektima koji uključuju veliki broj objekata. Sustav omogućuje automatizaciju složenih zadataka, uz minimalnu potrebu za ručnim intervencijama korisnika. Među preporukama za poboljšanje ističe se potreba za dodatnim vizualnim alatima za praćenje promjena u grafu te potreba za dodatnom edukacijom o sintaksi i definiranju pravila, osobito za korisnike bez programerskog predznanja. Evaluacija jasno potvrđuje da razvijeni pristup ima potencijal za primjenu u kartografskim vizualizacijama 3D

modela zgrada, kao i za daljnji razvoj u smjeru veće interaktivnosti i dostupnosti široj bazi korisnika.

Na temelju provedenog istraživanja može se zaključiti da sustav za prepisivanje grafova predstavlja inovativan i učinkovit alat za procedurno modeliranje 3D zgrada, povijesnih i modernih struktura. Visoka razina zadovoljstva među stručnjacima potvrđuje njegovu primjenjivost u profesionalnim projektima. Prednosti sustava uključuju brzu adaptaciju stilova, mogućnost masovne primjene na veći broj objekata te automatizaciju kompleksnih operacija bez dodatnog ručnog modeliranja. Uočene preporuke za poboljšanje, poput dodavanja vizualnih alata i jednostavnijeg definiranja pravila, nude jasne smjernice za buduće razvojne korake. Predloženi sustav značajno doprinosi suvremenim tehnikama procedurnog dizajna te otvara nove mogućnosti za učinkovito i kreativno oblikovanje 3D modela zgrada i njihovu vizualizaciju na 3D kartama i virtualnim 3D prostorima. Na temelju evaluacije možemo zaključiti da sustav za transformaciju grafova predstavlja izuzetno učinkovito rješenje za ubrzano i kreativno procedurno modeliranje. Kombinacija visoke fleksibilnosti, automatske primjene stilskih pravila i smanjenja potrebe za ručnim modeliranjem pruža značajne prednosti u arhitektonskom projektiranju i vizualizaciji. Predloženi sustav potvrđuje svoju primjenjivost u profesionalnim i akademskim kontekstima, s jasnim smjernicama za buduće unapređenje korisničkog iskustva i funkcionalnosti. S obzirom na rezultate evaluacije, razvijeni pristup predstavlja značajan doprinos procedurnom modeliranju i otvara nove mogućnosti za daljnji razvoj i profesionalnu primjenu.

U ovom poglavlju predstavljena je nova faza u razvoju procedurnog modeliranja, tzv. prepisivanje grafova. Predloženi pristup koristi generičke tehnike prepisivanja grafova za transformaciju skupa ulaznih pravila u nove skupove pravila, čime se omogućuje automatska ponovna upotreba, transformacija i generiranje pravila bez potrebe za ručnim pisanjem novih skripti. Primjena sustava prikazana je kroz procedurni jezik temeljen na CGA gramatikama, usmjeren na modeliranje fasada zgrada. Praktičnost pristupa demonstrirana je transformacijom stilova fasada iz jednog osnovnog modela u više različitih arhitektonskih stilova. Sustav pokazuje posebno veliku učinkovitost u situacijama kada treba napraviti iste izmjene na velikom broju zgrada, pri čemu su sve složene manipulacije automatizirane i transparentne za krajnjeg korisnika, bez dodatnog zahvata u modeliranju zgrada.

Kao prirodni nastavak razvoja, predlaže se izrada interaktivnog alata koji bi omogućio odabir i vizualizaciju pravila koja se primjenjuju na modele. Takav bi alat, dodatno povećao učinkovitost i intuitivnost sustava za krajnje korisnike i ostaje za buduća istraživanja. Posebno

su zanimljivi pravci razvoja kao što su apstraktniji mehanizmi selekcije elemenata, identifikacija i generalizacija pravila, te automatsko generiranje dizajna zgrada na temelju primjera. Umjetna inteligencija nije uključena u istraživanje, jer za sada još nije na tom stupnju razvoja da bi mogla samostalno analizirati 3D modele zgrada.

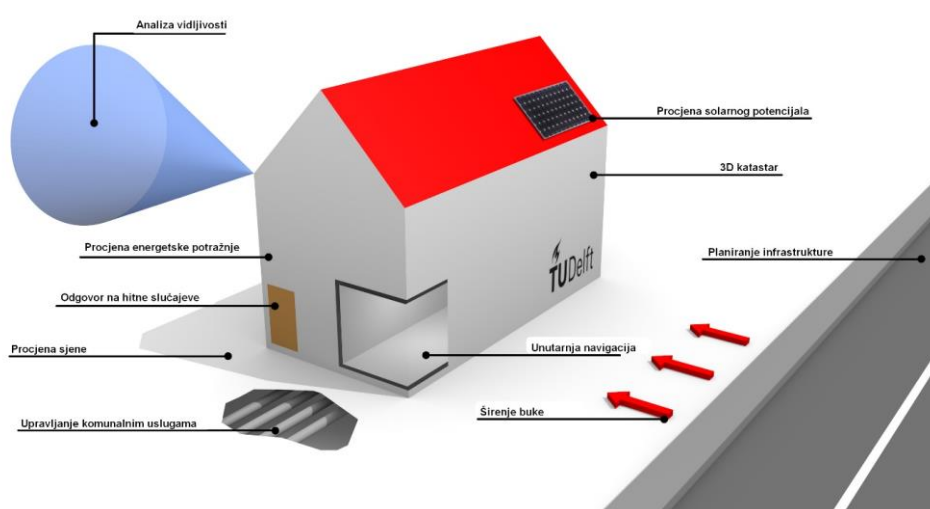
3. Optimizacija 3D modeliranja krovova zgrada korištenjem procedurnih metoda i CGA pravila

Glavna svrha procedurnog modeliranja jest primjena jednog skupa pravila na mnoštvo zgrada ili čitave urbane cjeline, no to često ne rezultira preciznim i vjernim prikazom objekata. Trodimenzionalni prikazi zgrada i gradova sve se češće koriste u prostornom planiranju, donošenju strateških odluka i optimizaciji upravljanja prostorom. Procedurene metode olakšavaju proces modeliranja smanjujući potrebu za ručnim radom. Međutim, razumijevanje modela otežano je kad su dostupna samo pravila, bez vizualnog rezultata. Stohastički modeli, koji za isti zadatak mogu ponuditi više rješenja, dodatno otežavaju uvid u mogući raspon dizajna. Ti izazovi nisu ograničeni samo na korisnike bez programerskih vještina. Ovaj rad pruža konkretne prijedloge za automatizaciju dijelova procesa procedurnog modeliranja 3D građevina.

3.1. Uloga 3D modela u urbanom planiranju i važnost razine detalja (LOD)

Ovdje je fokus stavljen na procedurno 3D modeliranje i vizualnu prezentaciju zgrada, posebice krovnih struktura, jer se njihove varijante mogu generirati parametarski ili pomoću pravila. Dok se do sada modeliranje krovova temeljilo na promjeni parametara, u ovom istraživanju predlažu se izmjene unutar skripte koje omogućuju izradu i jednostavnih i kompleksnih krovnih oblika. Takav pristup znatno skraćuje vrijeme potrebno za modeliranje – što je ključno u širim kontekstima, poput izrade modela čitavih gradova, gdje se očekuje gotovo trenutačna vizualizacija u stvarnom vremenu. Sve to istraženo je u softveru CityEngine, koji koristi procedurni pristup i često se primjenjuje za 3D urbanističko modeliranje. Prvo su preuzeti svi potrebni podaci dostupni javno putem interneta, uključujući attribute krovova, a rezultati su prikazani putem CityEngine WebScene uz vremensku analizu izrade više tipova krovova unutar istog modela. Također, stručnjaci iz geodetske struke i korisnici prostornih podataka sve češće primjenjuju 3D modele gradova. Takvi prikazi koriste se u procjenama za postavljanje solarnih panela, analizama sjene zgrada, širenja buke i drugim aspektima urbanog planiranja (Slika 35). Kao i kod 2D podataka, i trodimenzionalni modeli nude pojednostavljenu reprezentaciju stvarnosti. Koliko je sadržaj detaljan ovisi o konkretnim potrebama korisnika, izvorima informacija, tehnicima prikupljanja i prostornim relacijama. Stupanj složenosti modela određen je razinom detalja (LOD), koja ukazuje na preciznost geometrije i atributa te predstavlja ključan pojam u GIS-u i urbanom modeliranju. Ručno kreiranje detaljnih modela vrlo je naporno. Primjer je zgrada Sveučilišta u Brnu, čiji je model – s visokim LOD-om – izrađivala skupina

studenata i nastavnika pune godine. U tom kontekstu procedurne metode predstavljaju efikasnu alternativu. Omogućuju pohranu i ponovnu uporabu modela, uz minimalnu ručnu intervenciju. Nakon što se definira skup pravila (gramatika), moguće je proizvoljno stvarati raznolike varijante promjenom nekoliko ulaznih vrijednosti. Ovaj pristup može pokriti čitav spektar objekata – od bilja, namještaja, stambenih jedinica pa sve do čitavih urbanih cjelina. Zapravo, koncept LOD prisutan je u svim fazama 3D modeliranja – uključujući i inicijalno planiranje, još prije samog prikupljanja podataka. Budući da on utječe na odluke poput uključivanja krovova ili prometnica u model, LOD postaje relevantan i kod nabave podataka. Naime, skupovi podataka razlikuju se po razini detalja, a to određuje i njihovu vrijednost. Pri planiranju snimanja, uzimanje LOD-a u obzir nužno je za pravilan raspored resursa. Različite razine detalja zahtijevaju i različite metode – primjerice broj prolazaka kod zračnog laserskog skeniranja. LOD time postaje vodič u procesu prikupljanja i opisuje dubinu zahvata. Pojam se koristi već desetljećima u kontekstu 3D urbanih modela. Osim stvarnog prikaza, 3D modeli često se koriste i za zamišljene projekte, u arhitekturi, vizualizaciji ili simulacijama budućih razvojnih scenarija. LOD utječe i na aspekte poput pohrane, održavanja i krajnje upotrebe modela. Premda se većina studija fokusira na 3D prikaz zgrada, preporučuje se širenje fokusa i na ostale gradske elemente, kao što su prometnice ili vegetacija. Trenutačno kontrola kvalitete unutar 3D GIS sustava ne uključuje eksplicitnu procjenu LOD-a, što predstavlja značajan nedostatak. Stoga bi buduća istraživanja trebala uključiti integraciju razina detalja u okvire standarda za procjenu kvalitete prostornih podataka – kako bi se osigurala veća dosljednost i primjenjivost u praksi.



Slika 35. 3D modeli gradova imaju mnoštvo različite primjene za vizualizaciju okoliša i podršku odlučivanju u različitim domenama ljudskog djelovanja (modificirano iz (Biljecki et al., 2015)).

3.2. Pregled pristupa modeliranju zgrada i izazovi automatske konstrukcije

Tijekom posljednjih nekoliko godina, trodimenzionalni prikazi gradova postaju sve važniji u mnogim primjenama, što je rezultiralo porastom potražnje za njihovom izradom. Glavni izazov u njihovoj izradi leži u velikoj raznolikosti i kompleksnosti oblika gradnje (Adão et al., 2017; De Santis et al., 2018). Unatoč velikom broju predloženih metoda za modeliranje zgrada u zadnjih dvadesetak godina (Brenner, 2005, 2010; Haala & Kada, 2010; Wang, 2013), to područje i dalje ostaje vrlo aktivno unutar više znanstvenih grana. Postoje različiti načini klasifikacije pristupa modeliranju zgrada, a jedan od najčešćih temelji se na razini ljudske intervencije, izvorima podataka, tipu modela te smjeru kontrole – odozgo prema dolje, odozdo prema gore (podatkovno vođeno) ili hibridno. U literaturi nalazimo niz primjera hibridnih pristupa koji kombiniraju prednosti oba smjera: fleksibilnost podatkovno vođenog pristupa te mogućnosti transformacije oblika u modelom vođenom pristupu (Tarsha-Kurdi et al., 2007). Jedan takav primjer koristi podatke za definiranje osnovne geometrije krova, a model za dodavanje detalja poput potkrovlja (Satari et al., 2012). Jedan od izazova koji ostaje neriješen je automatska rekonstrukcija malih elemenata na krovovima, poput dimnjaka i mansardi. Uz rekonstrukciju postojećih objekata, raste i potreba za generiranjem potpuno virtualnih urbanih prostora – primjerice za upotrebu u industriji zabave i simulacija – gdje je brza i ekonomična produkcija ključna. Upravo tu značajnu ulogu igra procedurno modeliranje. Postojeće metode često se temelje na oskudnim ulaznim podacima te koriste različite razine formalizacije pravila za generiranje varijacija objekata. Nakon što se razvije procedurni opis (skup pravila), moguće je jednostavno manipulirati parametrima i stvarati raznovrsne modele. Ipak, rekonstrukcija kompleksnih i nepravilnih tlocrtnih oblika, kao i zgrada koje uključuju i interijer i eksterijer, ostaje izazov. Prva pretpostavka u ovom radu jest da je moguće automatski generirati zgrade nepravilnih tlocrtnih oblika. Moderne metode koriste sustave pravila (tzv. gramatike), od kojih su najpoznatije CGA pravila i L-sustavi (Lindenmayerove gramatike). L-sustavi primjenjuju pravila paralelno i koriste se i za modeliranje fraktala, dok CGA pravila omogućuju detaljno definiranje geometrije i atributa zgrada. Međutim, uporaba takvih sustava zahtijeva određeno programersko znanje, što može predstavljati prepreku stručnjacima iz drugih područja. Jedan od problema je i ograničena mogućnost razumijevanja modela samo iz pravila, pogotovo kod stohastičkih procedurnih modela koji za isti ulaz generiraju više varijanti. Također, kombinacija različitih pravila često otežava automatsku interoperabilnost, a raspon transformacija može biti ograničen. Uz to, postoje i tehnička ograničenja u samom prikupljanju podataka. Lasersko

skeniranje nudi visoku točnost, ali zahtijeva skupu opremu i dodatnu obradu, dok fotogrametrijski pristupi omogućuju jeftinije prikupljanje podataka, ali su ograničeni kvalitetom snimaka i zahtijevaju točnu kalibraciju. Na kraju, suvremena istraživanja u procedurnom modeliranju urbanih prostora – primjerice kroz CGA pristup – omogućuju visoku razinu vizualnog i geometrijskog detalja uz minimalan unos podataka, čineći ih idealnima za automatsku izradu vanjskih dijelova zgrada.

3.2.1. Pametni gradovi i potreba za ubrzanim vizualnim modeliranjem

Urbanističko planiranje je tradicionalno interdisciplinarna aktivnost, no u suvremenom kontekstu koncepta pametnog grada dodatno se usložnjava. Novi izazovi proizlaze iz pojave umjetnih infrastruktura koje generiraju kompleksne skupove podataka, čija obrada zahtijeva naprednu analitiku. Uz to, sve snažnije računalne mogućnosti i razvoj računalne grafike omogućili su širu dostupnost sofisticiranih alata korisnicima s naprednim digitalnim vještinama. Istovremeno, otvoreni kod i dostupni online resursi učinili su digitalne urbane podatke pristupačnijima no ikada (Healey, 2016). Takva tehnološka ekspanzija pruža velike mogućnosti za planiranje koje uključuje geovizualizaciju i infrastrukturnu analizu, ali stvara i potrebu za novom vrstom stručnjaka. Ti stručnjaci moraju povezivati znanja iz urbanizma, geoinformatike, analitike i simulacija s planerskim metodama. Osim toga, mijenjaju se odnosi između podataka i urbanih aktera, a načini komunikacije između njih postaju složeniji i još uvijek nisu u potpunosti razjašnjeni. Unatoč svim neizvjesnostima, ovaj tehnološki val pokrenuo je rastući interes za primjenu virtualnih gradova u svrhu prepoznavanja prostornih obrazaca te ispitivanja posljedica različitih intervencija ili urbanih scenarija. Ipak, ovakve analize su često kompleksne, zahtijevaju uključivanje vanjskih stručnjaka, a rezultati su nerijetko teški za interpretaciju, ažuriranje i komuniciranje prema javnosti. Kao rezultat toga, urbane analize rijetko postaju dio šire javne rasprave, niti se dosljedno primjenjuju u stvarnim planerskim procesima. Iako tehnološke aplikacije postaju sve naprednije, ostaju nedovoljno prepoznate u praksi. Strategijsko planiranje i dalje je složen, iterativan proces koji traži suradnju brojnih disciplina i suočava se s rastućom složenošću urbanih sustava. No upravo sadašnja era donosi potencijal za stvarnu interdisciplinarnost – kroz integraciju stručnih znanja i sposobnosti za komunikaciju između planera i zajednice. Da bi se takva integracija ostvarila, potrebno je razviti odgovarajući okvir i sučelje koje omogućuje spajanje raznovrsnih podataka i ekspertiza, uz istovremenu prilagodljivost eksperimentima unutar koncepta pametnih gradova. Iako su već pokrenuti pokušaji izgradnje integriranih sustava, oni su često nepovezani – više zbirka

odvojenih scenarijskih modela nego koherentan sustav. Upravo tu leži prostor za razvoj cjelovitog konceptualnog i analitičkog okvira koji bi sustavno obuhvatio viziju pametnog grada.

3.3. Istraživački okvir: poboljšanje procedurnog modeliranja kroz CGA pravila i urbanu geovizualizaciju

Ispitivanje mogućnost korištenja različitih tehnika vizualnog i urbanog modeliranja u svrhu informiranja i evaluacije novih pristupa geovizualizaciji treba biti na način koji je sveobuhvatan i primjenjiv u praksi. Ključno je ispitati kako se metode urbanog modeliranja provode, u kojoj su mjeri bile uspješne, te kakva je njihova logička i praktična struktura. Iako postoji niz mogućih istraživačkih pravaca, ovo istraživanje fokusira se na unaprjeđenje procedurnog modeliranja, posebice na automatsku generaciju zgrada. Cilj je razviti metode koje omogućuju varijacije unutar modela pomoću parametrizacije CGA pravila – čime se povećava fleksibilnost i skalabilnost. Trodimenzionalno modeliranje zgrada aktivno se razvija još od 1990-ih, no i dalje postoje izazovi u pogledu točnosti, koherentnosti i generalizacije, osobito pri modeliranju većih urbanih cjelina. Uz to, pojave poput mobilnih sustava za kartiranje pružaju nove mogućnosti obogaćivanja modela zgrada, osobito na razini fasade – uključujući geometriju, oblik i teksturu. Kombinacijom podataka iz zraka i s tla moguće je dobiti kvalitetniji i detaljniji prikaz objekata. Primjenom zračnih snimaka i oblaka točaka s ulice, istraživanja su razvila metode koje uključuju: segmentaciju i klasifikaciju točaka, geometrijsko modeliranje pročelja, te teksturiranje bez okluzije – sve s ciljem generiranja cjelovitih i kvalitetnih 3D modela zgrada. Virtualni modeli zgrada od iznimne su važnosti u disciplinama poput geodezije, kartografije, arhitekture, arheologije i razvojnih industrija poput računalnih igara. Ovisno o primjeni, zahtjevi mogu varirati između brzog popunjavanja virtualnih prostora i znanstvene analize kroz digitalne rekonstrukcije. Kod procedurnog modeliranja, nakon što se izgradi odgovarajući skup pravila (tzv. gramatika), moguće je generirati niz varijacija jednostavnim mijenjanjem nekoliko parametara – što je izuzetno korisno za modeliranje složenih oblika i tipologija gradnje. Jedan od najpoznatijih alata za ovakav pristup je CityEngine, koji omogućuje modeliranje čitavih urbanih sredina pomoću ograničenog skupa ulaznih podataka. Slika 36 prikazuje shemu kreiranja modela gradova pomoću softvera CityEngine. Korištenjem L-sustava i CGA gramatike, korisnici mogu definirati strukturu i ponašanje modela, čime se postiže automatska i kontrolirana proizvodnja virtualnih gradova. U praksi, proces uključuje sljedeće korake, izrada CGA pravila, primjena pravila na osnovne geometrijske oblike, generiranje zgrada uz pomoć statističkih i geografskih ulaznih podataka i vizualizacija rezultata u softveru koji može

interpretirati poligonalnu geometriju i teksture. CityEngine također omogućuje generiranje pročelja zgrada i integraciju različitih karata kao ulaznih parametara. Istraživanja su pokazala da se za izradu cjelovitog grada mogu koristiti dva L-sustava: jedan za ulice, a drugi za zgrade. Važno je istaknuti kako su CGA pravila oblikovana tako da opisuju i razvijaju oblik kroz simbole, geometrijske i numeričke atribute. Upravo ti atributi definiraju prostorni okvir i njegove transformacije (npr. rotacije, translacije), omogućujući vrlo preciznu i fleksibilnu kontrolu nad konačnim izgledom modela.



Slika 36. Standardni tijek rada u CityEngine-u (URL 12).

Jedna od najučinkovitijih metoda za generiranje velikih 3D modela u kratkom vremenu i s minimalnim troškovima je procedurno modeliranje. Ova metoda se temelji na primjeni pravila i algoritama za automatsku izradu modela, čime se značajno ubrzava proces modeliranja kompleksnih urbanih okruženja. Jedan od najpoznatijih alata u tom području je Esri CityEngine, koji koristi CGA pravila (Computer Generated Architecture) – skriptni jezik posebno razvijen za generiranje arhitektonskog sadržaja. Dosadašnja istraživanja potvrđuju da je CityEngine iznimno prikladan za izradu 3D modela urbanih prostora. Temeljna prednost procedurnog modeliranja leži u mogućnosti primjene jedinstvenog seta pravila na velik broj objekata, npr. cijele gradske četvrti. Međutim, takav pristup ima ograničenja kad je cilj postići visoku razinu detalja – poput preciznih fasadnih ukrasa, terasa, balkona ili izračuna volumena i površina. U tim slučajevima, često je potrebno razviti zaseban kod za svaku pojedinu zgradu, pa čak i dodatno doraditi model ručno pomoću alata za poligonalno modeliranje. U sredinama gdje su zgrade vrlo raznolike u pogledu stila i konstrukcije, procedurno modeliranje može postati dodatno kompleksno i vremenski zahtjevno. Unatoč tome, ono ostaje najbolji izbor za generiranje LOD1 modela za šira područja ili LOD2 modela u četvrtima gdje su zgrade međusobno slične. U tim kontekstima, procedurno modeliranje se ističe kao brz i učinkovit način izrade 3D modela s niskom do srednjom razinom detalja. Takvi modeli mogu poslužiti kao podloga za urbanističke analize, razvoj pametnih gradova, turističku vizualizaciju i druge digitalne primjene. Poseban potencijal za poboljšanja unutar procedurnog pristupa vidi se kod

modeliranja krovova, budući da su oni često raznoliki u geometriji. Pronalaskom obrazaca i sličnosti među tipovima krovova moguće je razviti metode koje dodatno ubrzavaju njihovu izradu, čineći procedurno modeliranje još učinkovitijim u realnim scenarijima.

3.4. Formalna gramatika definirana četverostrukim skupom

Procedurno modeliranje, koje se temelji na definiranju skupa pravila (gramatika), svoje teorijske početke pronalazi u konceptu formalnih gramatika koje je uveo Noam Chomsky 1956. godine (Chomsky, 1956b). Formalna gramatika definira se kao sustav:

$G = \langle NT, T, \omega, P \rangle$, gdje

NT - označava skup neterminalnih simbola,

T - skup terminalnih simbola,

$\omega \in NT$ - je početni simbol (aksiom),

P - je skup pravila proizvodnje.

Svako pravilo definira kako se niz simbola može transformirati: s lijeve strane stoji uzorak (najčešće s neterminalom), a desna strana daje novi niz simbola. Kroz uzastopnu primjenu pravila, od početnog simbola se generira niz, a kada on sadrži samo terminale, smatra se završenim. Najčešće se u računalnoj grafici koristi gramatika bez konteksta (tip 2), dok su kontekstno osjetljive gramatike (tip 1) rjeđe. Izvođenje gramatike vodi do stvaranja derivacijskog stabla, koje prikazuje tok transformacija. Iako se suvremeni sustavi procedurnog modeliranja više ne oslanjaju strogo na formalnu gramatiku, pravila ostaju njihova temeljna komponenta. Zbog toga se i dalje u literaturi nazivaju gramatikama, iako nadilaze izvorni teorijski okvir. Među ključnim tehnikama utemeljenima na gramatičkom pristupu ističu se dvije, L-sustavi (Lindenmayerovi sustavi). Zahvaljujući svojim svojstvima paralelnog izvođenja pravila, L-sustavi se često koriste za simulaciju fraktalnih struktura i modeliranje morfologije organizama. Također imaju primjenu u vizualnim platformama poput Sketchfaba (URL 10 i URL 11) i Gramatike oblika (Shape Grammars). Za razliku od L-sustava koji rade s nizovima simbola, gramatike oblika operiraju direktno nad grafičkim elementima – poput linija, pravokutnika i krugova. Kod ovog pristupa, pravila preslikavaju jedan raspored geometrijskih oblika u drugi. Poznat primjer je tzv. paladijska gramatika, kojom se generiraju tlocrti vila u stilu renesansnog arhitekta Palladija. Ova dva pristupa čine temelj u procedurnom modeliranju, gdje je cilj generirati raznolike i detaljne 3D strukture (poput zgrada ili cijelih urbanih područja) s visokom razinom kontrole i mogućnosti varijacije, uz automatizaciju procesa.

3.4.1. Nova metoda oblikovanja jednostavnih i složenih krovnih konstrukcija

U zadanim postavkama CGA pravila, sljemena krova – točke gdje se susreću krovne plohe – ne uključuju automatski tri vodoravne ravnine koje bi definirale oblik sljemena. Kako bi se to omogućilo, u skripti se koristi naredba `set(trim.horizontal, true)` i postavlja se ispred komponente `split`, čime se aktivira horizontalno obrezivanje ploha. Međutim, kod mnogih krovnih oblika, sljemena ili grebeni se prikazuju kao ravnomjerni, bez stvarnih varijacija u nagibu. U takvim slučajevima, parametar `even` koji bi trebao omogućiti razlike, ne daje željeni učinak. To dovodi do problema u realističnosti modela, pa je potrebna dodatna optimizacija geometrije krova unutar CGA skripte, što je i općepoznato. Nakon što se komponenta podijeli (`split`), svaka strana krova dobiva svoje sljemene plohe koje se umeću tijekom izvođenja. No, prema zadanim vrijednostima, horizontalne trim ravnine nisu prisutne. Zato je važno znati da atribut `trim` u CGA-u ima dvije vrijednosti koje definiraju kako i kada će se trim ravnine primijeniti na trenutni oblik. Vrsta trim ravnine ovisi o orijentaciji zajedničkog ruba između dviju susjednih ploha koje se dijele po kriteriju da ako je kut prema xz ravnini manji od 40°, smatra se da je trim ravnina vodoravna a ako je kut veći, ravnina se klasificira kao okomita (Slika 40). Prema URL 12, ovakva segmentacija i kontrola nad obrezivanjem omogućuju da se unutar skripte unaprijed definira više različitih tipova krovova (Slika 37), koji se zatim mogu primijeniti po potrebi, a prikazani su oblici krovova kao što je ravni krov, krov kuka, zabata, piramide i šupe. Kod složenijih krovova, poput onih s vertikalnim ili horizontalnim odrezima u specifičnim ravninama, npr. prikazanim ružičastom bojom (Slika 41), potrebna su dodatna rješenja. U tu svrhu koriste se kratke skripte unutar CGA pravila, koje omogućuju definiranje smjera i primjene reznih ravnina. Na primjer, za horizontalne ravnine obrezivanja, moguće je prebacivanjem atributa rezne ravnine kontrolirati kako će se krovšte rezati i prikazivati. Ravnine za obrezivanje, jasno vizualizirane u ružičastoj boji, služe za dodatnu segmentaciju i kontrolu nad krovnim ploham, osobito kod nagnutih i višedijelnih krovova.

Skripta automatski dodjeljuje `roofType` ovisno o visini objekta (ili nekom drugom kriteriju iz podataka):

```
def assign_roof_types(objects):  
    for o in objects:  
        height = o.getAttribute("avgHeight")  
        if height < 5:  
            o.setAttribute("roofType", "flat")  
        elif height < 10:
```

```
o.setAttribute("roofType", "gable")
```

else:

```
o.setAttribute("roofType", "hip")
```

Prikaz i testiranje se nakon generiranja, u Inspector panelu provjerava koji su atributi aktivni (npr. `roofType`, `trim.horizontal`). Po potrebi ih se mijenja ručno ili automatski pomoću skripte.

Ako treba orijentirati krov na temelju prve plohe tada je:

Roof -->

```
alignScopeToGeometry(yUp, 0)
```

```
roofShed(30)
```

a za ručno postavljanje orijentacije:

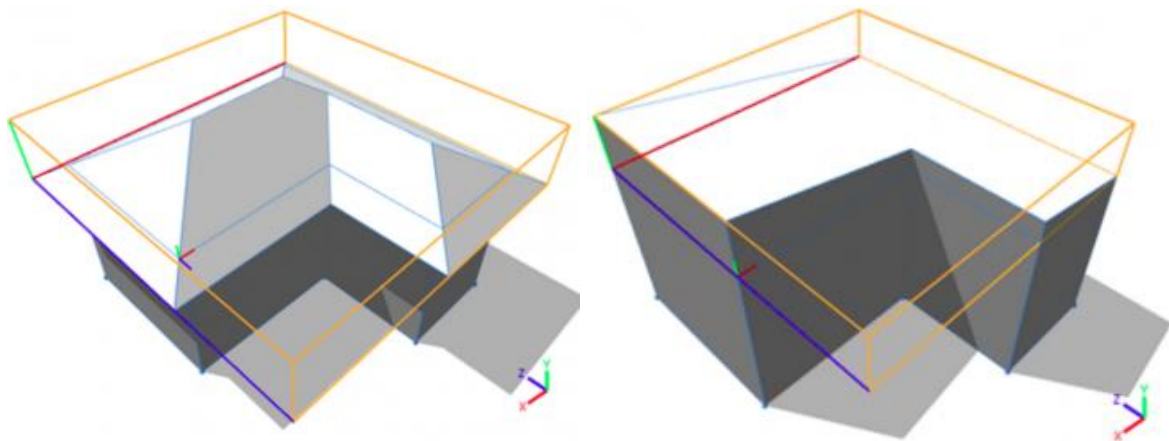
setupOrientation -->

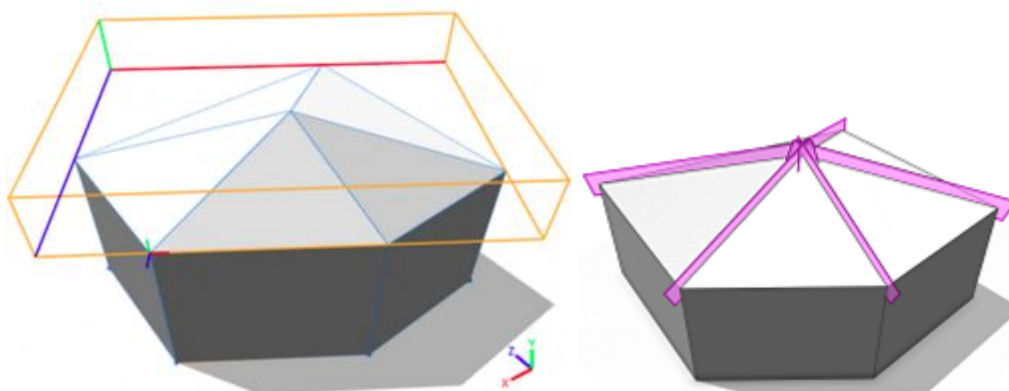
```
rotateScopeToAxes(y, z)
```

itd.....

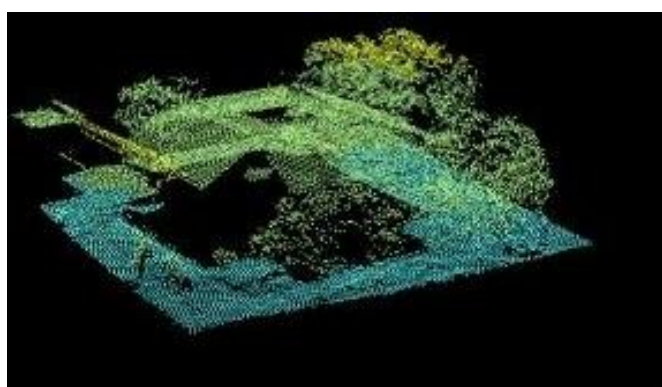
a cijela se skripta može preuzeti na URL 13 i podržava višedijelne (složeniye) krovove, uključujući podjelu krova u više visinskih segmenata (različite razine), različite tipove krovova po segmentima, fleksibilan način dodjele po segmentu temeljen na visini (simulacija podataka iz npr. LiDAR-a, Slika 38), te očuvanje kontrole reznih ravnina i scope orijentacije. U CityEngineu je upotreba jednostavna, potrebno je samo kreirati novi rule file i zalijepiti ovaj kod, potom ga spojiti na Layer s tlocrtima zgrada (npr. building footprints). Povezati atribut `roofType` s podacima (npr. iz CSV-a ili Python skripte) i pokrenuti "Generate" i može se testirati s različitim vrijednostima *roofType*.

Nakon izvođenja skripte, korisnik može odabrati tipove krovova iz unaprijed pripremljenih opcija, koje se potom mogu generirati i vizualno provjeriti radi dodatne optimizacije i usklađivanja s realnim podacima (Slika 41).

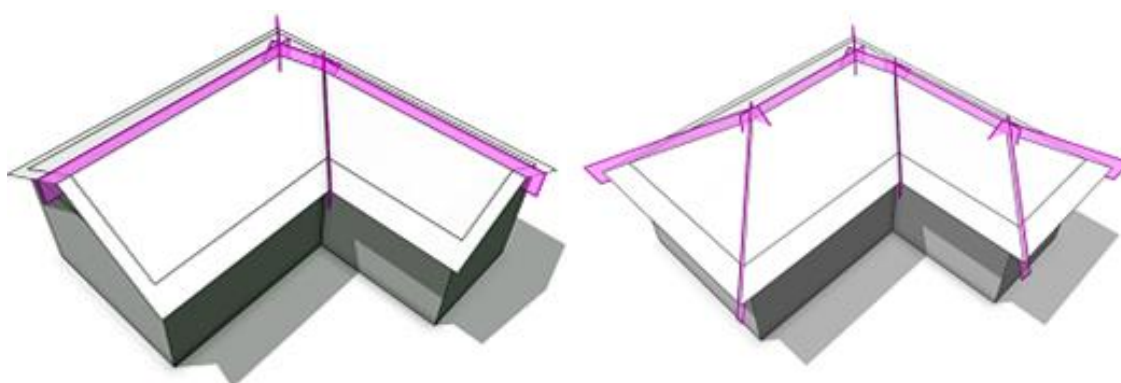




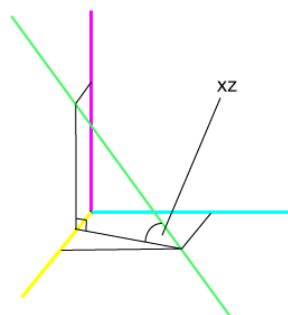
Slika 37. Različita geometrija krovova različitih oblika zgrada (URL 12).



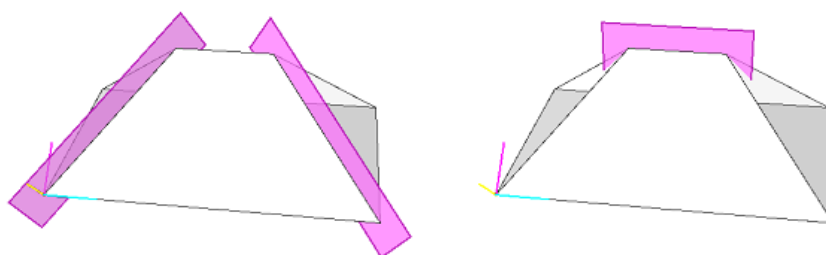
Slika 38. Lasersko snimanje i dobivanje vjernog prikaza krova zgrade u odnosu na dobivanje istovjetnog složenog oblika krova iste zgrade pojednostavljenim procedurnim modeliranjem (URL 12).



Slika 39. Nakon podjele komponente, svaka strana krova sadrži lomne ravnine prilikom umetanja (URL 12).

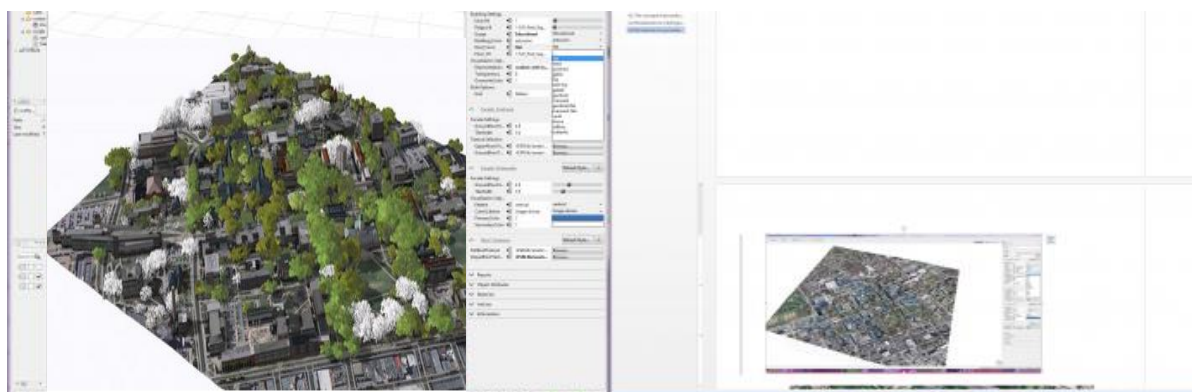


Slika 40. Klasifikacija trim ravnine ovisi o kutu između "generirajućeg" ruba i xz ravnine (URL 12).



Slika 41. Okomite ravnine za rezanje omogućene su prema zadanim postavkama (URL 12).

Ravnine obrezivanja prikazane su ružičastom bojom (Slika 39). Za generiranje različitih tipova krovova moguće je izabrati sve tipove koji su prethodno uneseni unutar skripte, odnosno preddefinirani, a cijeli model se može prikazati uz generiranje po potrebi i provjeru rezultata (Slika 42).



Slika 42. Primjer generiranja dijela grada procedurnim modeliranjem (URL 12).

3.5. Diskusija

U ovom poglavlju pokazalo se da procedurno modeliranje, unatoč svojoj osnovnoj orijentaciji na automatizaciju i učinkovitost, može uspješno odgovoriti i na složenije izazove 3D modeliranja kada se pravilno strukturira i optimizira. Kroz kombinaciju CGA pravila,

višeslojnih pristupa i integracije dodatnih podataka (npr. visina iz LiDAR-a ili atributnih izvora), moguće je znatno povećati razinu realističnosti i fleksibilnosti generiranih modela krovova. Primjena višesegmentnog modeliranja omogućuje korisnicima da ne tretiraju zgradu kao jedinstven entitet s jednim tipom krova, već kao skup međusobno povezanih volumena. Ovakav pristup ne samo da povećava vizualnu točnost, već otvara prostor za analitičke primjene kao što su procjene volumena, solarna analiza, simulacije toka zraka i slično. U kontekstu pametnih gradova, gdje su vizualizacija, analiza i simulacija sve važniji, ovakav pristup može biti ključan. Unatoč prednostima, modeliranje složenih krovova CGA pravilima nije bez izazova. Potrebna je visoka razina razumijevanja logike modeliranja, a za generiranje realističnih krovova često je nužno koristiti dodatne podatke ili pristupiti pisanju kompleksnijih skripti. U slučajevima kada zgrade imaju vrlo heterogenu arhitektonsku strukturu, potrebno je dodatno ručno intervenirati ili uspostaviti naprednije algoritme za klasifikaciju krovova. Implementacija horizontalnih i okomitih reznih ravnina te dinamička kontrola preko parametara kao što su *trim.horizontal* i *roofType* pokazala se korisnom i nužnom za detaljnu segmentaciju krovnih ploha. Uvođenjem dodatne logike unutar CGA pravila moguće je generirati više varijacija krovova uz manju količinu ručnog rada, što je jedan od temeljnih ciljeva procedurnog pristupa. Doprinos leži u prikazu kako se složene morfološke karakteristike zgrada, posebno krovova, mogu uspješno modelirati unutar jednog procedurnog sustava, a da se pritom zadrži automatiziran koncept. Predloženi pristup pokazuje se skalabilnim i primjenjivim u raznim kontekstima – od kartografije i urbanističkog planiranja, do virtualnih okruženja i geoinformacijskih sustava. Nadalje u nekom budućem radu može se napraviti povezivanje CGA pravila s klasificiranim oblacima točaka (npr. ML klasifikacija krova iz LiDAR-a) uz postojeću automatizaciju generiranja *roofType* atributa putem vanjskih Python skripti i izgradnja repozitorija često korištenih kombinacija krovništa za ponovnu upotrebu. S obzirom na stalni razvoj alata i sve veću dostupnost prostornih podataka visoke razlučivosti, procedurno modeliranje zadržava svoju važnost kao pristup koji balansira između automatizacije i kontrole – što je ključno u današnjoj geovizualizaciji objekata realnog svijeta oko nas.

4. Procedurni pristup automatskoj izradi 3D modela zgrada u urbanim sredinama

Tijekom posljednjih dvadeset godina, razvoj i primjena 3D modela gradova postali su važan čimbenik u planiranju urbanih prostora, regulaciji prometa, zaštiti okoliša i razvoju turističkih strategija. Kompleksna urbana okruženja zahtijevaju prikupljanje, pohranu i obradu velikih količina prostornih podataka, a 3D modeli gradova pokazali su se kao idealno rješenje za objedinjavanje tih informacija. Takvi modeli nalaze svoju primjenu u analizama urbanog prostora (Esch et al., 2020), detekciji promjena u urbanističkoj morfologiji (J. Qin et al., 2015), upravljanju infrastrukturom i gradskim četvrtima (Alomía et al., 2021; da Silveira & Musse, 2006), poticanju i planiranju turizma (Maim et al., 2007), održavanju i digitalizaciji katastra (Tekavec et al., 2020), kartografiji i prostornoj vizualizaciji (Fletcher et al., 2010), arhitektonskom i urbanističkom dizajnu (Buyuksalih et al., 2019; Kim et al., 2018; Yang & Delparte, 2022) i sl. Za detaljan pregled brojnih primjena 3D modela gradova, preporučuje se URL 14.

4.1. Uloga 3D modela u kartografiji i prednosti procedurnog pristupa

Procedurno modeliranje predstavlja napredan pristup generiranju trodimenzionalnih objekata i krajolika, temeljen na korištenju algoritamski definiranih pravila i postupaka. Umjesto ručnog modeliranja svakog pojedinog detalja, ovaj pristup omogućuje automatsko stvaranje složenih struktura kroz definirane parametre i nizove pravila (Ullrich et al., 2010). Takva metodologija osigurava značajnu učinkovitost i fleksibilnost, omogućujući stvarateljima da brzo prilagođavaju modele i eksperimentiraju s raznim varijacijama. Zahvaljujući svojoj prilagodljivosti, procedurno modeliranje koristi se u brojnim područjima, uključujući filmsku industriju, razvoj računalnih igara, arhitektonsku vizualizaciju i urbano planiranje (Watson et al., 2008). Njegova ključna prednost je sposobnost stvaranja velikih i kompleksnih virtualnih okruženja uz minimalan ručni rad. Ova tehnologija omogućuje i dinamično mijenjanje struktura u stvarnom vremenu, što značajno olakšava razvoj iterativnih rješenja. Primjene procedurnog modeliranja uključuju generiranje realističnih krajolika u igrama (Olsen, 2004; Smelik et al., 2014b; Watson et al., 2008), konstrukciju virtualnih gradova za potrebe kinematografije (Parish & Müller, 2001), precizno modeliranje zgrada za arhitekturu i urbanizam (Schwarz & Müller, 2015b), te čak i za potrebe arheologije (Haegler et al., 2009; Müller, Vereenoghe, et al., 2006). Procedurno modeliranje predstavlja napredan pristup generiranju trodimenzionalnih objekata i krajolika, temeljen na korištenju algoritamski definiranih pravila i postupaka. Umjesto ručnog

modeliranja svakog pojedinog detalja, ovaj pristup omogućuje automatsko stvaranje složenih struktura kroz definirane parametre i nizove pravila (Ullrich et al., 2010). Takva metodologija osigurava značajnu učinkovitost i fleksibilnost, omogućujući stvarateljima da brzo prilagođavaju modele i eksperimentiraju s raznim varijacijama. Zahvaljujući svojoj prilagodljivosti, procedurno modeliranje koristi se u brojnim područjima, uključujući filmsku industriju, razvoj računalnih igara, arhitektonsku vizualizaciju i urbano planiranje (Watson et al., 2008). Njegova ključna prednost je sposobnost stvaranja velikih i kompleksnih virtualnih okruženja uz minimalan ručni rad. Ova tehnologija omogućuje i dinamično mijenjanje struktura u stvarnom vremenu, što značajno olakšava razvoj iterativnih rješenja. Ova tehnika također omogućuje stvaranje prirodnih formacija poput šuma, planinskih lanaca i riječnih tokova, reproducirajući njihove specifične geometrijske karakteristike (Gasch et al., 2020; Smelik et al., 2014b). Primjene procedurnog modeliranja uključuju generiranje realističnih krajolika u igrama (Olsen, 2004; Smelik et al., 2014b), konstrukciju virtualnih gradova za potrebe kinematografije (Parish & Müller, 2001), precizno modeliranje zgrada za arhitekturu i urbanizam (Schwarz & Müller, 2015b), te čak i za potrebe arheologije (Haegler et al., 2009; Müller, Vereenoghe, et al., 2006).

Razvoj tehnologije LiDAR (Light Detection and Ranging) omogućio je brzo i precizno prikupljanje prostornih podataka velikih područja. Ovi podaci služe za stvaranje visoko detaljnih 3D modela objekata, posebice zgrada, što je otvorilo nove mogućnosti u automatizaciji urbanog modeliranja (Bassier & Vergauwen, 2020; Buyuksalih et al., 2019; Chen et al., 2018). Zahvaljujući napretku u fotogrametriji i daljinskom očitavanju, danas je moguće prikupiti oblake točaka visoke razlučivosti koji omogućuju automatsku izradu zgrada s višim razinama detalja (LOD). Međutim, obrada i vizualizacija ovih podataka još uvijek predstavlja izazov. Nadalje, usvojeni su pristupi temeljem RANSAC algoritma (Tarsha-Kurdi et al., 2008), koji detektira pravilne geometrijske oblike u oblacima točaka, dok dodatno integriraju semantičku analizu površina uz pomoć topoloških grafova u radu (Jarzabek-Rychard & Borkowski, 2016). U radu (F. Wu et al., 2014b) predstavljaju inverzno procedurno modeliranje pročelja zgrada pomoću podijeljenih gramatičkih pravila. Ostali primjeri mogu se vidjeti na URL 14.

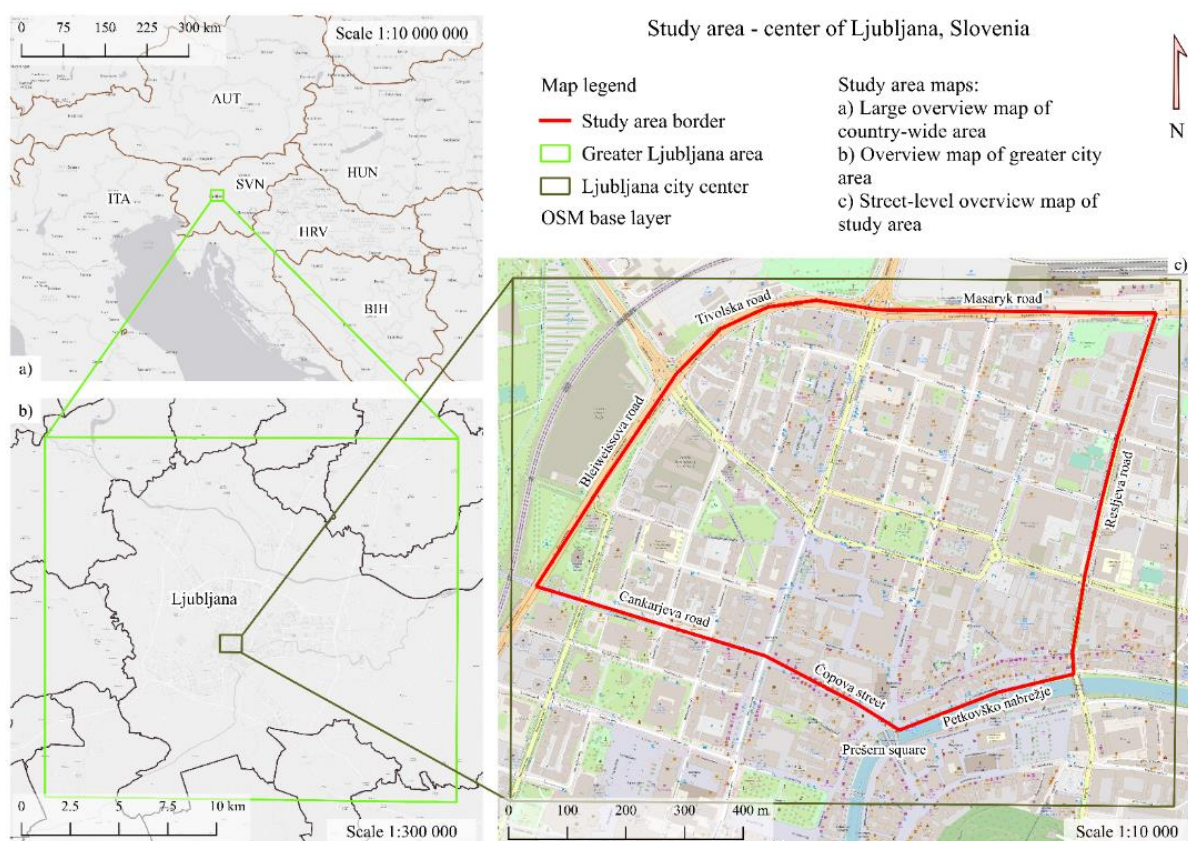
4.2. Područje istraživanja, podaci i metodologija

Izabrano je područje koje karakterizira raznolika struktura zgrada različitih namjena, visina i arhitektonskih stilova, što ga čini idealnim za testiranje automatskih metoda izrade 3D modela. Odabrano je središnje urbano područje Ljubljane zbog dostupnosti podataka, glavnog grada

Slovenije, koji broji približno 300.000 stanovnika i prostire se na površini od 164 km² (URL 15). Prednost izbora tog područja leži u činjenici da su LiDAR podaci za cijelu Sloveniju javno i besplatno dostupni.

4.2.1. Ograničeno područje

Odabrani segment grada omeđen je glavnim ulicama (Slika 43). LiDAR podaci preuzeti su s web portala slovenskog Ministarstva za okoliš, u rezoluciji od 5 točaka po m². Dostupni su u tri varijante, kao OTR —(Oblak Točaka Reljefa) georeferencirani oblak reljefnih točaka koji sadrži samo točke klasificirane na tlu (format pohrane je zLAS), GKOT —(Georeferencirani Klasificirani Oblak Točaka) georeferencirani i klasificirani oblak točaka, koji uključuje točke s tla, zgrade i tri različite vrste vegetacije (format pohrane je zLAS) i DEM (digitalni elevacijski model (DEM), koji je interpolacija reljefa na temelju OTR točaka), pohranjen u pravilnom gridu 1 m × 1 m u obliku ASCII datoteke.



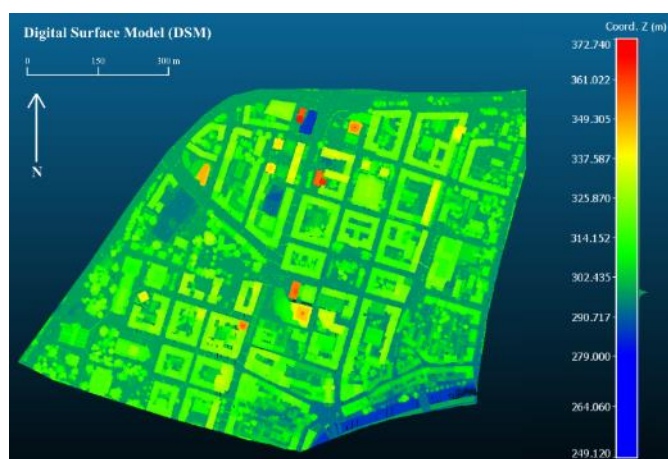
Slika 43. Granica analiziranog područja u središtu Ljubljane, Slovenija, prikazana je crvenom linijom (© OpenStreetMap contributors).

4.2.2. Obrada podataka

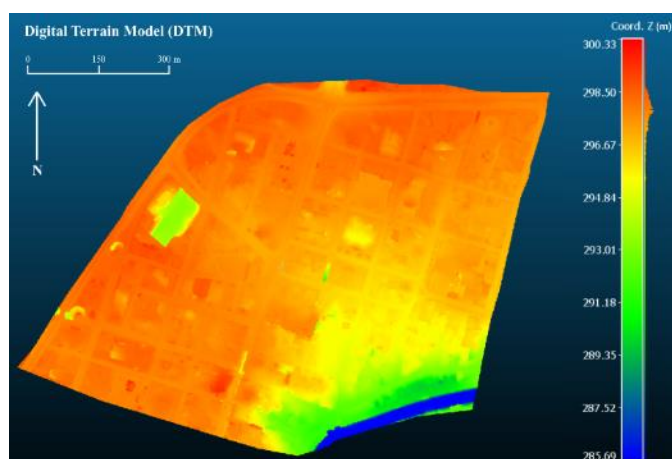
Za obradu su preuzete četiri prostorne jedinice (pločice) koje pokrivaju cijelo analizirano područje: TM_461_101, TM_461_102, TM_462_101 i TM_462_102. Svaka pločica sadržavala je georeferencirani oblak točaka s klasifikacijom (GKOT) i digitalni model reljefa (DEM). Podaci su dostupni u tri oblika: OTR (samo reljefne točke), GKOT (kompletno klasificirani oblak točaka) i DEM (interpolirani raster reljefa). Obrada je provedena kombinacijom alata CloudCompare (URL 16) i QGIS (URL 17). U CloudCompare-u, fragmentirani dijelovi klasificiranog oblaka točaka i DEM učitani su odvojeno, spojeni u cjelinu i grubo izrezani na promatrano područje. Rezultat su dvije osnovne komponente:

- Digitalni model površine (DSM) s ukupno 7.627.691 točkom
- Digitalni model terena (DTM) sa 721.807 točaka (Slika 44)

Nakon početne obrade, podaci su dodatno filtrirani tako da ostanu samo točke koje pripadaju objektima (ponajprije zgradama). Za njihovo točno pozicioniranje i identifikaciju korišteni su podaci iz OpenStreetMap-a (OSM), preuzeti kroz QGIS (URL 17) dodatak OSM Downloader (URL 18). Dobiveni višepoligonski slojevi (shapefile) ručno su pročišćeni tako da ostanu samo građevine, a zatim su im pridružene visine. Visina svake zgrade određena je kao razlika između DSM-a i DTM-a. Dodatno, korišten je alat Zonal Statistics za izračun prosječne, minimalne i maksimalne visine po objektu. Ova analiza omogućila je preciznije razlikovanje stvarnih visina krova od ekstremnih vrijednosti uzrokovanih antenama, tornjevima ili složenim arhitektonskim elementima. Budući da OSM ne sadrži pouzdane informacije o tipovima krovova, oblik krova je za većinu objekata ručno određen pomoću satelitskih snimki iz Google Eartha. Rezultat obrade OSM podataka su 453 objekta koji predstavljaju zgrade. Atributna tablica sadrži ukupno 25 stupaca s podacima o nazivu i vrsti objekta, geološkim značajkama, površini, korištenju zemljišta, kao i polja s nazivima 'craft', 'leisure', 'man-made', 'military', 'mjesto', 'trgovina', 'druge oznake' i drugi opisi objekata koji su zanemareni u ovom istraživanju (Slika 45). Za više detalja i objašnjenja o postupku pogledati na URL 19, a procedurnu skiptu na URL 20.

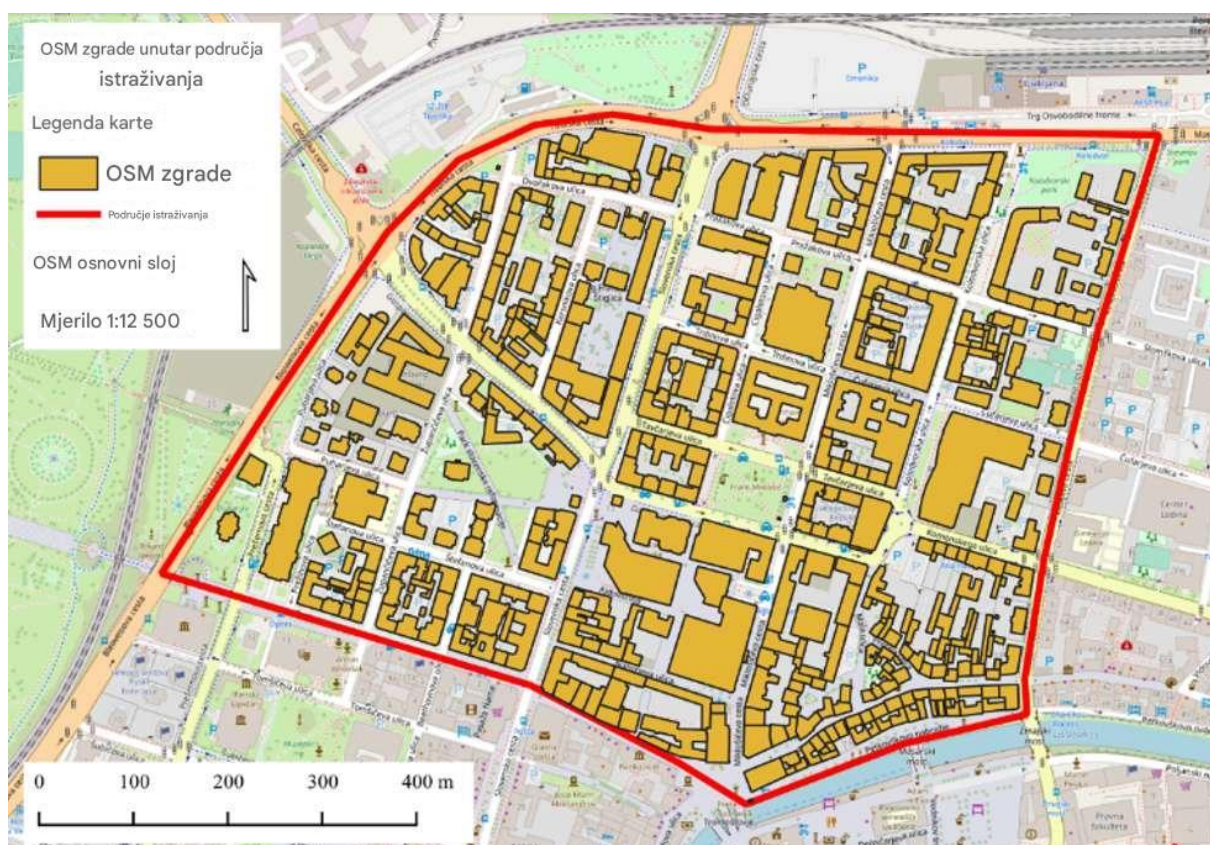


(a)



(b)

Slika 44. Izlazni podatci (elevacija) obrade oblaka točaka u CloudCompare: (a) DSM; (b) DTM (URL 19).



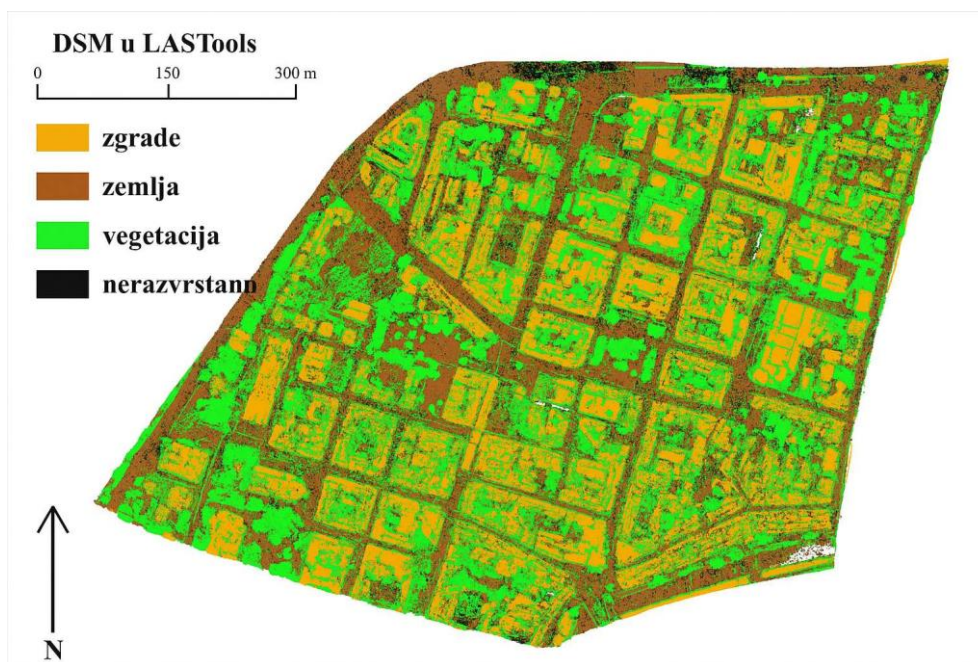
Slika 45. Sloj Shapefile s tlocrtima zgrada nakon filtriranja OSM podataka

(© *OpenStreetMap contributors*).

Sljedeći ključni korak u mojem istraživanju bio je unos visinskih vrijednosti za svaku pojedinu zgradu u odgovarajuće atributno polje. Za tu sam svrhu koristio LASTools – snažan skup alata namijenjen obradi LiDAR podataka (URL 21). Kao početni korak, oblake točaka koji su

predstavljali digitalni model terena (DEM) i klasificirani oblak točaka u *las* formatu pretvorio sam u rasterske datoteke. Vrijednosti piksela u tim datotekama odražavale su Z-koordinate iz oblaka točaka.

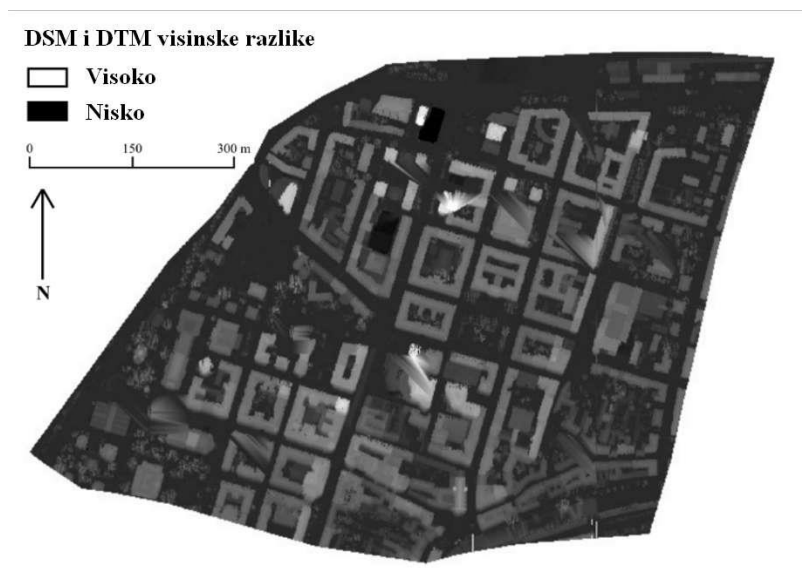
Unutar QGIS okruženja instalirao sam dodatak LAsTools (URL 21), čime mi je postalo omogućeno korištenje tih alata putem Processing Toolboxa. Pomoću modula Lasview generirao sam DSM (Digital Surface Model), odnosno prikaz 3D koordinata koji mi je služio za daljnju segmentaciju točaka (Slika 46).



Slika 46. Klasificirani DSM u LAsTools-u (URL 19).

U sljedećem koraku upotrijebljen je Lassplit i Las2demPro. Detaljni koraci objašnjeni su na URL 22. Rezultirajući isječci konvertirani su u rasterski format. Za izradu digitalnog modela terena (DEM) koristio sam verziju alata koja omogućuje obradu samo jedne datoteke jer u ovom slučaju nije bilo potrebe za dijeljenjem oblaka točaka na manje segmente. Na taj sam način dobio jednu rastersku sliku za DEM i po jednu sliku za svaki od šest klasificiranih isječaka oblaka točaka.

Svjetliji pikseli na slici označavali su veće visine. Da bih dobio visine zgrada, jednostavno sam oduzeo vrijednosti piksela iz DEM-a od odgovarajućih vrijednosti u klasificiranom oblaku točaka. Dobivena slika s prikazom rezultata prikazana je na Slika 47.



Slika 47. Visinske razlike između DSM i DTM (URL 19).

Tijekom pripreme analize imao sam mogućnost filtriranja i odabira statističkih pokazatelja koje želim izračunati. U ovom sam slučaju odlučio uzeti minimalnu, maksimalnu i srednju vrijednost visine kako bih mogao što preciznije razlikovati stvarnu visinu krova od povećanih vrijednosti uzrokovanih antenskim strukturama, dimnjacima i sličnim elementima koji se često nalaze na krovovima. Dobivene statistike automatski su pridružene zgradama kao atributi unutar njihove tablice. Osim same visine, shvatio sam da je za kvalitetno automatsko modeliranje zgrada ključno prepoznati i oblik krova. Budući da je u atributima OpenStreetMap podataka samo 25 od ukupno 453 zgrade imalo definiran tip krova, morao sam za sve ostale zgrade ručno nadopuniti to polje. U tu svrhu koristio sam Google Earth, gdje sam pregledom krovova vizualno identificirao njihov oblik i unio odgovarajuće vrijednosti u atributnu tablicu za što vjerniju geovizualizaciju.

4.2.3. Softverski alati

Za potrebe istraživanja ispitani su različiti softveri koji se koriste za automatsko modeliranje zgrada na temelju LiDAR podataka, pri čemu je svaki softver ocijenjen (Tablica 3) s obzirom na njegove prednosti i nedostatke.

Tablica 3. Izbor softvera

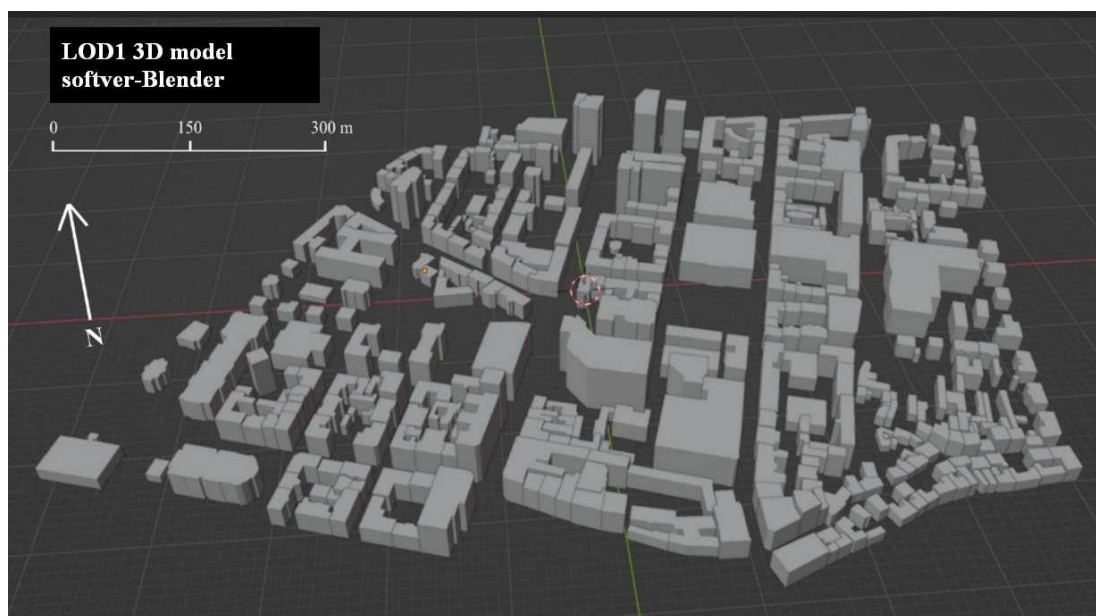
Softver	Prednosti	Nedostaci	URL
Houdini	Napredno procedurno modeliranje, podrška za LIDAR i shapefile i besplatna verzija Apprentice	Kompliciran za učenje i dugotrajna obuka	URL 23
CityEngine	Integracija s GIS-om, prepoznavanje krova iz LIDAR-a i vizualizacija u stvarnom vremenu	Ograničen za detaljno modeliranje, cijena ~100 USD/god i ograničeni edukacijski resursi	URL 24
Unity + CityGen3D	Jednostavno sučelje, nema potrebe za kodiranjem, podrška za OSM, DEM, teksture, brza urbana vizualizacija	Nema uređivanja atributa zgrada, slabija preciznost i cijena dodatka 125 USD	URL 25
Maya	Intuitivno sučelje, podrška za shapefile i LIDAR	Nedostatak edukacije za 3D modele	URL 26
Blender	Besplatan, open-source, geometry nodes za proceduralno modeliranje, velika zajednica i dokumentacija i minimalni zahtjevi	Potreban BlenderGIS dodatak za koordinate	URL 27
Geopipe	Automatizirano modeliranje, visoka razina detalja, kombinira LIDAR i satelitske podatke	Nepoznate cijene, ograničena kontrola nad modelima	URL 28
Omniverse	Suradnja u stvarnom vremenu, visoka vizualna kvaliteta (RTX) i sinkronizacija alata i timova	Namijenjen profesionalcima, fokus više na simulacije nego na GIS	URL 29
Mapbox	Napredne kartografske mogućnosti, prilagodljive interaktivne mape i lokacijska inteligencija	Fokusiran na 2D/karte, a ne na 3D modeliranje zgrada	URL 30
Cesium	Interaktivni 3D prikaz gradova, koristan za urbanizam i obranu i podrška za 3D Tiles	Nema mogućnosti izrade modela, ograničena dodjela semantičkih podataka	URL 31

Konačni izbor je Blender – besplatan softver otvorenog koda koji omogućuje 3D modeliranje, animaciju i renderiranje. Blender sam odabrao zbog njegove dostupnosti, niskih hardverskih

zahtjeva, bogate online dokumentacije, te podrške za OSM podatke. Posebno mi je bio koristan sustav Geometry Nodes, uveden u verziji 2.92, koji omogućuje napredno procedurno modeliranje geometrije. Korištenjem tog alata mogao sam automatizirati velik dio postupka.

4.3. Primjena u *Blenderu*

BlenderGIS mogu besplatno preuzeti s GitHuba (URL 32) i široko je primjenjiv za integraciju Blendera s prostornim (geografskim) podacima. Među funkcionalnostima koje nudi nalaze se operatori poput Basemaps, Get OSM i Get SRTM, koji omogućuju izravno preuzimanje OGC web karata, OSM podataka te SRTM (DEM podaci iz NASA-e). Osim toga, BlenderGIS olakšava uvoz i izvoz shapefile slojeva, georeferenciranih slika i OSM podataka, a dostupne su i koordinatne projekcije što je važno za precizno postavljanje objekata na teren, odnosno u realne koordinate. Prilikom uvoza shapefile datoteke kroz BlenderGIS, odabrao sam atributno polje koje sadrži podatke o visini, u ovom slučaju visinu zgrade izračunatu u prethodnim koracima. Nije korištena opcija offseta u odnosu na vrijednost atributnog polja, jer će objekti naknadno biti postavljeni na referentnu površinu. Na kraju sam podijelio objekte u pojedinačne cjeline ili entitete te izabrao odgovarajući koordinatni sustav. Kao rezultat, dobio sam uvezeni 3D model s visinama i tlocrtima koji odgovaraju stvarnim zgradama (Slika 48).



Slika 48. 3D model zgrade uvezen u Blender (URL 19).

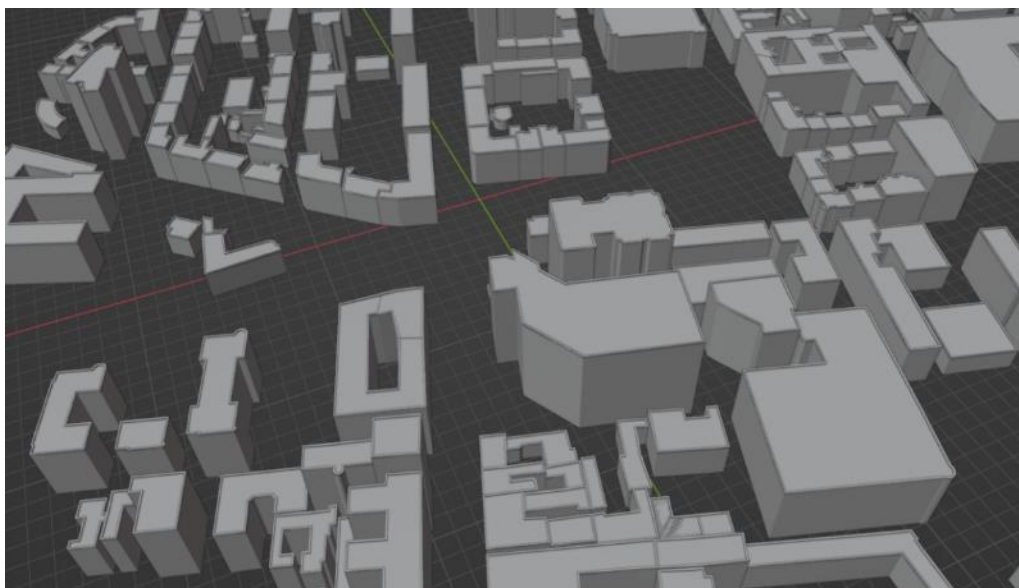
Od brojnih oblika krovova prisutnih u arhitekturi, na području koje sam analizirao identificirao sam tri tipa krovova (Slika 49): ravne, šatorske (ili kose) te dvostrešne (ili dvovodne). Šatorski

i kosi krovovi svrstavani su u jednu kategoriju jer se geometrijska pravila za njihovu izradu ne razlikuju, a konačni oblik ovisi samo o tlocrtu – šatorski se krovovi javljaju na kvadratnim tlocrtima, dok su kosi karakteristični za pravokutne. Ravni krovovi najzastupljeniji su na promatranom području, ukupno ih je 286. Uz njih, evidentirao sam 150 šatorskih ili kosih krovova, dok je dvostrešnih bilo samo 17 u toj klasifikaciji.



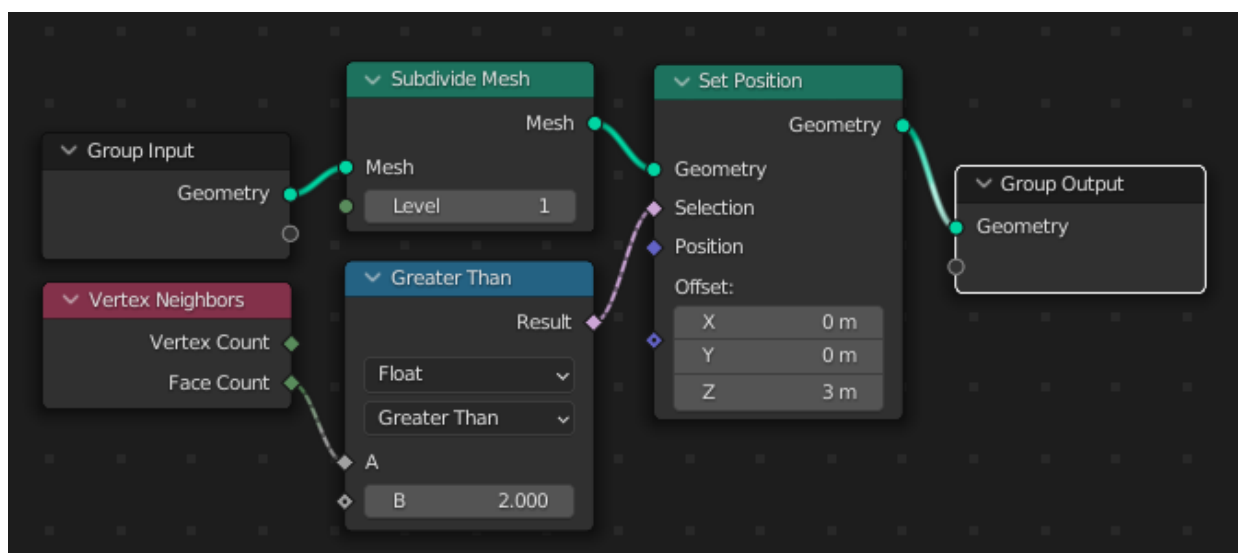
Slika 49. Vrste krovova na analiziranom području (URL 19).

Zbog jednostavne geometrije ravnih krovova, za njihovo modeliranje postupio sam na sljedeći način: (a) odabrao sam svih 286 zgrada s ovom vrstom krova; (b) odvojio sam krovove od zidova koristeći mod za uređivanje; (c) spremio sam ih u zasebne slojeve; te (d) otvorio sam sučelje čvorova Geometry i modelirao zidove visine 1,3 metra tako da sam izdubio krov prema temeljima zgrade za 1,3 metra. Zidovi svih zgrada u ovoj kategoriji imaju debljinu od 20 centimetara kako bi pročelja bila što realističnija (Slika 50).

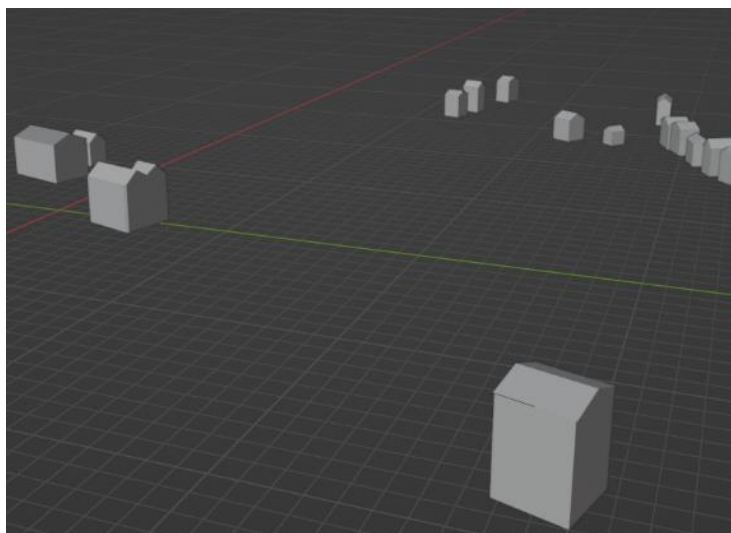


Slika 50. Rezultat pravila primijenjenih na ravne krovove (URL 19).

Kod modeliranja dvostrešnih krovova podijelio sam površinu na dva dijela tako što sam odabrao čvorišta koja povezuju dvije plohe i podigao ih za 3 metra. Ovaj postupak izveo sam kombinacijom nekoliko modifikatora unutar sučelja Geometry Nodes (Slika 51), što je rezultiralo jednostavnim modelom s fiksnom visinom sljemena krova (Slika 52). Kako bi model bio što vjerniji stvarnim objektima, dodatno sam prilagodio visinu krova u skladu s maksimalnim i prosječnim vrijednostima visine iz tablice atributa (Tablica 4).

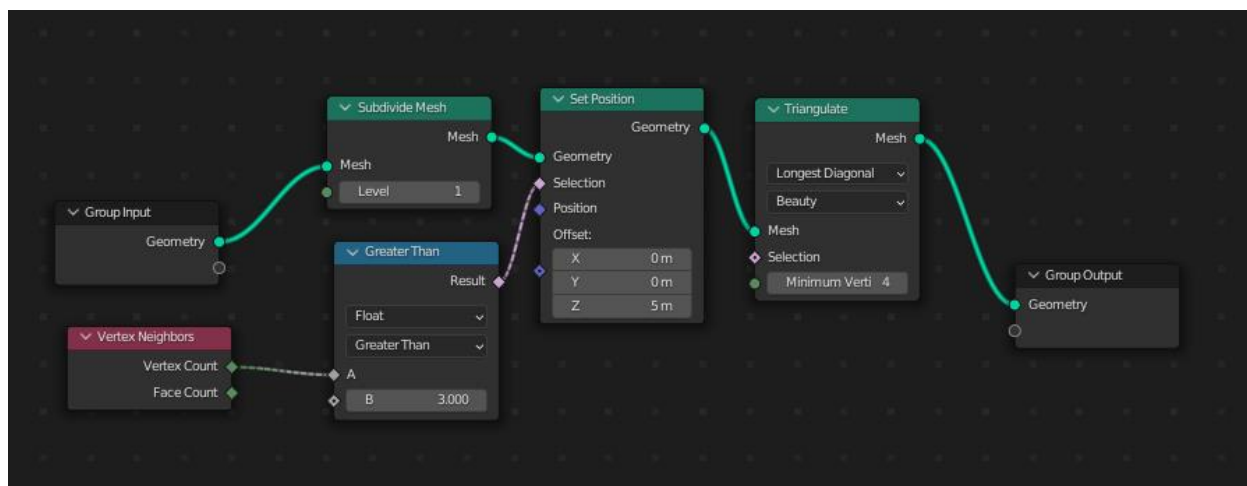


Slika 51. Postavke unutar sučelja geometrijskih čvorova za dvostrešne krovove (URL 19).



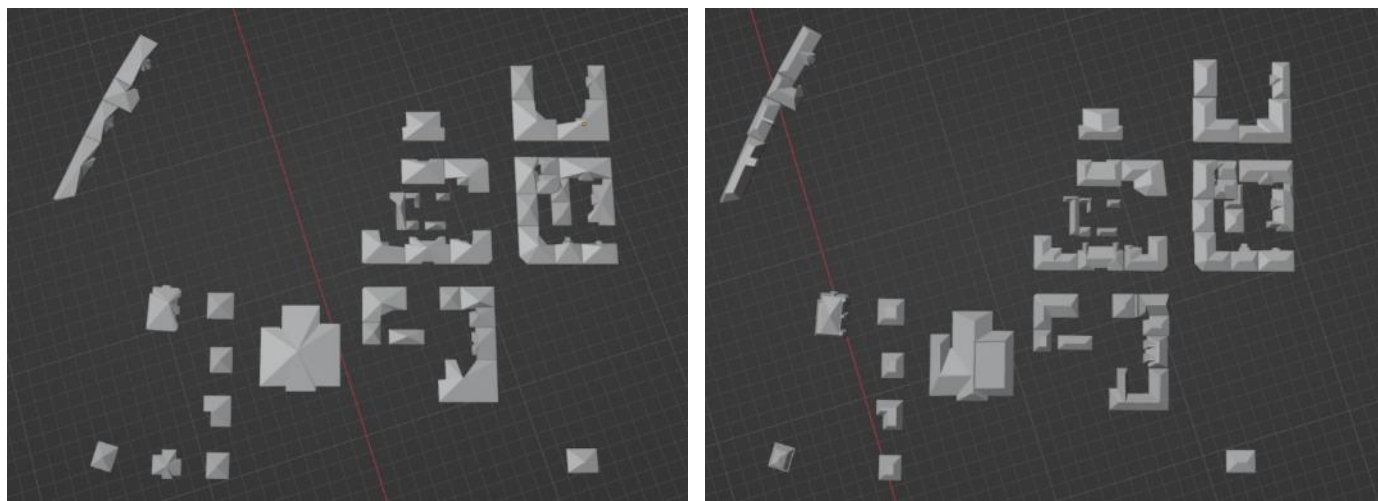
Slika 52. Rezultat matematičkih pravila primijenjenih na dvostrešne krovove (URL 19).

Za nagnute i nagnute krovove korišten je modifikator površinske triangulacije unutar sučelja čvorova Geometry (Slika 53).



Slika 53. Postavke unutar sučelja geometrijskih čvorova za šatorske i kose krovove (URL 19).

Jednostavno i prihvatljivo rješenje za zgrade čiji tlocrt nema izražene izbočine ili izdužene dijelove je triangulacija ploha i određivanje kuteva rubova prema zadanim kriterijima. Međutim, kod zgrada s izbočenim dijelovima pročelja, triangulacija je rezultirala prevelikim rubovima krova (Slika 54a), zbog čega sam morao ručno urediti te rezultate (Slika 54b).

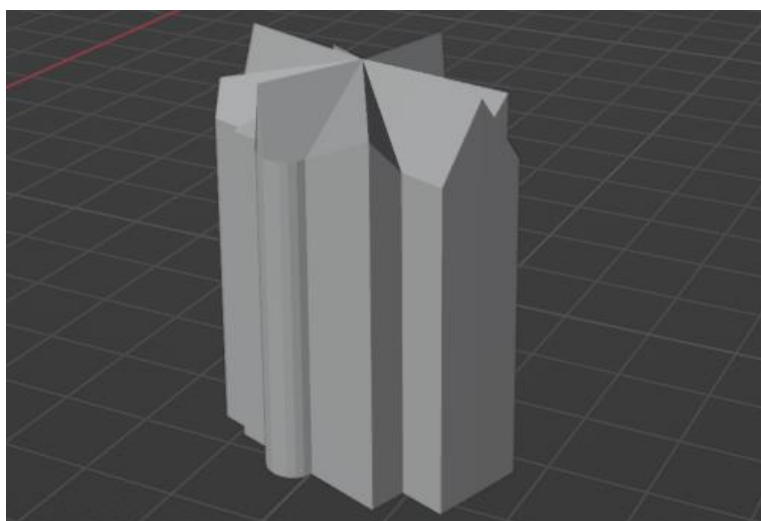


(a)

(b)

Slika 54. Vizualizacija šatorskih i kosih krovova u blenderu nakon **(a)** primjene matematičkih pravila; **(b)** ručnog uređivanja (URL 19).

U skupinu zgrada s dvostrešnim krovovima spadaju sakralne građevine složenih pročelja, tlocrta i krovova; pa ih je bilo potrebno ručno modelirati korištenjem funkcija moda za uređivanje sučelja (Slika 55).



Slika 55. Vizualizacija sakralne građevine u Blender-u (URL 19).

4.4. Izrada i validacija 3D modela zgrada

Korištenjem postupaka modeliranja opisanih u prethodnom poglavlju, izrađen je 3D model zgrada u širem središtu Ljubljane na LOD2 razini detalja (Slika 56). Dobiveni rezultati uspoređeni su s odgovarajućim dijelom trodimenzionalnog prikaza u Google Earthu.



Slika 56. Konačni 3D model u LOD2 (URL 19).

Vizualnom usporedbom utvrdio sam da modeli zgrada uglavnom odgovaraju stvarnom položaju, obliku i visini prikazanima u Google Earthu. Međutim, detaljnijom analizom primijetio sam da pojedine zgrade odstupaju u visini u odnosu na Google Earth modele. Posebno su sakralni objekti pojednostavljeni u strukturi i krovnom skeletu u usporedbi s modelima u Google Earthu. Razlog tome je što su zvonici crkava znatno viši od okolnih građevina, što utječe na LiDAR mjerenja iz zraka. Stoga srednja vrijednost visine koju dobijem za ovu vrstu zgrada korištenjem alata Zonal Statistics ima tendenciju precijeniti visinu krova cijele zgrade (Slika 57).



(a)



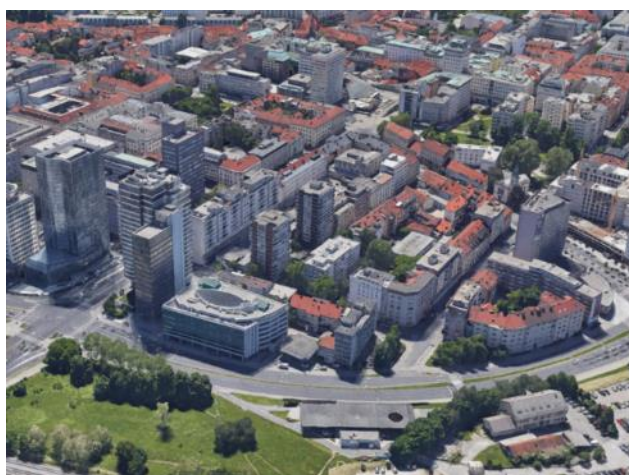
(b)

Slika 57. Usporedba dvaju prikaza za sakralni objekt u (a) Blender-u; (b) Google Earth-u (URL 19).

Drugu usporedbu proveo sam za poslovnu četvrt Ljubljane koristeći Google Earth (Slika 58). Kao što sam i očekivao, 3D model je vrlo usko usklađen s modelom u Google Earthu, osobito u ovom dijelu područja istraživanja. Ovo blisko vizualno podudaranje mogu pripisati jednostavnom i suvremenom dizajnu poslovnih zgrada, koje karakteriziraju brojne ravne površine poput fasada i krovova. Model sam izradio koristeći postojeće podatke i primijenjenu metodologiju, koliko je to bilo moguće. Budući da nisu dostupni detaljni obrisi objekata koji bi omogućili razdvajanje na više segmenata, nije bilo moguće detaljnije modelirati vjerski objekt u 3D. Šatorski i kosi krovovi svrstani su u istu kategoriju jer je proces njihovog modeliranja identičan, pa nije bilo potrebe za razlikovanjem. Velika zgrada iza vjerskog objekta također se vizualno razlikuje od modela prikazanog na Slika 57b; no razumijevanjem istraživačkog procesa i nužnih ustupaka zbog automatizacije pojedinih elemenata, poput visine zgrada i ograničenja LOD2 razine detalja, jasno je da je u procedurnom modeliranju potrebno napraviti određene kompromise u vizualnoj preciznosti. To proizlazi iz same definicije procedurnog modeliranja, koje ne mora dati savršeno vjeran prikaz objekta, već približan i funkcionalan prikaz.



(a)



(b)

Slika 58. Prikaz poslovne četvrti Ljubljane, prikazan u (a) Blender-u; (b) Google Earth-u (URL 19).

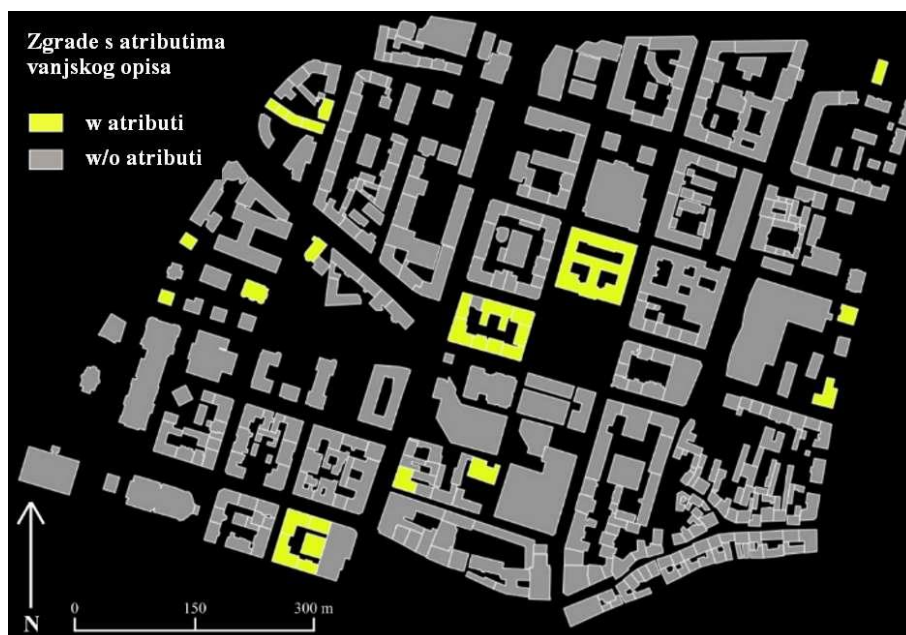
4.5. Ograničenja modeliranja i analiza podataka

Upravo su OSM podaci uzrokovali najviše nedosljednosti u konačnom modelu. Dostupnost i prostorna pokrivenost OSM podacima za područje istraživanja pokazala se vrlo dobrima, što sam i očekivao. Međutim, prilikom obrade OSM podataka za tablicu atributa, primijetio sam da samo 82 od ukupno 453 zgrade (Slika 59) imaju jedan ili više atributa koji opisuju vanjski izgled zgrade, poput ukupnog broja katova, boje fasade, vrste krova ili materijala krova.



Slika 59. Građevine s atributima u OSM-u koji opisuju vanjski izgled (narančasto) (URL 19).

Samo 28 od 453 građevine imalo je tip krova (Slika 60). Za preostale zgrade treba ručno unijeti vrstu krova u tablicu atributa.



Slika 60. Zgrade u OSM-u koje imaju atribut tipa krova (žuto) (URL 19).

Nakon što sam dovršio pripremu podataka i njihov unos u Blender, većinu tipova krovova uspio sam jednostavno generirati pomoću jedne od tri metode konstrukcije krova. Za 397 krovova,

što čini 87,64 % ukupnog broja, nije bilo potrebe za ručnim doradama geometrije. Takvi automatski generirani krovovi označeni su zelenom bojom na karti prikazanoj na Slika 61.



Slika 61. Pokrivenost zgrada s automatski generiranom geometrijom krova (zeleno) i geometrijom krova s potrebnim ručnim poboljšanjem (crveno) (URL 19).

S obzirom na to da čak 93,82 % zgrada u području istraživanja nije imalo podatke o tipu krova, kao vrijedan pristup za budućnost razmatram automatizaciju klasifikacije oblika krova putem strojnog učenja. Jedan od pristupa opisan je u radu (Castagno et al., 2018), gdje je postignuto automatsko prepoznavanje tipova krovova za 4500 zgrada u tri grada. Važno je napomenuti da određene vrste zgrada, poput komercijalnih ili uredskih prostora, mogu imati više stropove za smještaj dodatne infrastrukture (URL 33). Uz uobičajene vrste krovova kao što su šatorasti, kosi i polukosi, algoritam prepoznaje i ravne krovove s infrastrukturom za klimatizaciju kao posebnu kategoriju. Ulazni podaci uključuju LiDAR slike, obrise zgrada i satelitske snimke. Satelitske slike su dobar izvor informacija, no zbog njihove dvodimenzionalne prirode podložne su sjenama, smanjenom kontrastu i izobličenju perspektive. LiDAR podaci se koriste komplementarno jer omogućuju određivanje dubine i volumena krovova. Obrisi zgrada služe za filtriranje dijelova LiDAR i satelitskih podataka koji odgovaraju pojedinoj zgradi. Obrise sam izdvojio korištenjem ortofoto snimaka ili ih preuzeo s OSM-a, kao u ovom istraživanju. Sva tri sloja trebaju biti georeferencirana i imati isti koordinatni sustav kako bi se mogli integrirati. Rezultat je skup klasificiranih vrsta krovova s točnošću od 87 %. Ova metoda značajno bi ubrzala snimanje tipova krovova na velikim površinama i omogućila precizniju klasifikaciju nego što to omogućava vizualni pregled putem Google Eartha. Za modeliranje kosih krovova koristio sam jednostavne matematičke operacije koje transformiraju ravnu

površinu, prateći rub zgrade, u 3D oblik. Rezultat su jednostavne konstrukcije koje uglavnom dobro prikazuju odgovarajući krov. Za složenije krovove sa kompleksnim tlocrtnim oblicima i kosturima, bolje bi rješenje bilo korištenje Polyskela, planarnog algoritma za konstrukciju kostura razvijenog u Pythonu. Slovačka tvrtka Prochitecture razvila je biblioteku Bypolyskel s algoritmima za konstrukciju kostura i površina (URL 34). Za korištenje ove biblioteke trebalo bi programirati vlastiti dodatak za Blender koji bi omogućio integraciju koda u model. Glavni nedostatak istraživanja leži u ručnom uređivanju OSM podataka tijekom pripreme atributnih podataka za automatsko modeliranje. Optimalno bi bilo automatizirati unos nedostajućih podataka pomoću gore spomenutih metoda za detekciju tipova krovova. Također, postignuta razina detalja je LOD2, što nije dovoljno za pohranu nekih semantičkih informacija. Na primjer, u urbanističkoj analizi često se mjeri i bilježi vidljivost određenih zgrada i dijelova grada s određenih prozora, što nije moguće bez obilježja pročelja. Za to bi trebalo primijeniti strojno učenje na velik broj fotografija fasada karakterističnih za područje, tipizirati ih, razdvojiti na elementarne dijelove, te izvesti skup pravila prema kojima se ti dijelovi uklapaju u cjelinu i dodjeljuju modeliranim zgradama. Time bi se podigla razina detalja na LOD3, a bilo bi moguće unijeti i nove semantičke podatke, poput energetske učinkovitosti ili plana evakuacije zgrade. Za nastavak rada na 3D modelima preporučujem implementaciju u standarde CityGML ili CityJSON, te dodjelu klasa i semantičkih podataka. Istraživanja ukazuju na važnost korištenja GIS alata za izvlačenje što većeg broja informacija iz LiDAR oblaka točaka, budući da te informacije mogu poslužiti kao temelj za automatski generirane 3D modele u daljnjoj fazi. Stoga smatram da je doprinos ovog rada u jednostavnom pristupu LiDAR podacima i maksimalnom korištenju dostupnih podataka u okviru besplatnog softvera otvorenog koda. Opisani ručni koraci uključuju i automatizirane dijelove koji su povezani i izvršavaju se automatski, što je posebno vidljivo u programiranim segmentima na Pythonu, dostupnima na (URL 20).

4.6. Diskusija

U ovom sam poglavlju demonstrirao proces procedurnog modeliranja velikog urbanog područja koristeći besplatno dostupne LiDAR oblake točaka i OSM podatke. Ključni izazov bio je u pripremi podataka s ciljem automatizacije modeliranja. Uz LiDAR oblake, priprema je uključivala segmentiranje oblaka za izdvajanje dijelova relevantnih za zgrade, spajanje različitih oblaka u jedinstveni skup te filtriranje buke i nepotrebnih točaka koje nisu povezane sa zgradama. Kada su u pitanju OSM podaci, morao sam procijeniti kvalitetu korisnički

unesenih atributa, ukloniti irelevantne podatke i dopuniti nedostajuće informacije specifične za zgrade. Ti atributi imaju važnu ulogu u kasnijim fazama 3D modeliranja. Relevantni atributi za modeliranje i vizualizaciju mogu se unositi ručno ili automatski uz pomoć strojnog učenja, a razina moje intervencije ovisila je o dostupnosti i kvaliteti podataka. Za manje površine odgovarajuća je kombinacija ručnog i poluautomatskog pristupa koja omogućava veću detaljnost i realističniji prikaz, dok za veća područja preferiram procedurno modeliranje (Župan et al., 2018) jer smanjuje potrebu za ručnim radom i štedi vrijeme, a istovremeno generira topološki dosljedne modele koje se može naknadno modificirati. LiDAR oblaci prikupljeni zračnim snimanjem pružaju dobru osnovu za konstrukciju modela, naročito u kombinaciji s topološkim podacima i fotografijama. Iako postoje provjerene metode za detaljnu rekonstrukciju objekata iz zraka, primjena procedurnog 3D modeliranja na velike urbane površine, poput gradova ili država, još je ograničena. S obzirom da zračni LiDAR podaci imaju nižu rezoluciju nego terenski, modeli koje sam dobio djelomično automatiziranim pristupom imaju vrlo dobru točnost, unatoč razini detalja koju dopuštaju podaci. Istraživanje sam usmjerio na korištenje dostupnih zračnih LiDAR podataka i OSM prostornih podataka za generiranje 3D modela zgrada u središtu Ljubljane. Ovu sam lokaciju odabrao zbog dostupnosti podataka te raznolikosti oblika, visina i funkcija zgrada, što je predstavljalo izazov za automatizaciju. Visine zgrada izdvojio sam iz kombinacije OSM atributa, georeferenciranih i klasificiranih LiDAR oblaka točaka, te digitalnog modela visine (DEM) dobivenog od slovenske Agencije za okoliš. Iz LiDAR podataka sam proizveo digitalni model površine (DSM) i model terena (DTM). Obrise i attribute zgrada izvukao sam iz OSM-a koristeći QGIS. Prostorna pokrivenost OSM-om je izvrsna, ali samo 18 % zgrada ima attribute za vanjski izgled, a 6 % podatke o tipu krova. Za izračun visina koristio sam softver LASTools. Nakon usporedbe dostupnih softverskih rješenja za procedurno modeliranje, Blender sam odabrao zbog njegove sposobnosti obrade OSM podataka, dokumentacije i niskih hardverskih zahtjeva. Procedurno modeliranje na razini LOD1 u potpunosti sam automatizirao, dok je za LOD2 modeliranje krova automatizirano za 87,64 % zgrada. Za ostatak sam usporedio procedurno i ručno modeliranje krova, a potom napravio vizualnu usporedbu s Google Earth modelima. Glavni cilj bio mi je pokazati efikasan proces korištenja otvorenih podataka i besplatnog softvera za dobivanje preciznijih 3D modela nego u ranijim LOD2 iteracijama. Ukratko, procedurno modeliranje je snažna tehnika koja koristi algoritme i pravila za generiranje složenih i realističnih digitalnih sadržaja. Njegova sposobnost stvaranja raznolikih krajolika i modela dovela je do široke primjene u videoigricama, filmskoj produkciji, arhitekturi i urbanom planiranju. S daljnjim

razvojem strojnog učenja, potencijal procedurnog modeliranja je gotovo neograničen, što ga čini vrlo zanimljivim područjem istraživanja u računalnoj grafici i geovizualizaciji.

Tablica 4. Visinske razlike između podataka dobivenih Lidarom i OSM podataka u metrima (URL 14)

OSM ID	Razine izgradnje OSM-a	Visina zgrade OSM m	LiDAR srednja visina zgrade m	LiDAR		
				Visinska razlika m	maksimalna visina zgrade M	Visinska razlika m
2512227	1	2.60	28.4	25.8	37.4	34.8
4089569	4	10.40	21.5	11.1	27.8	17.4
24785631	4	10.40	15.3	4.9	24.2	13.8
24786527	2	5.20	7.2	2.0	9.2	4.0
24786528	14	36.40	38.5	2.1	52.4	16.0
61024039	12	31.20	35.3	4.1	43.9	12.7
111584761	4	10.40	16.4	6.0	29.8	19.4
153059527	13	33.80	10.8	-23,0	16.1	-17.7
176487482	4	10.40	14.4	4.0	17.9	7.5
176487486	4	10.40	20.2	9.8	26.2	15.8
177270343	3	7.80	14.7	6.9	17.4	9.6
179744268	0	0,00	7.3	7.3	20.0	20.0
179744269	8	20.80	22.4	1.6	30.3	9.5
179746610	4	10.40	16.1	5.7	27.8	17.4
186332693	4	10.40	16.8	6.4	19.0	8.6
186335765	15	39.00	54.8	15.8	71.0	32.0
186335774	6	15.60	24.1	8.5	28.9	13.3
186513050	11	28.60	32.4	3.8	37.6	9.0
186513051	11	28.60	34.7	6.1	37.3	8.7
186515083	6	15.60	16.3	0.7	33.3	17.7
186518586	4	10.40	19.8	9.4	23.8	13.4
186518625	4	10.40	19.7	9.3	24.4	14.0
186547097	3	7.80	15.7	7.9	21.5	13.7

186547098	14	36.40	41.0	4.6	44.5	8.1
186547099	2	5.20	14.6	9.4	33.9	28.7
186547100	6	15.60	19.3	3.7	24.3	8.7
186547113	8	20.80	24.7	3.9	29.0	8.2
186547122	10	26.00 sati	24.7	-1,3	40.0	14.0
186547124	14	36.40	41.6	5.2	45.7	9.3
186547129	7	18.20	27.0	8.8	31.4	13.2
186547134	7	18.20	22.6	4.4	27.2	9.0
186547137	2	5.20	9.4	4.2	13.1	7.9
186549962	7	18.20	19.4	1.2	23.9	5.7
196794006	21	54.60	-12.1	-66,7	81.0	26.4
197017824	4	10.40	22.9	12.5	26.1	15.7
235168944	13	33.80	49.8	16.0	60.4	26.6
248803106	3	7.80	16.4	8.6	21.4	13.6
248803110	5	13.00 sati	3.4	-9,6	12.7	-0,3
496313466	3	7.80	15.9	8.1	20.8	13.0
496313467	6	15.60	22.1	6.5	27.2	11.6
778984182	5	13.00 sati	19.1	6.1	24.3	11.3
824372657	7	18.20	23.3	5.1	28.3	10.1
824372661	3	7.80	12.5	4.7	17.6	9.8
936397467	22	57.20	4.4	-52,8	81.0	23.8
976077230	7	18.20	0,2	-18,0	27.0	8.8
976077231	8	20.80	0,2	-20.6	30.0	9.2
976077232	7	18.20	0,1	-18.1	7.9	-10.3
1030934401	2	5.20	13.6	8.4	17.8	12.6
1030934402	2	5.20	11.4	6.2	17.7	12.5
1036836792	3	7.80	14.0	6.2	19.0	11.2
1040050251	4	10.40	18.1	7.7	22.7	12.3
1040050252	4	10.40	20.1	9.7	24.1	13.7
1040050253	4	10.40	20.4	10.0	23.9	13.5
1040050254	4	10.40	18.7	8.3	22.2	11.8
1040050255	4	10.40	19.5	9.1	23.2	12.8
1040050256	4	10.40	19.5	9.1	23.0	12.6

1040050257	4	10.40	18.0	7.6	23.5	13.1
1055837765	4	10.40	20.6	10.2	24.3	13.9
1055837766	4	10.40	20.6	10.2	23.6	13.2
1055837767	4	10.40	20.5	10.1	24.7	14.3
1055837768	4	10.40	19.9	9.5	25.0	14.6
1118759071	4	10.40	15.8	5.4	18.9	8.5
1118759076	4	10.40	12.3	1.9	18.7	8.3
1118759078	4	10.40	8.7	-1,7	22.3	11.9
1118759079	4	10.40	19.8	9.4	22.9	12.5
1118759081	4	10.40	19.1	8.7	22.8	12.4
1118759082	4	10.40	18.8	8.4	22.7	12.3
1118759083	4	10.40	14.5	4.1	23.4	13.0
1118759084	4	10.40	12.0	1.6	19.6	9.2
1118759085	4	10.40	13.5	3.1	21.6	11.2
1120198927	4	10.40	20.3	9.9	24.7	14.3
1120210047	7	18.20	20.8	2.6	23.7	5.5
1120210048	7	18.20	23.8	5.6	25.7	7.5
1120210049	7	18.20	22.2	4.0	24.4	6.2
1121513868	3	7.80	14.1	6.3	17.3	9.5
186547113	8	20.80	23.4	2.6	34.6	13.8
153059527	13	33.80	41.6	7.8	70.6	36.8
153059527	13	33.80	8.0	-25.8	12.4	-21.4

5. Integracija umjetne inteligencije i procedurnog modeliranja u 3D geovizualizaciji zgrada

Procedurno modeliranje nudi značajne prednosti u odnosu na tradicionalne metode geovizualizacije 3D modela zgrada, posebice zbog upotrebe skriptnih jezika ili strojnog koda za opis struktura. Ovaj pristup se odlikuje visokom prilagodljivošću za računalnu obradu te omogućuje brzo renderiranje modela zgrada i urbanih krajolika, što je osobito korisno u scenarijima s manjom arhitektonskom raznolikošću (Kanda et al., 2024; Kubicka-Sowińska et al., 2024; Müller, Wonka, Haegler, Ulmer, & Van Gool, 2006). Njegova snaga leži u sposobnosti generiranja složenih urbanih pejzaža s minimalnim ručnim unosom, temeljem definiranih pravila i parametara (Parish & Müller, 2001). Usporedno, tradicionalni CAD i GIS alati, premda detaljni, zahtijevaju znatne resurse i vrijeme, te su manje skalabilni i osjetljivi na promjene u projektima (Kolbe, 2009). Iako takvi alati nalaze primjenu u arhitektonskom projektiranju, njihova ograničenja postaju vidljiva kod velikih urbanih sustava i zahtjevnih iterativnih procesa (Fatima & Sooda, 2023; X. Qin et al., 2023; Usta et al., 2023).

5.1. Motivacija, hipoteze i istraživački kontekst

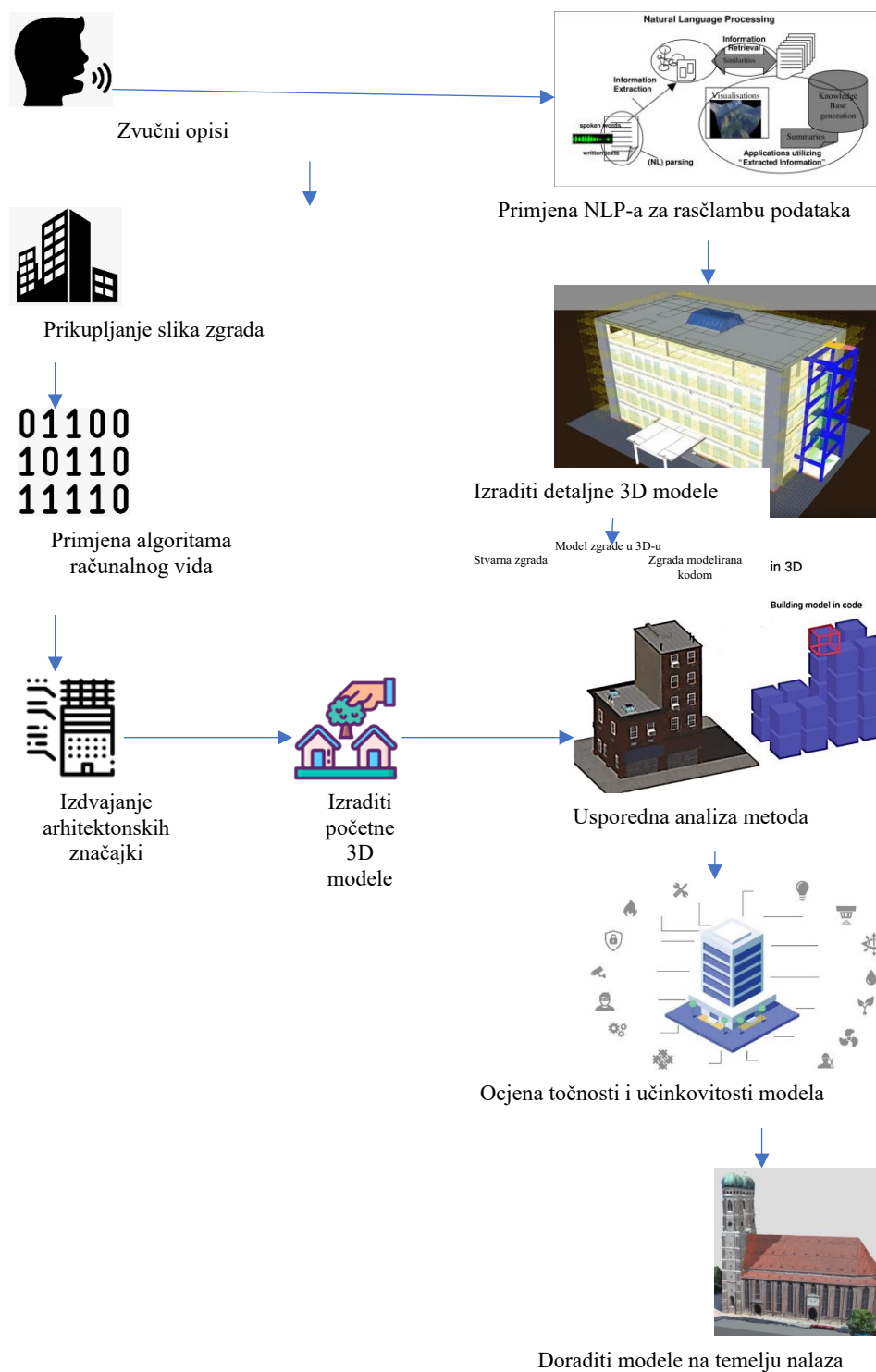
U ovom kontekstu postavljam dvije temeljne hipoteze. Prva pretpostavlja da se raznolikost zgrada iz stvarnog svijeta, unatoč njihovoj kompleksnosti, može uspješno obuhvatiti unutar procedurnog modeliranja. Druga hipoteza sugerira da se to može ostvariti kombinacijom suvremenih tehnika umjetne inteligencije (AI) i strojnog učenja, posebno kroz dva ključna pristupa: vizualno prepoznavanje karakteristika zgrada putem algoritama računalnog vida (CNN) i obrada tekstualnih opisa zgrada pomoću prirodnog jezika (NLP) (Adão et al., 2019; Kikuchi et al., 2023). Cilj je razviti metodologiju obrnutog procedurnog modeliranja, u kojem AI algoritmi generiraju skripte temeljene na vizualnim ili verbalnim podacima, automatizirajući tako cijeli proces izgradnje 3D modela (Ávila Parra, 2021; Sensmeier, 2017). Ovaj pristup predstavlja novu sintezu između pravila i podataka: umjesto tradicionalne izgradnje "pravila → model", ide se obrnutim putem "podatak → pravilo → model". Kako bih validirao ovu metodologiju, provodim komparativnu analizu modela generiranih AI alatima i onih izrađenih tradicionalnim procedurama. Evaluacija se temelji na kriterijima točnosti geometrije, vizualne vjernosti, skalabilnosti i računalne učinkovitosti (Oketunji, 2023; Yeguas et al., 2012). Rezultati ukazuju na značajan potencijal AI-integriranog procedurnog modeliranja za automatizaciju i poboljšanje 3D geovizualizacije zgrada. Korištenjem CNN-a za interpretaciju slika i GAN-a za generiranje tekstura postignuta je visoka razina realističnosti (Alzubaidi et al., 2021; Saldana &

Johanson, 2013). Ova metodologija nudi transformativne mogućnosti u kontekstu urbanističkog planiranja, arhitektonskog dizajna i razvoja pametnih gradova. Stoga, ovo istraživanje ne samo da pridonosi tehnološkom napretku u modeliranju, već i poticanju primjene automatiziranih i podatkovno vođenih metoda u praksi. Kroz sustavnu integraciju AI-a, procedurnih pravila i online izvora podataka, stvaram temelj za skalabilne, fleksibilne i inteligentne sustave za 3D modeliranje građevina.

5.1.1. Tehnološke mogućnosti i integracijski potencijal suvremenih metoda

Unatoč rastućem interesu za primjenu naprednih digitalnih tehnologija u području urbanog modeliranja, još uvijek postoji relativno mali broj znanstvenih istraživanja koja istovremeno integriraju procedurno modeliranje, strojno učenje i umjetnu inteligenciju (AI) za generiranje 3D modela zgrada. Tijekom ovog istraživanja, usmjerio sam se na analizu i primjenu relevantnih tehnika iz navedenih područja, pri čemu svaku razmatram s aspekta njezine metodološke učinkovitosti i potencijala za automatizaciju geometrijske i teksturne rekonstrukcije arhitektonskih objekata. U području procedurnog modeliranja, posebnu pozornost posvetio sam metodama koje omogućuju generiranje arhitektonskih struktura temeljenih na pravilima. Korištenjem parametarskih modela moguće je opisati elemente zgrada, kao što su visina, širina i stil, kroz numerički definirane varijable koje omogućuju varijacije i prilagodbe u stvarnom vremenu. Nadalje, gramatike oblika omogućuju strukturirano generiranje arhitektonskih stilova kroz hijerarhijske sustave pravila, dok procedurno teksturiranje koristi algoritamske obrasce poput fraktala ili šuma za definiranje vizualnih karakteristika površina (Leroux et al., 2024; Liu et al., 2023; Musgrave et al., 1994; Ruizhongtai Qi, 2020; Sun et al., 2023). Strojno učenje pokazuje se posebno korisnim u procesu prijelaza iz dvodimenzionalnih podataka u trodimenzionalnu reprezentaciju. Kroz eksperimentiranje s konvolucijskim neuronskim mrežama (CNN), trenirao sam modele koji mogu prepoznati i pretvoriti fotografije zgrada u odgovarajuće geometrijske oblike. Dodatno, generativne suparničke mreže (GAN) i varijacijski autoenkoderi (VAE) omogućili su mi stvaranje realističnih tekstura i nadopunu izgubljenih detalja u kompleksnijim fasadnim površinama kao i kodF(. Tehnike semantičke segmentacije omogućuju precizno izdvajanje građevinskih komponenti kao što su zidovi, prozori i vrata, što znatno doprinosi točnosti konačnog 3D modela. U kontekstu umjetne inteligencije, posebnu pažnju sam posvetio sustavima temeljenima na dubokim neuronskim mrežama koji omogućuju naprednu analizu i generiranje arhitektonskih struktura. Koristeći algoritme za prijenos stila, istraživao sam mogućnosti rekombinacije elemenata iz različitih povijesnih i suvremenih arhitektonskih stilova s ciljem

oblikovanja estetski koherentnih, ali inovativnih struktura. Također sam eksperimentirao s evolucijskim algoritmima za optimizaciju dizajna na temelju unaprijed definiranih kriterija, poput energetske učinkovitosti ili estetske privlačnosti, dok sam pomoću učenja pojačanja testirao mogućnosti automatskog poboljšanja modela kroz iterativne evaluacije i prilagodbe (Baduge et al., 2022). Naravno, iako sve ove tehnike nude znatne prednosti u pogledu učinkovitosti, skalabilnosti i razine automatizacije, suočavam se i s nizom izazova koji zahtijevaju dodatna istraživanja. Detaljnije se može vidjeti na URL 35. Automatizirani modeli nerijetko pokazuju manju preciznost u prikazu složenih arhitektonskih detalja. Njihova sposobnost generalizacije ograničena je obuhvatom skupa podataka na kojem su trenirani, što može dovesti do pogrešaka pri prijenosu modela na nove urbanističke kontekste (Liu et al., 2023). Nadalje, ovisnost o kvaliteti ulaznih podataka – bilo da je riječ o fotografijama, geolokacijskim informacijama ili tekstualnim opisima – izravno utječe na kvalitetu izlaznih modela. Obuka sofisticiranih modela zahtijeva velike količine računalnih resursa i visoko kvalitetne skupove podataka, što predstavlja tehnički i organizacijski izazov. Jedan od ključnih eksperimenata u ovom radu usmjeren je na evaluaciju dvaju pristupa unosu podataka u sustav za procedurno modeliranje. Usporedio sam učinkovitost verbalnih opisa – koji se obrađuju pomoću tehnika prirodnog jezika (NLP) – s rezultatima postignutima pomoću algoritama računalnog vida, konkretno uz korištenje DeepMindovog Alpha3D sustava. Dok prvi pristup omogućuje semantički bogatu interpretaciju arhitektonskih karakteristika, drugi koristi konvolucijske neuronske mreže za izdvajanje vizualnih elemenata i njihovu pretvorbu u trodimenzionalni prikaz (Slika 62). Kroz komparativnu analizu ovih metoda, ocjenjivao sam aspekte poput geometrijske točnosti, računalne učinkovitosti i subjektivne ocjene korisničkog iskustva. Rezultati pružaju vrijedne uvide u to koji se modalitet unosa pokazuje optimalnijim u različitim kontekstima primjene, čime se dodatno potvrđuje doprinos ove studije razvoju učinkovitih sustava za automatizirano 3D modeliranje građevinskih struktura.



Slika 62. Metodologija okvira istraživanja i komparativna analiza dvaju obrazaca istraživanja – prikupljanjem verbalnih opisa i prikupljanjem građevnih slika (URL 35).

(Slika 62) ilustrira dijagram toka eksperimentalnog procesa koji sam razvijao kako bih istražio učinkovitost različitih modaliteta unosa podataka u sustav procedurnog modeliranja temeljenog na umjetnoj inteligenciji. Cjelokupni tijek rada podijeljen je u dva paralelna smjera: prvi se temelji na verbalnim opisima, a drugi na vizualnim podacima – slikama i videozapisima zgrada.

U prvom smjeru, proces započinje prikupljanjem tekstualnih opisa arhitektonskih objekata. Ti opisi uključuju informacije o stilu, dimenzijama i građevinskim materijalima. Tehnike obrade prirodnog jezika (NLP) koristim za analizu i ekstrakciju semantičkih značajki iz tih tekstova, koje se zatim prevode u strukturirane parametre pogodne za generiranje početnih 3D modela. U drugom smjeru, radim s vizualnim podacima – arhivskim i terenskim fotografijama zgrada. Pomoću algoritama računalnog vida, konkretno CNN arhitektura kao što su Alpha3D i MIDAS, identificiram ključne arhitektonske elemente. Ti se elementi potom koriste za konstrukciju detaljnih 3D modela, koji precizno odražavaju stvarne karakteristike objekata. Nakon generiranja modela iz oba pristupa, provodim komparativnu analizu kako bih procijenio njihove prednosti i nedostatke. Ova analiza uključuje metrike kao što su točnost geometrije, vjerodostojnost prikaza, računalna učinkovitost te korisničko zadovoljstvo. Osim toga, iterativnim pristupom poboljšavam modele temeljem dobivenih nalaza, čime doprinosim povećanju njihove preciznosti i uporabljivosti. Za potrebe obuke AI modela koristim dva različita skupa podataka. Prvi sadrži tekstualne opise zgrada, koji služe za treniranje NLP modela – koristi se tokenizacija, sintaktička i semantička raščlamba. Drugi skup sadrži slike i videozapise, iz kojih se treniraju CNN modeli za izdvajanje vizualnih značajki. Oba modela treniram u iteracijama i optimiziram prema metrikama performansi. Kako bih validirao performanse, koristim više evaluacijskih pokazatelja: točnost modela se mjeri usporedbom s postojećim (ručno izrađenim ili stvarnim) modelima, računalna učinkovitost prema vremenu izrade i upotrebi resursa, a zadovoljstvo korisnika kroz ankete i stručne recenzije. Također evaluiram i skalabilnost – sposobnost sustava da modelira sve od pojedinačnih objekata do kompleksnih urbanih scenarija. Prva hipoteza – da se raznolike stvarne zgrade mogu prikladno modelirati procedurnim metodama – testiram razvojem fleksibilne gramatičke arhitekture koja uključuje različite stilove i morfološke varijacije. Modeli se zatim uspoređuju s izvornim strukturama prema točnosti, vizualnoj sličnosti i efikasnosti. Za referencu uključujem i modele izrađene tradicionalnim metodama kako bih osigurao analitički okvir za usporedbu. Druga hipoteza – da kombinacija AI i strojnog učenja može učinkovito informirati procedurno modeliranje – validira se implementacijom NLP i računalnog vida za generaciju modela. Rezultate AI modela uspoređujem s ručno izrađenima, vrednujući preciznost, razinu detalja i brzinu generiranja. Komparativnom analizom utvrđujem koja metoda bolje odgovara kontekstu raznolikih urbanih pejzaža. Kao dodatni korak, testiram hibridni pristup u kojem AI generira početne skripte za modeliranje, koje se zatim obrađuju procedurnim tehnikama. Time dobivam integrirani sustav koji kombinira prednosti oba pristupa. Evaluacija uključuje usporedbu rezultata s onima dobivenim zasebno, čime se istražuje sinergijski potencijal metode. Kako bih

osigurao znanstvenu validnost, koristim statističke metode poput t-testa i ANOVA testova za ispitivanje značajnosti razlika među eksperimentalnim skupinama. Time potvrđujem da su uočene razlike u performansama rezultat primijenjenih metoda, a ne slučajne varijacije. Ovakva evaluacija čini temelj pouzdanog zaključivanja i ponovljivosti rezultata u budućim istraživanjima.

5.1.2. Verbalni opisi kao input za 3D modeliranje

Korištenje verbalnih opisa uključuje davanje tekstualnih narativa ili formaliziranih podataka koji detaljno opisuju prostorne karakteristike, dimenzije te arhitektonske elemente određene zgrade. Ovaj način unosa omogućuje primjenu metoda obrade prirodnog jezika (NLP) radi analize i ekstrakcije ključnih informacija iz teksta. Prednost takvog pristupa leži u mogućnosti stvaranja dosljednog i računalno čitljivog formata kojim se jasno prenose tehničke specifikacije objekta, čime se olakšava njihova obrada putem AI algoritama. Ipak, učinkovitost ove metode uvelike ovisi o kvaliteti i preciznosti dostupnih opisa te o sposobnosti NLP sustava da prepozna i pravilno interpretira relevantne informacije. Na primjer, ista Frauenkirche u Münchenu s verbalnim opisom kao unosom nije potrebna jer je to dobro poznata crkva opisana mnogo puta na internetu i dostupna ChatGPT-u.

Ako se zahtjev „napišite mi python skriptu za procedurno modeliranje 3D modela za Frauenkirche u Münchenu unese u ChatGPT, generira se skripta dostupna u (MEGA, n.d.) i to je ono što nam daje AI.

5.1.3. Vizualna analiza i segmentacija slika zgrada za 3D rekonstrukciju

Učinkovitost procesa koji koristi tehnike digitalne obrade slike za identifikaciju i izdvajanje ključnih arhitektonskih elemenata, prostornih odnosa i strukturnih značajki iz vizualnih inputa i drugih naprednih modela dubokog učenja moguće je detektirati i najsitnije detalje na fasadama, što omogućuje izradu vrlo preciznih i informiranih procedurnih 3D modela. Implementacijom konvolucijskih neuronskih mreža (CNN) i drugih naprednih modela dubokog učenja moguće je detektirati i najsitnije detalje na fasadama, što omogućuje izradu vrlo preciznih i informiranih procedurnih 3D modela. ipak može biti ograničena vanjskim faktorima poput rezolucije snimke, nepovoljnih svjetlosnih uvjeta ili prisutnosti vizualnih zapreka (okluzija), koji otežavaju točno prepoznavanje i klasifikaciju značajki. U provedenim eksperimentima uspješno sam demonstrirao automatsko prepoznavanje zgrada i konstruiranje njihovih 3D prikaza. Iako su rezultati u većini slučajeva bili vizualno uvjerljivi, određeni

primjeri zahtijevaju dodatnu optimizaciju modela kako bi se poboljšala estetska i semantička točnost prikaza. Ovaj automatizirani pristup pokazao se izuzetno korisnim za potrebe kartografske i urbanističke geovizualizacije – kako na razini pojedinačnih objekata, tako i u kontekstu cjelovitih gradskih prostora. Prvi korak uključuje segmentaciju ulazne slike (Slika 63), kojom se izoliraju pojedini dijelovi zgrade (npr. prozori, vrata, krov), nakon čega umjetna inteligencija pronalazi odgovarajuće 3D komponente u unaprijed pripremljenim bazama podataka. Među korištenim izvorima nalaze se platforme poput OpenStreetMap (OSM), CityGML, SketchUp 3D Warehouse, Trimble Point Cloud Library (PCL), Google Poly, kao i BIM repozitoriji (npr. BIMobject, NBS National BIM Library) te standardizirani IFC modeli dostupni putem alata kao što su BIMserver i IfcOpenShell. Nakon prepoznavanja i selekcije relevantnih elemenata, primjenjuje se proces sastavljanja – oblikovanja i povezivanja dijelova u koherentnu cjelinu – čime nastaje konačni 3D model zgrade (Slika 64). Vrijeme generiranja pregleda modela varira: za brzi prikaz jednostavne strukture potrebno je otprilike 10 sekundi (Slika 64), dok složeniji model zahtijeva do 19 sekundi (Slika 65). Za potpuno renderirani model visoke razlučivosti, proces traje u prosjeku oko 45 minuta, ovisno o razini detalja i resursima korištene računalne infrastrukture.



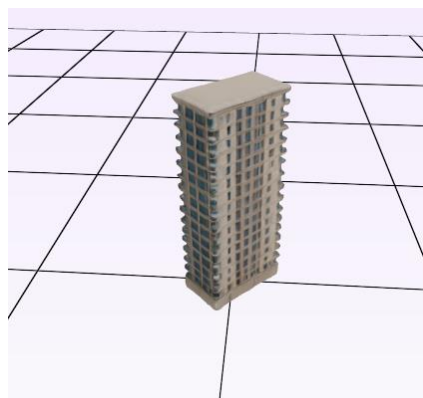
2 (a)



2 (b)



2 (c)



2 (d)

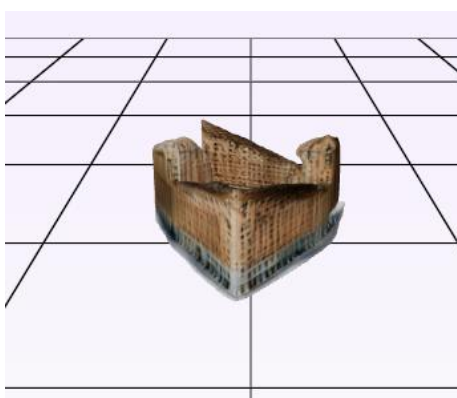
Slika 63. (a) Izvorna slika, (b) optičko prepoznavanje i segmentirana slika, (c, d) 3D model iste zgrade. Izvor: kombinacija Python-a i Blender-a (URL 35).



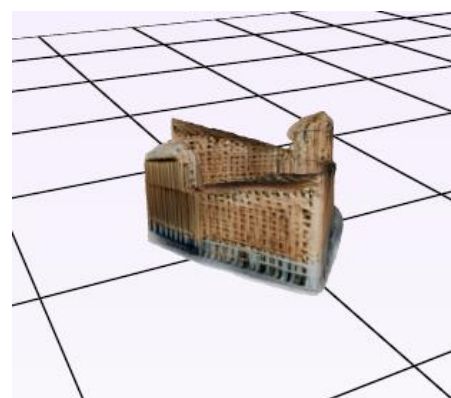
3 (a)



3 (b)



3 (c)



3 (d)

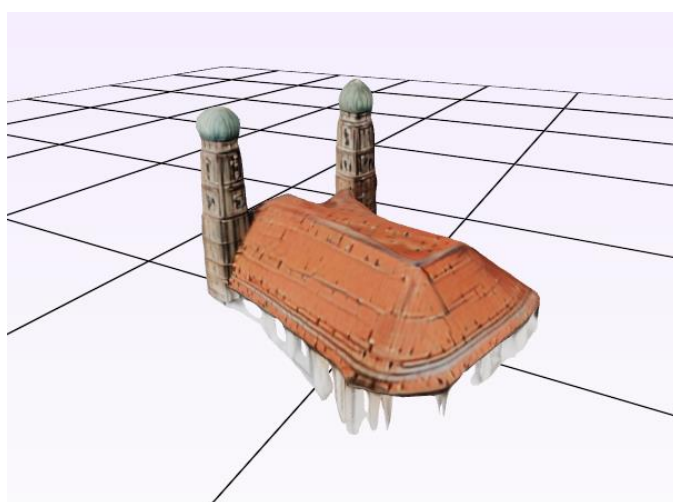
Slika 64. (a) Izvorna slika, (b) optičko prepoznavanje i segmentirana slika, (c, d) 3D model iste zgrade (URL 35).



(a)



(b)



(c)

Slika 65. (a) Izvorna slika, (b) optičko prepoznavanje i segmentirana slika, (c) 3D model iste zgrade (URL 35).

S obzirom na trenutno dostupnu razinu tehnološkog razvoja, postaje očito da kvaliteta i vjerodostojnost rezultirajućih 3D modela značajno ovise o vrsti i kutu vizualnog inputa. Na primjerima prikazanim na Slika 63 i Slika 64, može se primijetiti kako je model zgrade sa Slika 63 vizualno uvjerljiv i tehnički stabilan, dok zgrada prikazana na Slika 64 pokazuje izražene deformacije i smanjenu vjerodostojnost. Ključno ograničenje ovakvih metoda proizlazi iz činjenice da trenutačna tehnologija – uključujući i najnaprednije računalne algoritme – u najboljem slučaju može samo procijeniti kako izgleda neregistrirana (nefotografirana) strana objekta. Kada su dostupne slike samo s jedne strane zgrade, što je čest slučaj u urbanim okruženjima, algoritmi podrazumijevaju simetriju – odnosno, pretpostavljaju da je stražnja strana objekta jednaka ili vrlo slična prednjoj. Iako ova pretpostavka može biti zadovoljavajuća za jednostavne, simetrične građevine, ona postaje problematična kada se modelira kompleksna

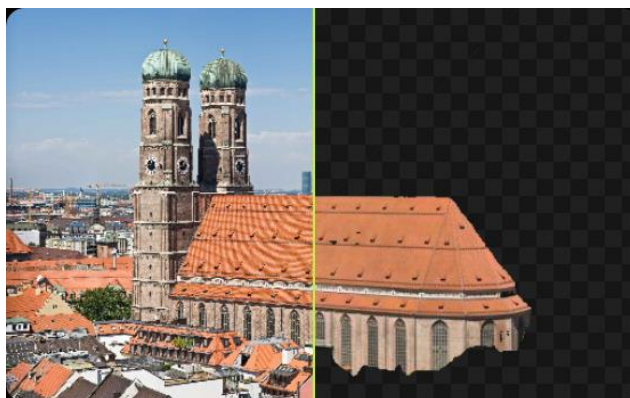
ili asimetrična arhitektura. Primjer na Slika 65 jasno pokazuje taj problem: crkveni tornjevi su neproporcionalno razmaknuti, a čitava forma građevine neprirodno je izdužena na suprotnu stranu od one s koje je fotografija snimljena. Takva deformacija narušava vizualni identitet objekta i dovodi u pitanje upotrebljivost modela u profesionalnim kontekstima. Zbog toga smatram nužnim uvesti dodatni korak u proces modeliranja: prethodno uklanjanje vizualno i semantički irelevantnih elemenata sa slike. Riječ je o filtriranju tzv. neželjenog viška – dijelova slike koji ne pripadaju ciljanoj zgradi i koji mogu ometati algoritme u preciznom izdvajanju arhitektonskih značajki. U slučajevima kada je zgrada okružena drugim objektima, ovaj proces može se automatizirati putem segmentacijskih modela koji detektiraju i uklanjaju okolne strukture, vegetaciju ili druge smetnje, čime se osigurava čišći ulazni signal za generiranje 3D modela. Ovaj dodatni korak – automatizirano čišćenje slike prije rekonstrukcije – ne samo da poboljšava točnost rezultata, već i smanjuje rizik od pogrešnog mapiranja značajki te pomaže u očuvanju integriteta arhitektonskog prikaza. Time se dodatno potvrđuje važnost kvalitetne pripreme ulaznih podataka i višekutne vizualne dokumentacije pri korištenju AI-sustava za modeliranje stvarnog svijeta. Međutim, s obzirom na dugotrajnost procesa, ovaj korak je zanemariv i može se izvesti online; na primjer, u Ref. (24AI, n.d.). Ograničenje ovakvog modela vidljivo je na Slika 65 i Slika 66 jer optičko prepoznavanje dijelova s izvorne slike nije moguće u donjem dijelu katedrale, gdje su okolne kuće i zgrade zaklanjaju podnožje katedrale, čineći je nevidljivom na izvornoj slici. Izvorna slika objekta uvijek treba biti cijeli objekt sa svih strana, što se u ovom slučaju može postići dronom ili terestričkom fotogrametrijom.



5 (a)



5 (b)



5 (c)

Slika 66. Slika prije (a) i nakon (b) koraka odvajanja glavnog objekta na slici i kombinacije (c) koja se namjerava na kraju oblikovati u 3D model pomoću umjetne inteligencije (URL 35).

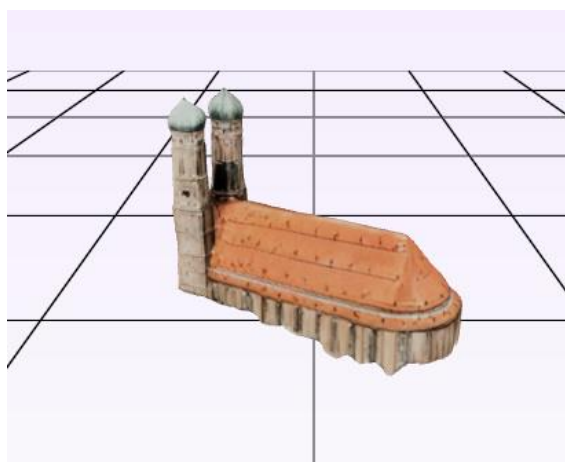
U prvoj segmentiranoj slici, donji dio katedrale nije adekvatno odvojen od okolnih objekata, poput susjednih zgrada i kuća, osobito onih smještenih u donjem dijelu kadra. Zbog toga segmentacija ne pruža dovoljno jasnu separaciju glavnog objekta, što značajno otežava pouzdano automatsko 3D modeliranje. Rezultat takvog procesa je deformiran model s neprirodno velikom širinom objekta te pretjeranim razmakom između dvaju tornjeva katedrale (Slika 65 c). Tek kada sam primijenio prethodno opisanu metodu automatskog modeliranja na pravilno pripremljenu sliku, bilo je moguće generirati adekvatan 3D model zgrade. Rezultat ove primjene prikazan je na Slika 67. Moja metoda, koja koristi unaprijed segmentiranu originalnu sliku, pokazuje se učinkovitijom – ne nužno u fazi same segmentacije (jer ona ostaje usporediva s rezultatima drugih metoda segmentacije glavnog objekta), već u kasnijoj fazi izgradnje modela. Naime, konačni 3D model u većoj mjeri odgovara stvarnom izgledu objekta u smanjenom mjerilu, što je jasno vidljivo na Slika 67 c i Slika 67 d.



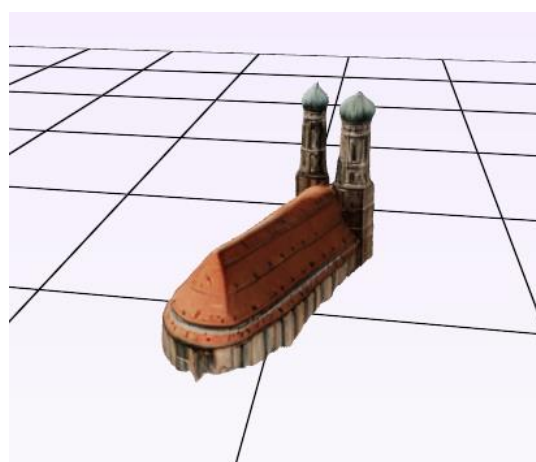
(a)



(b)



(c)



(d)

Slika 67. Mnogo bolji rezultat 3D modela crkve u Münchenu u odnosu na model na slici 3. **(a)** Izvorna slika, **(b)** optičko prepoznavanje i segmentirana slika, **(c, d)** 3D model iste zgrade (URL 35).

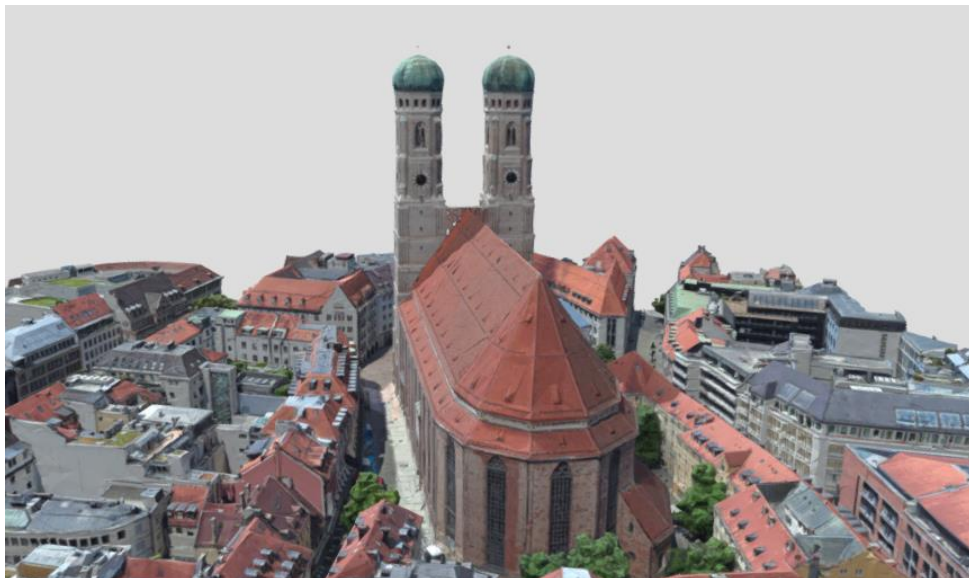
Ako se automatizacija i sve prethodno opisane metode primijene na katedralu i okolne objekte, može se dobiti kompletan model kao na Slika 68. U uskim ulicama najveći je izazov doći do fotografija koje služe kao izvor za optičko prepoznavanje i kasniji proces modeliranja, a koje najbolje opisuju svaku pojedinu zgradu.



(a)



(b)



(c)



(d)

Slika 68. Model katedrale (Frauenkirche u Münchenu) i okolnih zgrada iz različitih smjerova (a – d) (URL 35).

5.1.4. Interpretacija i izazovi u primjeni

Iako sam ovom studijom uspio istaknuti potencijale integracije umjetne inteligencije i procedurnog modeliranja u kontekstu 3D vizualizacije zgrada, važno je prepoznati niz ograničenja koja su se pojavila tijekom istraživačkog procesa. Jedno od ključnih ograničenja proizlazi iz same prirode korištenih podataka – budući da su treninzi i evaluacije modela temeljeni na specifičnim skupovima slika i ograničenom broju arhitektonskih stilova, mogućnost generalizacije rezultata ostaje upitna. Modeli koji su razvijeni i trenirani u ovom okviru možda neće ostvariti jednaku razinu uspješnosti kada se primijene na građevine iz drugačijih kulturnih, geografskih ili estetskih konteksta. Osim toga, pouzdanost procesa u velikoj mjeri ovisi o kvaliteti ulaznih podataka. Kod primjene metoda strojnog učenja, razlike u rezoluciji slike, svjetlosnim uvjetima i kutu snimanja značajno utječu na preciznost izlaznog 3D modela. Takve varijacije ne samo da umanjuju točnost, već otežavaju konzistentnu reprodukciju rezultata. Još jedan izazov koji se pokazao jest znatan računalni kapacitet potreban za izvođenje algoritama temeljenih na dubokom učenju. U sredinama gdje su računalni resursi ograničeni, to predstavlja ozbiljno ograničenje za svakodnevnu upotrebu ovih metoda. Također sam uočio kako fokus na algoritamsku stranu problema može dovesti do zanemarivanja ljudskog faktora. Usvajanje novih tehnologija, osobito u tradicionalnim disciplinama poput arhitekture i urbanog planiranja, sa sobom nosi određenu krivulju učenja. Taj aspekt implementacije – spremnost i sposobnost krajnjih korisnika da učinkovito primjenjuju ove alate – mora biti uzet u obzir u daljnjim istraživanjima. U tom smislu, budući radovi trebali bi proširiti analizu i na utjecaj alata vođenih umjetnom inteligencijom na svakodnevni tijek rada stručnjaka. U pokušaju da dodatno istražim koji su oblici ulaza najučinkovitiji u procesu

modeliranja, proveo sam komparativnu analizu između dva temeljna modaliteta: verbalnih opisa građevina i vizualnog (optičkog) prepoznavanja karakteristika zgrada. Procjena je obuhvatila niz metrika, uključujući točnost konačnih modela, vjernost strukturama stvarnog svijeta, računalnu učinkovitost, ali i zadovoljstvo korisnika. Usporedno su analizirani i kvalitativni aspekti – poput jednostavnosti prikupljanja ulaznih podataka, interpretabilnosti rezultata te potencijala za skaliranje sustava. Rezultati analize ukazali su na to da izbor modaliteta unosa može značajno utjecati na ishod modeliranja. Ovisno o prirodi zadanog problema, jedan se modalitet može pokazati učinkovitijim od drugoga. Na primjer, u slučajevima gdje su dostupni detaljni tekstualni opisi zgrade, verbalni unos može pružiti dovoljno informacija za konstrukciju adekvatnog modela. S druge strane, kada su na raspolaganju bogati vizualni podaci, kao što su fotografije ili video snimke zgrada, tehnike optičkog prepoznavanja omogućuju veću preciznost i prostornu vjernost modela. Zanimljivo je primijetiti da kombinirani – hibridni – pristupi, koji istodobno koriste verbalne i vizualne informacije, mogu ponuditi sinergijske prednosti, ublažiti slabosti svakog pojedinog modaliteta te unaprijediti ukupnu robusnost modela. Na temelju ove analize, mogu zaključiti da je pitanje izbora modaliteta unosa u procedurnom modeliranju zasnovanom na umjetnoj inteligenciji od ključne važnosti. Ima neposredne posljedice ne samo na točnost i izvedbu modela, već i na način dizajna, primjene i evaluacije sustava u raznim domenama. U tom kontekstu, detaljno sam istražio pojedine aspekte svakog pristupa. Kada se govori o verbalnim opisima, važno je istaknuti da oni mogu biti zapisani u raznim oblicima – od jednostavnih tekstualnih prikaza, preko arhitektonskih nacрта, do tehničkih dijagrama. Alati za obradu prirodnog jezika (NLP) omogućuju raščlambu ovih opisa i izdvajanje ključnih podataka koji se potom koriste za generiranje 3D modela građevine. Kod vizualnog unosa, algoritmi računalnog vida, a posebno konvolucijske neuronske mreže (CNN), pokazali su se ključnima za prepoznavanje elemenata kao što su fasade, prozori, vrata i nosive strukture. Segmentacijom slike i detekcijom značajki moguće je točno locirati i klasificirati komponente građevine (3DCSM, n.d.), čime se stvara osnova za vjerodostojnu 3D rekonstrukciju. Nadalje, sustavi temeljeni na umjetnoj inteligenciji omogućuju analizu kompleksnijih odnosa – uključujući prostorne konfiguracije, arhitektonske stilove i projektantske intencije. Duboke neuronske mreže, poput RNN i transformatora, koriste se za interpretaciju tekstualnih opisa, dok Generative Adversarial Networks (GANs) omogućuju sintetiziranje realističnih 3D modela učenjem iz bogatih arhitektonskih baza. Pristup mrežnim bazama podataka također se pokazao korisnim. Takve baze sadržavaju katalogizirane elemente građevina – poput tekstura, 3D komponenti i materijala – koji se mogu ugraditi u modeliranje. Tehnike poput API pristupa ili web struganja omogućuju automatsko

dohvaćanje tih podataka i njihovu integraciju u cjelokupan cjevovod modeliranja. U svrhu optimizacije dizajna, implementirao sam algoritme strojnog učenja koji kroz iterativno učenje – potaknuto povratnim informacijama korisnika – poboljšavaju kvalitetu izlaznog modela. Tehnike poput učenja pojačanjem i evolucijskih strategija koriste se kako bi se otkrila najefikasnija dizajnerska rješenja. Također, parametarsko modeliranje se pokazalo korisnim u stvaranju fleksibilnih i dinamičnih modela. Korištenjem alata poput Grasshoppera ili Autodeskovog Dynamo, moguće je precizno definirati dimenzije i odnose među građevnim elementima, što omogućuje brze izmjene i testiranje različitih scenarija dizajna. Uvođenjem sustava temeljenih na pravilima postigao sam veću usklađenost s arhitektonskim normama, zakonskim propisima i ekološkim standardima. Pravovremena validacija modela značajno smanjuje rizik od grešaka i potrebe za naknadnim korekcijama. Tijekom procesa, veliki značaj pridavao sam i aspektu interaktivnosti. Korištenjem sučelja virtualne stvarnosti (VR) i grafičkih korisničkih sučelja (GUI), omogućio sam korisnicima izravno manipuliranje modelima uz trenutni vizualni feedback. Ovakva interakcija dodatno poboljšava korisničko iskustvo i omogućuje intuitivno oblikovanje prostora. Na kraju, razmatrao sam i pitanja skalabilnosti i prilagodljivosti. Implementacijom distribuiranih računalnih sustava i otvorenih softverskih arhitektura postignuta je visoka razina fleksibilnosti, što omogućava korištenje ovog pristupa u širokom rasponu aplikacija – od urbanističkog planiranja do izgradnje individualnih objekata. Zaključno, mogu potvrditi da sinergijska primjena umjetne inteligencije, strojnog učenja, napredne vizualizacije i podatkovno vođenih pristupa otvara nove dimenzije u arhitektonskom modeliranju. Time ne samo da se ubrzava i pojednostavljuje proces dizajna, već se stvaraju temelji za održivija, efikasnija i vizualno bogatija izgrađena okruženja.

5.1.5. Ključni elementi za izgradnju kvalitetnog tekstualnog inputa u 3D modeliranju

Kako bih osigurao da 3D model zgrade bude precizan, vjeran stvarnosti i dosljedan geovizualizacijskim zahtjevima, ključno je prikupiti sveobuhvatan opis koji obuhvaća niz međusobno povezanih arhitektonskih i kontekstualnih značajki. Proces modeliranja uvelike ovisi o kvaliteti i dubini ulaznih podataka, stoga svaki element mora biti pažljivo specificiran kako bi digitalni prikaz vjerno reflektirao stvarni objekt. Prije svega, polazim od identifikacije arhitektonskog stila same zgrade. Razumijevanje stilske pripadnosti – bilo da se radi o suvremenoj, modernističkoj, tradicionalnoj arhitekturi ili određenim povijesnim pravcima poput gotike, Bauhauusa, postmoderne ili Art Decoa – pomaže mi postaviti vizualni i konceptualni okvir unutar kojeg model mora biti razvijen. Ova informacija izravno utječe na oblikovne principe, izbor materijala i proporcije. Vanjski izgled zgrade čini sljedeći važan sloj

opisa. Uključujem detalje o fasadi, koristeći podatke o vrstama materijala poput betona, stakla, čelika ili cigle, kao i o rasporedu i geometriji dekorativnih elemenata poput balkona, sjenila, nadstrešnica i arhitektonskih ornamenata. Posebnu pažnju posvećujem shemama boja i površinskim tretmanima, budući da upravo oni određuju vizualni identitet objekta u prostoru. U nastavku, razrađujem ukupni volumen i formu zgrade. Opisujem raspored i organizaciju prostora, uključujući broj etaža, proporcije, oblik u tlocrtu (pravokutan, kutan, cilindričan i slično), kao i ključne statičke elemente poput stupova, greda i konzola. Pritom specificiram i glavne dimenzije – dužinu, širinu i visinu – kako bi se osigurala skalabilnost i preciznost 3D modela. Poseban segment posvećujem prozorima i vratima, pri čemu ne navodim samo njihov položaj i dimenzije, već i konstrukcijske osobitosti poput tipologije (klizna, dvokrilna, prozorska, jednostruka ili dvostruka), materijala izrade (staklo, drvo, metal) i stilskih detalja koji doprinose estetskom izrazu objekta. Krovna konstrukcija zahtijeva zasebnu analizu. Opisujem njen oblik – primjerice, je li riječ o ravnom, kosom, zabatnom ili četverostrešnom krovu – te preciziram korištene materijale poput crijepa, metala ili šindre. Također, u obzir uzimam eventualne dodatke poput krovnih prozora, terasa, zelenih površina i drugih elemenata koji utječu na funkcionalnost i vizualnu dinamiku gornje etaže. Okruženje zgrade neraskidivo je povezano s njenom arhitekturom. Stoga uključujem informacije o uređenju okoliša, što obuhvaća opis vrtova, pješačkih staza, zelenih površina, stabala, fontana ili zona za odmor, jer svi ti elementi doprinose cjelokupnom dojmu i uklapanju zgrade u širi kontekst. U slučajevima kada je cilj izraditi i unutarnji model, potrebno je unijeti i podatke o unutarnjoj organizaciji prostora. To uključuje raspored prostorija, oblik i dimenzije unutarnjih pregrada, visinu stropova, vrste podova, rasvjetu i eventualne trajne elemente namještaja poput stubišta ili ugradbenih ormara. Međutim, u ovom istraživanju fokusiram se primarno na vanjsku strukturu modela. Uz tehničke i oblikovne karakteristike, važno mi je prikupiti i šire kontekstualne podatke. Lokacija zgrade, njena orijentacija, klimatski uvjeti, ograničenja parcele i zahtjevi održivosti mogu utjecati na način na koji se objekt projektira, ali i na metode koje koristim za njegovu digitalnu rekonstrukciju. U specifičnim klimatskim ili urbanim uvjetima određene značajke – poput sjenila, ventilacijskih otvora ili zelenih krovova – dobivaju na funkcionalnom značaju. Detaljna razrada fasade često zahtijeva poseban fokus. Osim odabranih materijala, dokumentiram uzorke i tekture, odnos između punih i praznih površina te integraciju elemenata poput lukova, stupova, terasa ili konzola. Takva analiza omogućuje mi preciznu rekonstrukciju fasadnog rastera i osigurava estetsku vjernost digitalnog modela. Na kraju, završni sloj dokumentacije odnosi se na sve fine detalje koji daju završni karakter građevini. To uključuje boje, površinske premaze, reljefne uzorke, ali i dekorativne dodatke na fasadi,

krovu ili oko ulaza. Premda često zanemareni u tehničkoj dokumentaciji, upravo ti elementi oblikuju identitet zgrade i čine razliku između generičkog i uvjerljivog modela. Kako bih dodatno osigurao vjernost interpretacije i izvedbe, uz tekstualni opis nastojim uključiti i dodatne vizualne reference. To mogu biti arhitektonski crteži, skice, renderi ili referentne fotografije koje omogućuju preciznije prepoznavanje proporcija, materijala i odnosa između elemenata. Kombinacijom ovih informacija mogu uspješno rekonstruirati složene arhitektonske strukture na način koji zadovoljava funkcionalne, estetske i tehničke zahtjeve projekta.

5.1.6. Komparativne analize: točnost, skalabilnost i primjenjivost

Optički pristup, iako računalno zahtjevniji i ovisan o kvaliteti ulaznih slika, nadmašuje verbalni u svojoj sposobnosti hvatanja zamršenih arhitektonskih detalja. Oba pristupa imaju svoje prednosti i izazove, što je jasno vidljivo u izvedbi rezultirajućih modela. U provedbi ove studije, proveo sam komparativnu analizu koja otkriva kako pristup generiranja 3D modela temeljen na umjetnoj inteligenciji, koji koristi verbalne opise kao ulazne podatke, predstavlja strukturiran i učinkovit način modeliranja zgrada. Ipak, takav pristup pokazuje određena ograničenja u pogledu detaljne preciznosti, osobito kada se usporedi s metodom temeljenom na optičkom prepoznavanju vizualnih značajki. Tijekom istraživanja, uz vlastite doprinose, oslanjao sam se i na uvid u recentnu znanstvenu literaturu. Poseban doprinos mog istraživanja ogleda se u razvoju metodologije za procedurno modeliranje vođeno umjetnom inteligencijom, koje povećava preciznost i učinkovitost na razini pojedinačne građevine. S druge strane, studija (Kamra et al., 2023) bavi se izazovima modeliranja u urbanim sredinama velikih razmjera, pri čemu je naglasak stavljen na smanjenje računalnog opterećenja. Kombiniranjem njihovih uvida s mojima, dolazim do zaključka da se procedurno modeliranje može učinkovito povezati s metodama lagane rekonstrukcije, čime se otvaraju novi putevi za optimizaciju 3D vizualizacija unutar pametnih gradova. Nadalje, moje istraživanje nadovezuje se i na studiju (Khayyal et al., 2022), u kojoj je procedurno modeliranje prepoznato kao ključna metoda za brzo generiranje kompleksnih urbanih struktura. Iako njihov pristup omogućuje stvaranje opsežnih modela pogodnih za urbanističke simulacije, uočene su slabosti poput ograničene raznolikosti modela, ograničenja u ručnoj prilagodbi i nedovoljno precizne teksture. Ovdje sam intervenirao uvođenjem umjetne inteligencije u obliku konvolucijskih neuronskih mreža (CNN) i generativnih kontradiktornih mreža (GAN), koje omogućuju automatsko prevođenje slika u geometriju i generiranje tekstura. Time sam uspio prevladati identifikacijske slabosti tradicionalnih procedurnih pristupa, smanjujući potrebu za ručnim radom, uz očuvanje visoke razine realizma, čime se dodatno potvrđuju ciljevi. Ova sinergija procedurnih i AI metoda

predstavlja temeljni doprinos u razvoju budućih alata za urbaniste i arhitekte. Kombinacija omogućuje znatno poboljšanu vizualizaciju, bolju prilagodljivost i veću učinkovitost u kontekstu 3D modeliranja pametnih gradova. Dodatno sam ispitao kako se ovaj pristup nadograđuje na spoznaje iz rada (Somanath et al., 2024), gdje je fokus bio na tijekovima rada digitalnih blizanaca u velikim urbanim prostorima. Za razliku od njihova pristupa koji se temelji na tradicionalnim procedurama, moja metodologija vođena umjetnom inteligencijom optimizira kreiranje složenih građevinskih struktura i tekstura te omogućuje stvaranje dinamičnih i realističnih modela pogodnih za simulaciju urbanih sustava. Zanimljivo je i povezivanje s istraživanjem (Usta et al., 2024), koji razvijaju web-aplikaciju za upravljanje 3DCM (3D Composite Manufacturing). Dok se njihov alat fokusira na obradu i prostornu analizu, moj pristup unaprjeđuje preciznost modela generiranih iz 2D podataka. Umjetna inteligencija u ovom kontekstu ne samo da povećava točnost izrade zgrada već i znatno ubrzava cjelokupan proces modeliranja, čime se značajno doprinosi učinkovitosti web-baziranih geovizualizacijskih platformi. Osim toga, pokazao sam kako se doprinos mog rada logički nadovezuje na studiju (M. Zhang et al., 2022), u kojoj se procedurno modeliranje temelji na CGA gramatici i tradicionalnim pravilima oblikovanja. Integracijom AI tehnika, kao što su CNN-ovi i GAN-ovi, postizem veću razinu automatizacije, fleksibilnosti i kreativnosti u urbanističkom dizajnu, čime dodatno proširujem njihovu metodološku osnovu. S obzirom na sve navedeno, ključni znanstveni doprinos mog istraživanja može se sažeti u nekoliko točaka. Prvo, razvio sam novi integrirani okvir za automatizirano procedurno modeliranje koji objedinjuje prednosti oba pristupa – tekstualnog i vizualnog unosa. Drugo, pružio sam empirijsku validaciju učinkovitosti tog okvira, potvrdivši njegove prednosti u pogledu točnosti, računalne učinkovitosti i vjernosti modela. Treće, uočio sam šire implikacije ovog pristupa na domene urbanog planiranja, arhitekture i digitalne vizualizacije prostora. U svrhu provjere postavljenih hipoteza i osiguranja znanstvene konzistentnosti, primijenio sam niz preciznih tehničkih metoda i strukturirao eksperimentalni dizajn koji obuhvaća višestruke ulazne scenarije i metrike evaluacije. U prvoj hipotezi polazio sam od pretpostavke da procedurno modeliranje može precizno prikazati raznolikost arhitektonskih formi. Validacija je provedena kroz usporedbu tradicionalnih metoda geovizualizacije s mojom AI-proširenom procedurom. Algoritmi poput CNN-a korišteni su za vizualno prepoznavanje karakteristika, dok su metode obrade prirodnog jezika (NLP) iskorištene za semantičku analizu tekstualnih opisa zgrada. Razvijeni modeli ocijenili su stručnjaci na temelju mjerila kao što su strukturalna točnost, vizualna vjernost i računalna učinkovitost. Druga hipoteza odnosila se na potencijal umjetne inteligencije da automatizira proces izrade modela zgrada korištenjem različitih modaliteta

unosu, uključujući slike i tekst. Da bih je ispitao, implementirao sam algoritme koji integriraju višestruke izvore podataka u proces generiranja. Posebnu ulogu imao je pristup obrnutog procedurnog modeliranja, gdje AI koristi opisne i vizualne informacije za automatsko generiranje skripti potrebnih za izradu 3D modela. Usporedba izlaznih modela s ručno izrađenim i stvarnim građevinama omogućila mi je preciznu procjenu točnosti, razine detalja i računalnog opterećenja. Nadalje, kroz korisničke studije u kojima su sudjelovali stručnjaci iz područja kartografije i urbanizma, procijenjena je i praktična upotrebljivost razvijenog sustava. Kako bih dodatno konkretizirao rezultate, proveo sam niz studija slučaja. U jednoj sam simulirao modeliranje urbanog stambenog područja, demonstrirajući kako AI-procedurno modeliranje može vjerno prikazati tipološku raznolikost kuća na temelju tekstualnih i vizualnih podataka. U drugoj sam se fokusirao na povijesno gradsko središte, koristeći arhivske fotografije i opise za rekonstrukciju složenih stilskih obilježja, čime sam ilustrirao primjenjivost metode u očuvanju baštine. Treća studija odnosila se na suvremenu poslovnu zonu, gdje sam istraživao mogućnosti modeliranja velikih, modularnih objekata kao što su poslovne zgrade i trgovački centri, pokazujući kako AI može efikasno upravljati repetitivnim elementima i velikim prostornim obuhvatima. Na temelju navedenih nalaza, mogu potvrditi kako integracija umjetne inteligencije u procedurno modeliranje ne samo da poboljšava tehničke aspekte vizualizacije, već i omogućuje skalabilnost, prilagodljivost i primjenjivost u različitim domenama – od urbanističkog planiranja do digitalne kartografije i arhitektonske analize.

5.2. Izazovi, ograničenja i budući smjerovi

Učinkovitost modela umjetne inteligencije izravno ovisi o raznovrsnosti i preciznosti ulaznih podataka. Kada se koriste verbalni opisi, nedostaci poput nepreciznosti ili nedovršenosti često rezultiraju pogrešnim generiranjem 3D modela. S druge strane, modeli temeljeni na optičkom prepoznavanju oslanjaju se na kvalitetu vizualnog materijala; niska rezolucija, prepreke u kadru ili nepovoljni svjetlosni uvjeti mogu znatno smanjiti pouzdanost rezultata. Iako prepoznajem značajan napredak koji donosi pristup procedurnom modeliranju temeljenom na umjetnoj inteligenciji, posebno u kontekstu poboljšanja geovizualizacije i automatizacije 3D modeliranja zgrada, svjestan sam brojnih izazova i ograničenja koje je potrebno adresirati kako bi se ova metodologija u potpunosti mogla primijeniti u praksi. Među ključnim izazovima ističem kvalitetu i dostupnost podataka. Također sam uočio kako je generalizacija različitih arhitektonskih stilova i urbanih okruženja još jedno ograničenje. Kada se modeli treniraju na ograničenim skupovima podataka, njihova primjenjivost na nove, složenije ili nepoznate

tipologije zgrada postaje upitna. To posebno dolazi do izražaja u kontekstima s bogatom arhitektonskom raznolikošću. Uz to, smatram da je visoka računalna zahtjevnost modela dubinskog učenja – osobito onih usmjerenih na obradu slike – prepreka za širu primjenu, osobito u okruženjima s ograničenim tehničkim resursima. Još jedan izazov koji bih istaknuo odnosi se na nedostatak transparentnosti i interpretabilnosti ovih sustava. Iako su rezultati često impresivni, procesi odlučivanja ostaju zatvoreni i teško razumljivi, što može predstavljati problem u disciplinama poput urbanističkog planiranja ili zaštite kulturne baštine, gdje je razumijevanje procesa ključan preduvjet za donošenje informiranih odluka. Praktične poteškoće prepoznajem i u integraciji ovih tehnologija unutar postojećih profesionalnih radnih tokova. Pitanja kompatibilnosti s etabliranim softverskim rješenjima, kao i potreba za specijaliziranim znanjima u radu s AI sustavima, predstavljaju realne prepreke za njihovo šire prihvaćanje. U budućem razvoju metodologije, usmjerio bih se na stvaranje bogatijih i raznovrsnijih skupova podataka koji bi obuhvaćali širi spektar stilova i okolišnih uvjeta, čime bi se poboljšala sposobnost modela za generalizaciju. Također bih istražio mogućnosti hibridnih modela koji kombiniraju verbalne i vizualne inpute, jer bi takva sinergija mogla nadomjestiti slabosti svakog pojedinačnog pristupa. Uz to, primjena naprednih AI tehnika – poput generativnih kontradiktornih mreža (GAN) za obogaćivanje podataka ili učenja pojačanjem za optimizaciju modela – mogla bi dodatno unaprijediti kvalitetu i fleksibilnost modela. Vjerujem i da bi se primjenom tih tehnika mogao postići viši stupanj kontekstualne svjesnosti u generiranim 3D prikazima. Povećanje interpretabilnosti modela također smatram prioritetom. Razvoj objašnjivih modela umjetne inteligencije (XAI) mogao bi značajno pridonijeti razumijevanju i prihvaćanju rezultata, osobito u profesionalnim kontekstima. Nadalje, istražio bih načine za razvoj energetski učinkovitijih AI sustava, kako bi njihova implementacija bila moguća i na slabijem hardveru ili kroz cloud rješenja. Time bi se znatno povećala dostupnost ovih alata. Konačno, usmjerio bih se na unaprjeđenje integracije AI procedurnog modeliranja u postojeće CAD i BIM platforme kroz razvoj dodataka ili API-ja. Takva tehnička interoperabilnost mogla bi znatno olakšati usvajanje ove tehnologije unutar sektora arhitekture, građevine i inženjeringa (AEC). Iako smatram da ovaj pristup donosi snažan iskorak, njegovo potpuno ostvarenje ovisit će o sposobnosti da se sustavno rješavaju identificirani izazovi i da se kontinuirano razvijaju nova, poboljšana rješenja. U budućem istraživačkom radu planiram istražiti povezivanje procedurnog modeliranja vođenog umjetnom inteligencijom s naprednim tehnikama generativne AI, konkretno s alatima za generiranje slike iz teksta kao što je Midjourney. Smatram da se ovi alati mogu koristiti za podršku kako modernoj tako i baštinskoj arhitekturi. U kontekstu suvremenog dizajna, koristio bih tekstualne

opise kao osnovu za brzu produkciju vizualnih koncepata, koji bi potom bili razrađeni pomoću AI procedura. Time bi se omogućio kreativniji i eksperimentalniji pristup projektiranju. U domenu baštinske arhitekture i urbanizma, vjerujem da bi generativna umjetna inteligencija mogla značajno doprinijeti restauraciji i očuvanju povijesnih lokaliteta. Kombiniranjem tekstualnih opisa nestalih ili oštećenih objekata s procedurnim modeliranjem, mogli bismo konstruirati točne i vizualno uvjerljive 3D rekonstrukcije, koje bi služile i fizičkoj obnovi i digitalnoj arhivaciji kulturne baštine. Ovakva spekulativna povezivanja u skladu su s pristupima koje opisuje (Barrio, 2023), koji se bavi geovizualizacijom kao sredstvom za kreativno reinterpretiranje prostora. Također uzimam u obzir analizu (Sukkar et al., 2024), koja naglašava nedostatke trenutnih AI prikaza islamske arhitekture – što mi pokazuje gdje bi procedurno modeliranje moglo nadopuniti te tehnologije. Osim toga, nalazi (Jaalama et al., 2021) ukazuju na važnost spajanja stručnog znanja s inovativnim vizualnim alatima u urbanom planiranju, čime se jača kvaliteta komunikacije i dizajna. U konačnici, buduće bih istraživanje usmjerio na integraciju ovih tehnologija kako bih pridonio razvoju metodologija koje spajaju teorijski napredak s praktičnom primjenom u kartografiji i geovizualizaciji 3D modela zgrada.

5.3. Uloga umjetne inteligencije u budućnosti geovizualizacije i urbanog modeliranja

Na temelju uvida koje sam stekao, mogu zaključiti da izbor modaliteta unosa mora biti usklađen s kontekstom i ciljevima modeliranja. Primjerice, u situacijama gdje su dostupni precizni tekstualni opisi (npr. arhitektonski nacrti), verbalni unos može biti dovoljan. U vizualno bogatim okruženjima, poput onih s obiljem slika visoke rezolucije, optičko prepoznavanje pokazuje nadmoć. Konačni izbor ovisi o prioritetima – želimo li veću točnost, bržu izvedbu ili veću prilagodbu korisniku. U radu sam ustanovio da verbalni opisi omogućuju jednostavno i fleksibilno definiranje detaljnih specifikacija, osobito u slučajevima kada su tekstualne informacije lako dostupne. Uz primjenu naprednih tehnika obrade prirodnog jezika (NLP), moguće je izvući ključne informacije koje potom služe kao temelj za generiranje modela. Ipak, prepoznao sam da učinkovitost ovog pristupa uvelike ovisi o jasnoći i potpunosti opisa. Kod složenih arhitektonskih detalja, koji su teško prenosivi verbalno, ovaj modalitet pokazuje slabosti jer ne uspijeva precizno rekonstruirati vizualne nijanse. U provedenoj komparativnoj analizi procedurnog modeliranja potpomognutog umjetnom inteligencijom, usmjerio sam se na usporedbu učinkovitosti verbalnih opisa i optičkog prepoznavanja građevinskih slika kao dva temeljna modaliteta unosa. Ova usporedba mi je omogućila dublje razumijevanje specifičnih

prednosti i ograničenja svakog pristupa. Uzimajući u obzir metrike poput točnosti modela, razine vjernosti strukturama iz stvarnog svijeta, računalne učinkovitosti i zadovoljstva korisnika, postalo mi je jasno koliko je važan strateški odabir ulaznog modaliteta u procesu dizajniranja i primjene sustava za 3D modeliranje. Nasuprot tome, korištenjem optičkog prepoznavanja slika, uz potporu konvolucijskih neuronskih mreža (CNN), uspio sam dohvatiti kompleksne arhitektonske karakteristike iz vizualnih podataka. Ova metoda se pokazala vrlo učinkovitom za precizno modeliranje elemenata poput fasada, prozora i vrata, zahvaljujući detaljnoj analizi slike i mogućnostima segmentacije. Ipak, za ostvarenje visoke razine točnosti potreban je pristup velikim količinama označenih podataka visoke kvalitete, kao i snažnoj računalnoj infrastrukturi, što može predstavljati značajnu prepreku u pojedinim kontekstima. Tijekom analize došao sam do zaključka da hibridni pristupi, koji integriraju tekstualne i vizualne ulaze, predstavljaju najperspektivniji smjer razvoja. Na taj način mogu se kombinirati prednosti oba pristupa – dok tekst pruža strukturalne i dimenzionalne informacije, slike osiguravaju detalje konteksta i vizualne karakteristike. Ova kombinacija omogućuje kreiranje modela visoke preciznosti, ali i fleksibilnosti. Prepoznajem da je za praktičnu primjenu ovih tehnologija ključno pitanje skalabilnosti i prilagodljivosti. Korištenjem računalnih okruženja temeljenih na oblaku i distribuiranih sustava, moguće je učinkovito upravljati modeliranjem u velikim razmjerima. Modularne softverske arhitekture otvorenog koda omogućuju mi visoku razinu prilagodbe i nadogradnje, čime se sustavi mogu oblikovati prema specifičnostima pojedinih projektnih zadataka. Gledajući unaprijed, vjerujem da buduća istraživanja trebaju istražiti integraciju naprednih modela poput generativnih kontradiktornih mreža (GAN) i algoritama za učenje s pojačanjem. Njihova upotreba mogla bi dodatno unaprijediti kvalitetu i učinkovitost procedurnog modeliranja. Također smatram važnim razvijati sveobuhvatne baze podataka i mehanizme za prikupljanje povratnih informacija u stvarnom vremenu, kako bi sustavi bili responsivniji i kontekstualno svjesniji. U konačnici, ova analiza mi je omogućila duboko razumijevanje odnosa između različitih pristupa procedurnom modeliranju temeljenom na umjetnoj inteligenciji. Integracijom njihovih prednosti, mogu razviti naprednije i pouzdanije modele, s izravnim implikacijama na arhitektonsko projektiranje, urbanističko planiranje i šire područje geovizualizacije.

6. Zaključci

Cilj ovog doktorskog rada bio je razviti, evaluirati i prikazati sveobuhvatan pristup automatizaciji procedurnog modeliranja 3D zgrada, koristeći kombinaciju formalnih gramatika, algoritama strojnog učenja, LiDAR podataka, OSM-a, te različitih softverskih alata otvorenog i komercijalnog koda. Poseban naglasak stavljen je na integraciju vizualnih i semantičkih podataka, kao i na primjenu u urbanim kontekstima s ciljem unaprjeđenja geovizualizacije.

Polazišna točka rada jest promišljanje kako reducirati ručno 3D modeliranje zgrada i omogućiti skalabilnu primjenu modela u kontekstu urbanističkog planiranja, geodezije, prostorne analize i simulacija. U tom smislu, procedurno modeliranje nudi mogućnost stvaranja velikog broja realističnih, podatkovno bogatih i vizualno uvjerljivih 3D zgrada, što je i demonstrirano kroz primjere i studije slučaja u radu. U radu je detaljno prikazan sustav za procedurno modeliranje krovova i fasada koristeći CityEngine, Blender i vlastite Python skripte, čime su omogućene varijacije i automatizacije generiranja raznovrsnih stilova i oblika. Pristup je evaluiran i dokazana je visoka učinkovitost u smanjenju ručnog rada, pogotovo kod velikih skupova zgrada (npr. čitavi kvartovi ili gradske jezgre), međutim, ukazano je i na nekoliko ograničenja:

Niska razina detalja (npr. LOD2) ne omogućuje semantičko obogaćivanje modela (npr. evakuacijski planovi). Modeli nemaju realistične fasade – potrebno je integrirati metode segmentacije i tipizacije pročelja, odnosno ulične vizure i vizualna hijerarhija zgrada nisu dovoljno obuhvaćeni u baznim procedurama bez dodatnih ulaznih podataka. Četvrto poglavlje rada bavi se komparacijom dva pristupa:

Verbalni opisi → obrada prirodnog jezika (NLP);

Optičko prepoznavanje građevinskih slika → konvolucijske neuronske mreže (CNN).

Pokazalo se da verbalni opisi omogućuju brzu i jednostavnu interpretaciju kada su tekstualni podaci dostupni, no često pate od nedostatka preciznosti i kontekstualne vjernosti. Optičko prepoznavanje, s druge strane, nudi visoku točnost, ali traži velike računalne resurse i velike baze podataka. Najveći potencijal pokazali su hibridni modeli, koji kombiniraju tekstualne i vizualne ulaze, omogućujući sintezu točnih i semantički bogatih 3D modela. Skalabilnost i prilagodljivost procedurnog modeliranja ovise o korištenju modularne softverske arhitekture, standardizaciji (npr. CityGML ili CityJSON formati), integraciji sa softverom otvorenog koda i oblacima za obradu velikih podataka i razini semantičkih metapodataka dostupnih za zgrade. Preporučuje se korištenje GIS alata za izvlačenje maksimalne količine informacija iz dostupnih prostornih podataka i njihova automatska transformacija u semantičke attribute zgrada. U evaluaciji sustava kod krajnjih korisnika (stručnjaci, urbanisti, studenti), istaknuto je visoko

zadovoljstvo sustavom, mogućnost brzih prilagodbi arhitektonskih stilova, smanjenje potrebe za ručnim intervencijama, potreba za dodatnim edukativnim i vizualnim alatima, potencijal za široku primjenu u urbanističkom planiranju, geovizualizaciji, kartografiji, virtualnoj stvarnosti i edukaciji. Identificirano je više razina ograničenja, ograničena generalizacija AI-modela na nepoznate stilove i regije, visoki zahtjevi za hardverom i obradom, potreba za standardizacijom semantičkih oznaka i atributa, nedovoljna uključenost ljudskih čimbenika – potrebno je više pažnje posvetiti sučeljima prilagođenim ne-programerima. Kao prioritetne smjernice istraživanja i razvoja izdvojen je razvoj interaktivnog alata za vizualizaciju pravila, kako bi korisnici mogli pratiti kako se pravila primjenjuju na modele, zatim uvođenje apstraktnih mehanizama selekcije elemenata i automatizirano generiranje dizajna na temelju primjera, dodatna integracija umjetne inteligencije u interpretaciju složenih urbanih scena, istraživanje primjene modela u kontekstu digitalnog očuvanja kulturne baštine, proširenje metodologije na regije s nedostatkom prostornih podataka i razvoj robusnih modela otpornih na nedostupne ulazne podatke. Istraživanje ima višestruke znanstvene doprinose, razvoj novog okvira za automatizirano procedurno modeliranje, koji kombinira više metoda, povezivanje metoda računalnog vida, gramatika i NLP-a u jedinstven model, uvođenje graf-transformacijskih mehanizama za automatsku izmjenu stilova i geometrije, pokazana je primjena u različitim softverima, što potvrđuje interoperabilnost i praktičnost pristupa, te empirijska validacija učinkovitosti metoda kroz komparativne analize i evaluacije stručnjaka. Istraživanja pokazuju da automatizacija procedurnog modeliranja 3D zgrada nije samo tehnički moguća već i visoko iskoristiva u profesionalnom kontekstu. S obzirom na ubrzan razvoj umjetne inteligencije, računalnog vida i semantičkih GIS sustava, daljnji razvoj ovog pristupa mogao bi značajno promijeniti paradigmu vizualizacije urbanih prostora. Rad pruža čvrstu osnovu za buduća istraživanja, unaprjeđenje postojećih alata i razvoj nove generacije modela urbanih 3D okruženja u kojima se integriraju semantički, funkcionalni i vizualni aspekti zgrada.

Pojmovnik

(Prilog s definicijama i objašnjenjima nekih pojmova iz disertacije)

1. **Procedurno modeliranje** – Tehnika automatskog generiranja 3D modela pomoću unaprijed definiranih pravila i algoritama, bez potrebe za ručnim modeliranjem svakog elementa.
2. **Gramatika (u kontekstu modeliranja)** – Skup pravila koja određuju kako se osnovni elementi (oblici) mogu kombinirati i mijenjati kako bi nastali složeniji 3D objekti.
3. **CGA pravila (Computer Generated Architecture)** – Poseban jezik korišten u alatima poput CityEngine-a, koji opisuje kako se zgrade generiraju iz osnovnih geometrijskih oblika pomoću pravila.
4. **Neterminalni simboli** – Elementi unutar gramatike koji se još mogu razložiti u druge simbole; koriste se za označavanje struktura koje se dodatno obrađuju (npr. "zgrada", "kat").
5. **Terminalni simboli** – Završni elementi unutar pravila koji više ne zahtijevaju dodatnu obradu – predstavljaju konkretne dijelove modela (npr. prozor, vrata).
6. **Start shape (početni oblik)** – Početna geometrija na kojoj se primjenjuju pravila za generiranje složenijih modela, npr. osnovna površina zgrade.
7. **Split (podjela)** – Naredba koja dijeli neki element (npr. fasadu) u manje dijelove (npr. prozore i zidove).
8. **Repeat (ponavljanje)** – Omogućuje da se neki element (npr. prozor) automatski ponavlja duž određene osi, dok god ima prostora.
9. **Parametrizacija** – Dodjeljivanje svojstava modelu (npr. visina, širina, broj katova) koja se mogu mijenjati za dobivanje različitih varijanti.
10. **Inverzno procedurno modeliranje** – Tehnika u kojoj se pravila generiraju na temelju postojećih 3D modela, npr. zgrada skeniranih u stvarnosti.
11. **Manipulatori** – Interaktivni alati unutar softverskog sučelja (npr. ručke za povlačenje) koji omogućuju korisniku da mijenja geometriju modela u stvarnom vremenu.
12. **Fasada** – Vanjski izgled zgrade koji uključuje prozore, vrata, zidove i druge arhitektonske detalje.
13. **Oblak točaka (point cloud)** – Skup digitalno zabilježenih točaka u prostoru, najčešće prikupljenih pomoću 3D skenera, koji predstavljaju površinu objekta.
14. **LiDAR (Light Detection and Ranging)** – Tehnologija koja koristi laserske zrake za precizno mapiranje prostora i generiranje 3D podataka.
15. **Semantička segmentacija** – Proces identifikacije i označavanja različitih dijelova 3D modela prema njihovoj funkciji (npr. fasada, krov, prozor).
16. **Geovizualizacija** – Vizualni prikaz geografskih i prostornih podataka u 2D ili 3D obliku, radi lakšeg razumijevanja prostora.
17. **Hijerarhija pravila** – Organizacija pravila u sustav gdje jedno pravilo poziva drugo, čime se omogućuje modularno i slojevito modeliranje.
18. **Stohastička pravila** – Pravila koja uključuju nasumičnost, omogućujući varijabilnost i raznolikost generiranih modela.
19. **Voxel mreža** – Trodimenzionalna mreža malih kockica (voxela) koja se koristi za prikaz i obradu volumenskih podataka.
20. **Funkcija preferencije** – Algoritamski model koji pokušava predvidjeti korisnikove želje (npr. estetika) i prema tome usmjeriti generiranje modela.
21. **Bayesova indukcija gramatike** – Statistička metoda kojom se na temelju postojećih primjera uči nova gramatika za generiranje sličnih modela.

22. **Monte Carlo Markov Chain (MCMC)** – Algoritam koji koristi slučajnost za pronalaženje najboljeg rješenja među velikim brojem mogućnosti.
23. **Level of Detail (LOD)** – Razina detalja 3D modela; koristi se za prilagodbu prikaza ovisno o udaljenosti ili svrsi (npr. pregled cijelog grada vs. detalj fasade).
24. **Urbanističko modeliranje** – Proces izrade digitalnih prikaza gradova za potrebe analize, planiranja ili simulacije.
25. **CityEngine** – Softver koji koristi CGA pravila za generiranje 3D gradova i urbanih scena, posebno koristan u urbanizmu, GIS-u i arhitekturi.
26. **Forma (oblik)** – Geometrijski prikaz pojedinog objekta u prostoru, često osnovna jedinica u procedurnom modeliranju.
27. **Distribucija vjerojatnosti** – Statistički prikaz vjerojatnosti pojavljivanja određenih svojstava modela unutar skupa mogućih rješenja.
28. **Kontekstno slobodna gramatika** – Tip gramatike kod koje pravila vrijede neovisno o mjestu u kojem se primjenjuju – pogodna za formalni opis modela.
29. **Podijeljena gramatika** – Gramatika u kojoj su pravila organizirana po dijelovima modela (npr. krovna pravila, pravila za fasade), omogućujući veću kontrolu.
30. **Preferencijska funkcija korisnika** – Sklonosti korisnika izražene kroz ocjene ili interakcije, koje sustav koristi za prilagodbu rezultata modeliranja.
31. **Procedurni jezik** – Jezik razvijen za opis i izvođenje generativnih pravila u modeliranju.
32. **Vizualno sučelje** – Grafičko korisničko sučelje koje omogućuje manipulaciju modelima bez potrebe za unosom koda.
33. **Topološka mreža** – Struktura koja opisuje način na koji su dijelovi modela povezani (npr. koje plohe dijele bridove).
34. **Modeliranje u stvarnom vremenu** – Proces oblikovanja i uređivanja modela pri čemu su promjene odmah vidljive u 3D prikazu.
35. **Interaktivna vizualizacija** – Vrsta vizualizacije gdje korisnik može izravno utjecati na prikaz podataka (npr. rotirati, povećavati, filtrirati).
36. **Segmentacija geometrije** – Proces razdvajanja kompleksne geometrije na manje dijelove s ciljem lakšeg analiziranja ili obrade.
37. **Parametarski manipulator** – Alat koji omogućuje mijenjanje vrijednosti parametara direktno u prikazu modela.
38. **Tekstualna gramatika** – Gramatika zapisana kao niz tekstualnih pravila, obično u obliku programskog jezika.
39. **Vizualna gramatika** – Reprezentacija gramatičkih pravila pomoću grafičkih elemenata (čvorova i veza).
40. **Tokenizacija** – Proces pretvaranja niza znakova u manje jedinice (tokene), obično korišten u obradi teksta i jezika.
41. **Parametarsko kodiranje** – Način opisivanja objekata korištenjem parametara koji određuju njihova svojstva (npr. visinu, širinu, boju).
42. **Generativni dizajn** – Dizajnerski pristup u kojem računalno automatski generira oblike na temelju zadanih pravila i ograničenja.
43. **Top-down pristup** – Pristup gdje se modeliranje započinje od općeg oblika i zatim se razrađuje u detalje.
44. **Bottom-up pristup** – Pristup gdje se model gradi od manjih dijelova prema cjelini.
45. **Graf struktura** – Skup čvorova povezanih vezama, često korišten u vizualnim jezicima i gramatikama.
46. **Razina apstrakcije** – Stupanj udaljenosti od konkretnog prikaza; više apstraktna pravila opisuju opće postupke, ne konkretne detalje.

47. **Vizualno programiranje** – Programiranje pomoću grafičkih elemenata, umjesto pisanja koda.
48. **Čvor (node)** – Element u vizualnoj gramatici koji predstavlja akciju, funkciju ili oblik.
49. **Veza (edge)** – Linija koja povezuje čvorove i prikazuje odnos ili tijek izvršavanja.
50. **Renderiranje** – Proces pretvaranja 3D modela u 2D sliku s efektima poput svjetla, sjene i tekstura.
51. **Mesh (mreža)** – Geometrijska struktura sastavljena od točaka, bridova i ploha koja definira oblik objekta.
52. **NURBS (Non-Uniform Rational B-Splines)** – Matematički model za prikazivanje glatkih zakrivljenih površina u 3D.
53. **Implicitno modeliranje** – Metoda opisivanja oblika pomoću funkcija, bez eksplicitnog definiranja površine.
54. **Eksplicitno modeliranje** – Modeliranje pomoću jasno definiranih površina, rubova i točaka.
55. **Geometrijski shader** – Dio grafičkog sustava koji obrađuje geometriju objekata prije prikaza.
56. **Normalna vektora** – Vektor okomit na površinu, koristi se za izračun svjetla i sjene.
57. **Transformacija (translate, rotate, scale)** – Operacije pomicanja, rotiranja i skaliranja objekata u prostoru.
58. **3D koordinatni sustav** – Sustav s osi X, Y i Z koji određuje položaj objekata u prostoru.
59. **Orijentacija objekta** – Smjer u kojem je objekt „okrenut” u 3D prostoru.
60. **Interoperabilnost** – Mogućnost međusobne suradnje različitih sustava i formata podataka.
61. **OpenGL** – Standardni programski sučaj za prikazivanje 2D i 3D grafike.
62. **GPU (Graphics Processing Unit)** – Procesor specijaliziran za izvođenje grafičkih operacija.
63. **Shader** – Mali program koji definira kako se površine prikazuju u grafičkom sustavu.
64. **Render farma** – Skupina računala koja paralelno izvršavaju renderiranje za brže rezultate.
65. **Interaktivno modeliranje** – Modeliranje pri kojem korisnik vidi i kontrolira rezultat u stvarnom vremenu.
66. **Funkcionalna modularnost** – Podjela sustava na neovisne dijelove koji imaju jasno definirane funkcije.
67. **Algoritamski dizajn** – Dizajniranje uz pomoć algoritama koji generiraju oblike na temelju pravila.
68. **Deklarativno programiranje** – Stil programiranja u kojem se opisuje što želimo postići, a ne kako.
69. **Imperativno programiranje** – Stil programiranja u kojem se točno opisuje redoslijed koraka za postizanje rezultata.
70. **Metapodatci** – Podatci koji opisuju druge podatke, npr. autor, vrijeme stvaranja modela.
71. **Binarna reprezentacija** – Kodiranje informacija u obliku 0 i 1 za računalo.
72. **Verzija modela** – Različite inačice modela koje prikazuju evoluciju tijekom vremena.
73. **Arhitektonska tipologija** – Klasifikacija građevina prema funkciji, obliku i strukturi.
74. **Regeneracija modela** – Ponovno generiranje modela na temelju novih pravila ili podataka.
75. **Simulacija urbanog rasta** – Modeliranje razvoja gradova kroz vrijeme korištenjem algoritama.

76. **Analiza pogleda (view analysis)** – Proračun vidljivosti iz određenih točaka u prostoru.
77. **Osjenčavanje (shading)** – Tehnika u renderiranju koja dodaje dubinu i volumen površinama.
78. **Tekstura** – Slika koja se primjenjuje na površinu modela radi realističnog izgleda.
79. **UV mapiranje** – Proces kojim se tekstura „razvlači” preko površine 3D modela.
80. **Top-down modeliranje** – Pristup u kojem se modeli najprije generiraju kao cjelina, a zatim se razlažu u dijelove.
81. **Parametarski dizajn** – Oblikovanje objekata pomoću varijabli koje definiraju njihove karakteristike.
82. **CAD (Computer-Aided Design)** – Softverski sustav za tehničko i inženjersko crtanje.
83. **BIM (Building Information Modeling)** – Metoda projektiranja zgrada koja kombinira geometriju i informacije o građevini.
84. **GIS (Geografski informacijski sustav)** – Sustav za pohranu, analizu i prikaz prostornih podataka.
85. **Kombinatorna gramatika** – Gramatika koja kombinira više pravila kako bi proizvela nove oblike.
86. **Topološka konzistentnost** – Pravilo da geometrijski elementi moraju biti pravilno povezani bez rupa ili preklapanja.
87. **Kontrolne točke** – Točke koje određuju oblik zakrivljene linije ili površine.
88. **Implicitna reprezentacija** – Opisivanje oblika pomoću jednadžbi umjesto eksplicitne geometrije.
89. **Real-time rendering** – Tehnika prikaza gdje se slika računa dovoljno brzo za trenutni prikaz.
90. **Procedurna tekstura** – Tekstura koja se generira algoritamski, a ne iz slike.
91. **Virtualna rekonstrukcija** – Digitalna obnova izgleda povijesnih ili nestalih objekata.
92. **Skeniranje iz zraka (aerofotogrametrija)** – Prikupljanje prostornih podataka iz zraka pomoću kamera ili senzora.
93. **Referentni sustav koordinata** – Sustav za određivanje položaja objekata u prostoru.
94. **Normalizacija podataka** – Usklađivanje vrijednosti podataka u usporedivi raspon.
95. **Digitalni model terena (DTM)** – 3D prikaz površine zemlje bez objekata poput zgrada.
96. **Digitalni model površine (DSM)** – 3D prikaz koji uključuje sve objekte na površini, uključujući zgrade i vegetaciju.
97. **Ray tracing** – Tehnika renderiranja koja simulira ponašanje svjetlosti za realistične efekte.
98. **Volumetrijska analiza** – Analiza količine prostora koje objekt zauzima.
99. **Teksturalna klasifikacija** – Proces prepoznavanja površinskih svojstava objekata (npr. glatko, grubo).
100. **Automatizirana generacija pravila** – Proces u kojem sustav sam generira pravila na temelju primjera ili podataka.

Popis slika i tablica

Slika 1. Jedan od pristupa kada se uzima unešeni 3D model, izvlači komponente, pronalazi uzorke, konstruira grane stabla, gradi gramatiku bez konteksta i omogućava interaktivnu sintezu operacija (URL 1). 5

Slika 2. Izdvajanje pravila i/ili parametara za stvaranje modela iz zadanih podataka zgrada-fotografija, podataka iz oblaka točaka i sl.) (URL 2).	8
Slika 3. Razdvajanje na komponente (lijevo) i njihovo razdvajanje na uzorke i pravila (desno).	8
Slika 4. Povratno se može podijeliti ravninu fasade na dijelove (URL 3).	8
Slika 5. Naći gramatiku koja daje željeni 3D model.	9
Slika 6. Preferencije korisnika kroz ocjenjivanje nekoliko uzoraka	29
Slika 7. Pročelje ili prednja fasada muzeja Mimara modelirana težinskom atributnom gramatikom.	41
Slika 8. Pojedinačni dijelovi središnjeg dijela zgrade Mimare.	42
Slika 9. Dijagram toka WACFG modela strojnog učenja za fasade zgrada. Rezultati faze obuke su SVM-model (za klasifikaciju fasadne strukture-označeno crvenom) i MLN-model (procjena parametara fasadnih dijelova-označeno žutom). Ti se koraci (označeno plavom) primjenjuju u fazi predviđanja kako bi se na kraju izveo model fasade.	47
Slika 10. Shema korištene relacijske baze podataka (RBDB).	48
Slika 11. Poslijeratne fasade zgrada u Njemačkoj	49
Slika 12. Ilustrativni model analize primjera fasade. Izvedeno je izlazno stablo analize y, dana sekvenca dijelova građevine x pomoću vektora frekvencije pravila $W_{\phi x}$; y_{ϕ} .	51
Slika 13. Stablo gramatike. Svaki "Kat" se dalje razlaže na "Prozore" i "Vrata", a ti se simboli na kraju pretvaraju u konkretne objekte.	62
Slika 14. Početna jednostavna zgrada za geovizualizaciju i početak transformacije u druge stilove.	64
Slika 15. Shema koja prikazuje cijeli proces prepisivanja grafa korak po korak.	68
Slika 16. Dijagram koji objašnjava svaki korak transformacije grafa – od početne fasade, preko primjene pravila, do složenije strukture.	70
Slika 17. Transformacija na realističnoj fasadi. Primjećuje se kako se različita pravila mogu kombinirati bilo kojim redoslijedom i kako se mogu koristiti s različitim stilovima (gornji red: ciglena fasada, donji red: betonska fasada).	71
Slika 18. Pojednostavljeni različiti stilovi fasada i gradnje do danas.	75
Slika 19. Dva primjera jednostavne zgrade koja je napravljena s osnovnim elementima prozora, vrata, i jednostavnim krovom zbog lakše analize i nakon toga transformacije u različite stilove.	76
Slika 20. Gotički stil. Vidljive su varijacije koje se događaju i sve razlike za lijevu i desnu zgradu sa (Slika 19) i njihove varijacije u transformacijama za gotički stil.	76
Slika 21. Također gotički stil za početne zgrade iz (Slika 19).	77
Slika 22. Slike modela (Slika 20) u perspektivnom prikazu. Ovdje su slike građevine u gotičkom stilu iz perspektive zraka koje prikazuju njezine šiljaste lukove, strmi krov, i zamršene arhitektonske detalje.	77
Slika 23. Barokni i rokoko stil.	78
Slika 24. Klasicizam i neoklasicizam.	78
Slika 25. Historicizam i neogotika.	79
Slika 26. Moderna i suvremena arhitektura.	79
Slika 27. Dijelovi za npr. gotički stil zgrade (URL 8).	80
Slika 28. Dijelovi za vrata za zgrade u rimskom stilu (URL 8).	80
Slika 29. Prosječne ocjene ispitanika po pitanjima na Likertovoj skali (1-7).	81
Slika 30. Postotak visoko pozitivnih odgovora (ocjene 5, 6, ili 7) za svako postavljeno pitanje.	82
Slika 31. Prosječne ocjene za dodatna pitanja vezana uz funkcionalnost i korisnost sustava.	82
Slika 32. Prosječne ocjene za dodatna pitanja vezana uz funkcionalnost i korisnost sustava.	83
Slika 33. Postotak pozitivnih odgovora za dodatna pitanja.	83

Slika 34. Postotak pozitivnih odgovora na dodatna pitanja.	84
Slika 35. 3D modeli gradova imaju mnoštvo različite primjene za vizualizaciju okoliša i podršku odlučivanju u različitim domenama ljudskog djelovanja (modificirano iz (Biljecki et al., 2015)).	88
Slika 36. Standardni tijek rada u CityEngine-u.	92
Slika 37. Različita geometrija krovova različitih oblika zgrada.	96
Slika 38. Lasersko snimanje i dobivanje vjernog prikaza krova zgrade u odnosu na dobivanje istovjetnog složenog oblika krova iste zgrade pojednostavljenim procedurnim modeliranjem.	96
Slika 39. Nakon podjele komponente, svaka strana krova sadrži lomne ravnine prilikom umetanja.	96
Slika 40. Klasifikacija trim ravnine ovisi o kutu između "generirajućeg" ruba i xz ravnine.	97
Slika 41. Okomite ravnine za rezanje omogućene su prema zadanim postavkama. Ravnine obrezivanja prikazane su ružičastom bojom.	97
Slika 42. Primjer generiranja dijela grada procedurnim modeliranjem.	97
Slika 43. Granica analiziranog područja u središtu Ljubljane, Slovenija, prikazana je crvenom linijom (© OpenStreetMap contributors).	101
Slika 44. Izlazni podatci (elevacija) obrade oblaka točaka u CloudCompare: (a) DSM; (b) DTM.	103
Slika 45. Sloj Shapefile s tlocrtima zgrada nakon filtriranja OSM podataka (© <i>OpenStreetMap contributors</i>).	103
Slika 46. Klasificirani DSM u LAStools-u.	104
Slika 47. Visinske razlike između DSM i DTM.	105
Slika 48. 3D model zgrade uvezen u Blender.	107
Slika 49. Vrste krovova na analiziranom području.	108
Slika 50. Rezultat pravila primijenjenih na ravne krovove.	109
Slika 51. Postavke unutar sučelja geometrijskih čvorova za dvostrešne krovove.	109
Slika 52. Rezultat matematičkih pravila primijenjenih na dvostrešne krovove.	110
Slika 53. Postavke unutar sučelja geometrijskih čvorova za šatorske i kose krovove.	110
Slika 54. Vizualizacija šatorskih i kosih krovova u blenderu nakon (a) primjene matematičkih pravila; (b) ručnog uređivanja.	111
Slika 55. Vizualizacija sakralne građevine u Blender-u.	111
Slika 56. Konačni 3D model u LOD2.	112
Slika 57. Usporedba dvaju prikaza za sakralni objekt u (a) Blender-u; (b) Google Earth-u.	112
Slika 58. Prikaz poslovne četvrti Ljubljane, prikazan u (a) Blender-u; (b) Google Earth-u.	113
Slika 59. Građevine s atributima u OSM-u koji opisuju vanjski izgled (narančasto).	114
Slika 60. Zgrade u OSM-u koje imaju atribut tipa krova (žuto).	114
Slika 61. Pokrivenost zgrada s automatski generiranom geometrijom krova (zeleno) i geometrijom krova s potrebnim ručnim poboljšanjem (crveno).	115
Slika 62. Metodologija okvira istraživanja i komparativna analiza dvaju obrazaca istraživanja – prikupljanjem verbalnih opisa i prikupljanjem građevnih slika.	124
Slika 63. (a) Izvorna slika, (b) optičko prepoznavanje i segmentirana slika, (c, d) 3D model iste zgrade. Izvor: kombinacija Python-a i Blender-a.	128
Slika 64. (a) Izvorna slika, (b) optičko prepoznavanje i segmentirana slika, (c, d) 3D model iste zgrade.	128
Slika 65. (a) Izvorna slika, (b) optičko prepoznavanje i segmentirana slika, (c) 3D model iste zgrade.	129
Slika 66. Slika prije (a) i nakon (b) koraka odvajanja glavnog objekta na slici i kombinacije (c) koja se namjerava na kraju oblikovati u 3D model pomoću umjetne inteligencije (24AI 2024).	131

Slika 67. Mnogo bolji rezultat 3D modela crkve u Münchenu u odnosu na model na slici 3. (a) Izvorna slika, (b) optičko prepoznavanje i segmentirana slika, (c, d) 3D model iste zgrade.	132
Slika 68. Model katedrale (Frauenkirche u Münchenu) i okolnih zgrada iz različitih smjerova (a – d).	134
Tablica 1. Popis korištenih predikata	52
Tablica 2. Za predviđene slične predikate na temelju MWS i BP, F1 rezultati.	55
Tablica 3. Izbor softvera.	106
Tablica 4. Visinske razlike između podataka dobivenih Lidarom i OSM podataka u metrima (URL 14)	118

Literatura

- 24AI. (n.d.). *Tools*. Retrieved October 19, 2024, from <https://24ai.tech/en/tools/cut-object/>
- Adão, T., Pádua, L., Hruška, J., Peres, E., Sousa, J. J., Morais, R., & Magalhães, L. G. (2017). Bringing together UAS-based land surveying and procedural modelling of buildings to set up enhanced VR environments for cultural heritage. *2017 24o Encontro Português de Computação Gráfica e Interação (EPCGI)*, 1–8.
- Adão, T., Pádua, L., Marques, P., Sousa, J. J., Peres, E., & Magalhães, L. (2019). Procedural Modeling of Buildings Composed of Arbitrarily-Shaped Floor-Plans: Background, Progress, Contributions and Challenges of a Methodology Oriented to Cultural Heritage. *Computers*, 8(2), 38. <https://doi.org/10.3390/computers8020038>
- Aliaga, D. G., Rosen, P. A., & Bekins, D. R. (2007). Style grammars for interactive visualization of architecture. *IEEE Transactions on Visualization and Computer Graphics*, 13(4), 786–797.
- Alomía, G., Loaiza, D., Zúñiga, C., Luo, X., & Asorey-Cacheda, R. (2021). Procedural modeling applied to the 3D city model of bogota: a case study. *Virtual Reality & Intelligent Hardware*, 3(5), 423–433. <https://doi.org/https://doi.org/10.1016/j.vrih.2021.06.002>
- Alzubaidi, L., Zhang, J., Humaidi, A. J., Al-Dujaili, A., Duan, Y., Al-Shamma, O., Santamaría, J., Fadhel, M. A., Al-Amidie, M., & Farhan, L. (2021). Review of deep learning: concepts, CNN architectures, challenges, applications, future directions. *Journal of Big Data*, 8(1), 53. <https://doi.org/10.1186/s40537-021-00444-8>
- Armeni, I., Sener, O., Zamir, A. R., Jiang, H., Brilakis, I., Fischer, M., & Savarese, S. (2016). 3d semantic parsing of large-scale indoor spaces. *Computer Vision and Pattern Recognition*.
- Attene, M., Falcidieno, B., & Spagnuolo, M. (2006). Hierarchical mesh segmentation based on fitting primitives. *Vis. Comp*, 22(3), 181–193.
- Ávila Parra, R. (2021). *House generation using procedural modeling with rules*. Universitat Politècnica de Catalunya.
- Baduge, S. K., Thilakarathna, S., Perera, J. S., Arashpour, M., Sharafi, P., Teodosio, B., Shringi, A., & Mendis, P. (2022). Artificial intelligence and smart vision for building and construction 4.0: Machine and deep learning methods and applications. *Automation in Construction*, 141, 104440. <https://doi.org/10.1016/j.autcon.2022.104440>
- Bao, F., Schwarz, M., & Wonka, P. (2013). Procedural facade variations from a single layout. *ACM Trans. Graph.*, 32(1), 8:1–8:13.
- Barrio, R. S. (2023). Reimaging Earth. Architecture and the critical and speculative uses of geovisualization. *City, Territory and Architecture*, 10(1), 22. <https://doi.org/10.1186/s40410-023-00206-4>
- Bassier, M., & Vergauwen, M. (2020). Unsupervised reconstruction of Building Information Modeling wall objects from point cloud data. *Automation in Construction*, 120, 103338. <https://doi.org/https://doi.org/10.1016/j.autcon.2020.103338>
- Becker, S. (2009). Generation and application of rules for quality dependent façade reconstruction. *ISPRS Journal of Photogrammetry and Remote Sensing*, 64(6), 640–653. <https://doi.org/10.1016/j.isprsjprs.2009.06.002>
- Bekins, D., & Aliaga, D. G. (2005). Build-by-number: Rearranging the real world to visualize novel architectural spaces. *VIS 05. IEEE Visualization, 2005.*, 143–150.
- Berner, A., Bokeloh, M., Wand, M., Schilling, A., & Seidel, H.-P. (2008). A graph-based approach to symmetry detection. *Proc. of the IEEE VGTC Conference on Point-Based Graphics*, 1–8.

- Biljecki, F., Stoter, J., Ledoux, H., Zlatanova, S., & Çöltekin, A. (2015a). Applications of 3D city models: State of the art review. *ISPRS International Journal of Geo-Information*, 4(4), 2842–2889.
- Bokeloh, M., Wand, M., Koltun, V., & Seidel, H.-P. (2011). Pattern-aware shape deformation using sliding dockers. *ACM Trans. Graph.*, 30(6), 123:1–123:10.
- Bokeloh, M., Wand, M., & Seidel, H.-P. (2010a). A connection between partial symmetry and inverse procedural modeling. *ACM Trans. Graph.*, 29(4), 104:1–104:10.
- Bokeloh, M., Wand, M., & Seidel, H.-P. (2010b). A connection between partial symmetry and inverse procedural modeling. In *ACM SIGGRAPH 2010 papers* (pp. 1–10).
- Bokeloh, M., Wand, M., Seidel, H.-P., & Koltun, V. (2012). An algebraic model for parameterized shape editing. *ACM Trans. Graph.*, 31(4), 78:1–78:10.
- Botsch Leif; Pauly Mark; Alliez Pierre; Levy Bruno, M. K. (2010). Polygon Mesh Processing. In *NA: Vol. NA* (Issue NA). A K Peters/CRC Press. <https://doi.org/10.1201/b10688>
- Brenner, C. (2005). Building reconstruction from images and laser scanning. *International Journal of Applied Earth Observation and Geoinformation*, 6(3–4), 187–198.
- Brenner, C. (2010). Building extraction. *Airborne and Terrestrial Laser Scanning*, 36, 169–212.
- Buyuksalih, G., Baskaraca, P., Bayburt, S., Buyuksalih, I., & Abdul Rahman, A. (2019). 3D CITY MODELLING OF ISTANBUL BASED ON LIDAR DATA AND PANORAMIC IMAGES – ISSUES AND CHALLENGES. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, XLII-4/W12, 51–60. <https://doi.org/10.5194/isprs-archives-XLII-4-W12-51-2019>
- Castagno, J., Ochoa, C., & Atkins, E. (2018). Comprehensive Risk-based Planning for Small Unmanned Aircraft System Rooftop Landing. *2018 International Conference on Unmanned Aircraft Systems (ICUAS)*, 1031–1040. <https://doi.org/10.1109/ICUAS.2018.8453483>
- Charniak, E. (1996). AAI/IAAI, Vol. 2 - Tree-bank Grammars. *NA, NA(NA)*, 1031–1036. <https://doi.org/NA>
- Chen, K., Lu, W., Xue, F., Tang, P., & Li, L. H. (2018). Automatic building information model reconstruction in high-density urban areas: Augmenting multi-source data with architectural knowledge. *Automation in Construction*, 93, 22–34. <https://doi.org/https://doi.org/10.1016/j.autcon.2018.05.009>
- Chomsky, N. (1956). Three models for the description of language. *IEEE Transactions on Information Theory*, 2(3), 113–124. <https://doi.org/10.1109/tit.1956.1056813>
- Chomsky, N. (1959). On certain formal properties of grammars. *Information and Control*, 2(2), 137–167. [https://doi.org/10.1016/s0019-9958\(59\)90362-6](https://doi.org/10.1016/s0019-9958(59)90362-6)
- Correia, J., Baeta, F., & Martins, T. (2023). Evolutionary Generative Models. In *Handbook of Evolutionary Machine Learning* (pp. 283–329). Springer.
- da Silveira, L. G., & Musse, S. R. (2006). Real-Time Generation of Populated Virtual Cities. *Proceedings of the ACM Symposium on Virtual Reality Software and Technology*, 155–164. <https://doi.org/10.1145/1180495.1180527>
- De Santis, R., Gloria, A., Viglione, S., Maietta, S., Nappi, F., Ambrosio, L., & Ronca, D. (2018). 3D laser scanning in conjunction with surface texturing to evaluate shift and reduction of the tibiofemoral contact area after meniscectomy. *Journal of the Mechanical Behavior of Biomedical Materials*, 88, 41–47.
- Dehbi, Y., Hadiji, F., Gröger, G., Kersting, K., & Plümer, L. (2017). Statistical Relational Learning of Grammar Rules for 3D Building Reconstruction. *Transactions in GIS*, 21(1), 134–150. <https://doi.org/10.1111/tgis.12200>
- Demir, I., Aliaga, D., & Benes, B. (2014). Proceduralization of buildings at city scale. *3DV*, 456–463.

- Demir, I., & Aliaga, D. G. (2018). Guided proceduralization: Optimizing geometry processing and grammar extraction for architectural models. *Computers & Graphics*, 74, 257–267.
- Demir, I., Aliaga, D. G., & Benes, B. (2015a). Coupled segmentation and similarity detection for architectural models. *ACM Trans. Graph.*, 34(4), 104:1–104:11.
- Demir, I., Aliaga, D. G., & Benes, B. (2015b). Procedural editing of 3d building point clouds. *International Conference on Computer Vision (ICCV)*.
- Engel Jürgen, J. D. (2012). Immersive Visualization of Virtual 3D City Models and its Applications in E-Planning. *International Journal of E-Planning Research*, 1(4), 17–34. <https://doi.org/10.4018/ijepr.2012100102>
- Esch, T., Zeidler, J., Palacios-Lopez, D., Marconcini, M., Roth, A., Mönks, M., Leutner, B., Brzoska, E., Metz-Marconcini, A., Bachofer, F., Loekken, S., & Dech, S. (2020). Towards a Large-Scale 3D Modeling of the Built Environment—Joint Analysis of TanDEM-X, Sentinel-2 and Open Street Map Data. *Remote Sensing*, 12(15). <https://doi.org/10.3390/rs12152391>
- Fatima, I., & Sooda, K. (2023). Enhancing Immersive Virtual Reality Product Experiences: Strategies for Graphics Quality, Performance Optimization, and User-Centric Interfaces. *2023 IEEE Engineering Informatics*, 1–5.
- Fletcher, D., Yue, Y., & Kader, M. Al. (2010). Challenges and Perspectives of Procedural Modelling and Effects. *2010 14th International Conference Information Visualisation*, 543–550. <https://doi.org/10.1109/IV.2010.80>
- Gürsel Dino, \.Ipek. (2012). *Creative design exploration by parametric generative systems in architecture*.
- Haala, N., & Kada, M. (2010). An update on automatic 3D building reconstruction. *ISPRS Journal of Photogrammetry and Remote Sensing*, 65(6), 570–580.
- Haegler, S., Müller, P., & Van Gool, L. (2009). Procedural modeling for digital cultural heritage. *EURASIP Journal on Image and Video Processing*, 2009, 1.
- Healey, P. (2016). Re-thinking key dimensions of strategic spatial planning: sustainability and complexity. In *Fuzzy Planning* (pp. 21–41). Routledge.
- Henn Christoph; Gröger Gerhard; Plümer Lutz, A. R. (2011). Automatic classification of building types in 3D city models. *GeoInformatica*, 16(2), 281–306. <https://doi.org/10.1007/s10707-011-0131-x>
- Jaalama, K., Fagerholm, N., Julin, A., Virtanen, J.-P., Maksimainen, M., & Hyypä, H. (2021). Sense of presence and sense of place in perceiving a 3D geovisualization for communication in urban planning – Differences introduced by prior familiarity with the place. *Landscape and Urban Planning*, 207, 103996. <https://doi.org/10.1016/j.landurbplan.2020.103996>
- Jarząbek-Rychard, M., & Borkowski, A. (2016). 3D building reconstruction from ALS data using unambiguous decomposition into elementary structures. *ISPRS Journal of Photogrammetry and Remote Sensing*, 118, 1–12. <https://doi.org/https://doi.org/10.1016/j.isprsjprs.2016.04.005>
- Jesus Gustavo; Coelho António; Sousa A. Augusto, D. P. (2018). Generalized selections for direct control in procedural buildings. *Computers & Graphics*, 72(NA), 106–121. <https://doi.org/10.1016/j.cag.2018.02.003>
- Jiang Dong-Ming; Zhang Xiaopeng; Wonka Peter, H. Y. (2018). Selection Expressions for Procedural Modeling. *IEEE Transactions on Visualization and Computer Graphics*, 26(4), 1775–1788. <https://doi.org/10.1109/tvcg.2018.2877614>
- Kalogerakis, E., Chaudhuri, S., Koller, D., & Koltun, V. (2012). A probabilistic model for component-based shape synthesis. *ACM Trans. Graph.*, 31(4), 55:1–55:11.
- Kalogerakis, E., Hertzmann, A., & Singh, K. (2010). Learning 3d mesh segmentation and labeling. *ACM Trans. Graph.*, 29(4), 102:1–102:12.

- Kamra, V., Kudeshia, P., ArabiNaree, S., Chen, D., Akiyama, Y., & Peethambaran, J. (2023). Lightweight Reconstruction of Urban Buildings: Data Structures, Algorithms, and Future Directions. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 16, 902–917. <https://doi.org/10.1109/JSTARS.2022.3232758>
- Kanda, J., He, Y., Xie, H., & Miyata, K. (2024). Sketch2Tooncity: sketch-based city generation using neurosymbolic model. *International Workshop on Advanced Imaging Technology (IWAIT) 2024*, 13164, 431–436.
- Khayyal, H. K., Zeidan, Z. M., & Beshr, A. A. A. (2022). Creation and Spatial Analysis of 3D City Modeling based on GIS Data. *Civil Engineering Journal*, 8(1), 105–123. <https://doi.org/10.28991/CEJ-2022-08-01-08>
- Kikuchi, T., Fukuda, T., & Yabuki, N. (2023). Development of a synthetic dataset generation method for deep learning of real urban landscapes using a 3D model of a non-existing realistic city. *Advanced Engineering Informatics*, 58, 102154. <https://doi.org/10.1016/j.aei.2023.102154>
- Kim, J.-S., Kavak, H., & Crooks, A. (2018). Procedural City Generation beyond Game Development. *SIGSPATIAL Special*, 10(2), 34–41. <https://doi.org/10.1145/3292390.3292397>
- Knuth, D. E. (1968). Semantics of context-free languages. *Mathematical Systems Theory*, 2(2), 127–145. <https://doi.org/10.1007/bf01692511>
- Knuth, D. E. (1971). Top-down syntax analysis. *Acta Informatica*, 1(2), 79–110. <https://doi.org/10.1007/bf00289517>
- Kok, S., & Domingos, P. (2005). Learning the structure of Markov logic networks. *Proceedings of the 22nd International Conference on Machine Learning*, 441–448.
- Krecklau Darko; Kobbelt Leif, L. P. (2010). Generalized Use of Non-Terminal Symbols for Procedural Modeling. *Computer Graphics Forum*, 29(8), 2291–2303. <https://doi.org/10.1111/j.1467-8659.2010.01714.x>
- Krecklau Leif, L. K. (2012). SMI 2012: Full Interactive modeling by procedural high-level primitives. *Computers & Graphics*, 36(5), 376–386. <https://doi.org/10.1016/j.cag.2012.03.028>
- Kubicka-Sowińska, A., Misk, Ł., Zachar, P., Fijałkowska, A., Ostrowski, W., Modrzewski, J., & Papuci-Władyka, E. (2024). Reconstructing the Urban Fabric of Nea Paphos by Comparison with Regularly Planned Mediterranean Cities, Using 3D Procedural Modeling and Spatial Analysis. *Bulletin of the American Society of Overseas Research*, 391, 163–189. <https://doi.org/10.1086/730152>
- Leroux, F., Germain, M., Clabaut, É., Bouroubi, Y., & St-Pierre, T. (2024). Improving Three-Dimensional Building Segmentation on Three-Dimensional City Models through Simulated Data and Contextual Analysis for Building Extraction. *ISPRS International Journal of Geo-Information*, 13(1), 20. <https://doi.org/10.3390/ijgi13010020>
- Lienhard Cheryl; Müller Pascal; Wonka Peter; Pauly Mark, S. L. (2017). Design Transformations for Rule-based Procedural Modeling. *Computer Graphics Forum*, 36(2), 39–48. <https://doi.org/10.1111/cgf.13105>
- Lin, J., Cohen-Or, D., Zhang, H., Liang, C., Sharf, A., Deussen, O., & Chen, B. (2011). Structure-preserving retargeting of irregular 3d architecture. *ACM Trans. Graph.*, 30(6), 183:1–183:10.
- LinJinjie ; ZhangHao ; LiangCheng ; SharfAndrei ; DeussenOliver ; ChenBaoquan, ; Cohen-OrDaniel. (2011). Structure-preserving retargeting of irregular 3D architecture. *ACM Transactions on Graphics*, 30(6), 1–10. <https://doi.org/10.1145/2070781.2024217>
- Lipp, M., Wonka, P., & Wimmer, M. (2008). Interactive visual editing of grammars for procedural architecture. *ACM Trans. Graph.*, 27(3), 102:1–102:10.

- Lipp Matthias; Lau Cheryl; Wonka Peter; Müller Pascal, M. S. (2019). Local Editing of Procedural Models. *Computer Graphics Forum*, 38(2), 13–25.
<https://doi.org/10.1111/cgf.13615>
- Lipp Peter; Wimmer Michael, M. W. (2008). Interactive visual editing of grammars for procedural architecture. *ACM Transactions on Graphics*, 27(3), 102–110.
<https://doi.org/10.1145/1360612.1360701>
- Liu, R., Li, H., & Lv, Z. (2023). Modeling Methods of 3D Model in Digital Twins. *Computer Modeling in Engineering & Sciences*, 136(2), 985–1022.
<https://doi.org/10.32604/cmes.2023.023154>
- Maim, J., Haegler, S., Yersin, B., Mueller, P., Thalmann, D., & Van Gool, L. (2007). Populating Ancient Pompeii with Crowds of Virtual Romans. *Proceedings of the 8th International Conference on Virtual Reality, Archaeology and Intelligent Cultural Heritage*, 109–116.
- Martin, I., & Patow, G. (2019). Ruleset-rewriting for procedural modeling of buildings. *Computers & Graphics*, 84, 93–102.
- Martinovic, A., & Van Gool, L. (2013). Bayesian grammar learning for inverse procedural modeling. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 201–208.
- Mathias, M., Martinovic, A., Weissenberg, J., & Van Gool, L. (2011). Procedural 3D building reconstruction using shape grammars and detectors. *2011 International Conference on 3D Imaging, Modeling, Processing, Visualization and Transmission*, 304–311.
- MEGA. (n.d.). *Supplement material*. Retrieved October 19, 2024, from https://mega.nz/file/YchWCSbD#D6Gxl9SwBYqzyUG6OpE4J_ocY06RgnopSurEuGBdiqM
- Mitra, N., Wand, M., Zhang, H. R., Cohen-Or, D., Kim, V., & Huang, Q.-X. (2013). Structure-aware shape processing. *SIGGRAPH Asia 2013 Courses*, 1:1–1:20.
- Müller Gang; Wonka Peter; Van Gool Luc, P. Z. (2007). Image-based procedural modeling of facades. *ACM Transactions on Graphics*, 26(3), 85.
<https://doi.org/10.1145/1276377.1276484>
- Müller, P., Vereenoghe, T., Wonka, P., Paap, I., & Van Gool, L. (2006). Procedural 3D Reconstruction of Puuc Buildings in Xkipché. *VAST, 2006*, 7th.
- Müller, P., Wonka, P., Haegler, S., Ulmer, A., & Van Gool, L. (2006). Procedural modeling of buildings. *ACM Transactions on Graphics*, 25(3), 614–623.
<https://doi.org/10.1145/1141911.1141931>
- Müller, P., Zeng, G., Wonka, P., & Van Gool, L. (2007). Image-based procedural modeling of facades. *ACM Trans. Graph.*, 26(3), 85.
- Müller Peter; Haegler Simon; Ulmer Andreas; Van Gool Luc, P. W. (2006). Procedural modeling of buildings. *ACM Transactions on Graphics*, 25(3), 614–623.
<https://doi.org/10.1145/1141911.1141931>
- Musgrave, F. K., Peachey, D., Perlin, K., & Worley, S. (1994). *Texturing and modeling: a procedural approach*. Academic Press Professional, Inc.
- Musialski Peter; Aliaga Daniel G.; Wimmer Michael; Van Gool Luc; Purgathofer Werner, P. W. (2012). Eurographics (State of the Art Reports) - A Survey of Urban Reconstruction. *NA, NA(NA)*, 1–28. <https://doi.org/NA>
- Nishida, G., Garcia-Dorado, I., Aliaga, D. G., Benes, B., & Bousseau, A. (2016a). Interactive sketching of urban procedural models. *ACM Trans. Graph.*
- Nishida, G., Garcia-Dorado, I., Aliaga, D. G., Benes, B., & Bousseau, A. (2016b). Interactive sketching of urban procedural models. *ACM Transactions on Graphics (TOG)*, 35(4), 1–11.

- Nishida Ignacio; Aliaga Daniel G.; Benes Bedrich; Bousseau Adrien, ; Garcia-Dorado. (2016). Interactive sketching of urban procedural models. *ACM Transactions on Graphics*, 35(4), 111–130. <https://doi.org/10.1145/2897824.2925951>
- Oketunji, F. (2023). Evaluating the Efficacy of Hybrid Deep Learning Models in Distinguishing AI-Generated Text. *ArXiv Preprint ArXiv:2311.15565*.
- Olsen, J. (2004). *Realtime procedural terrain generation*.
- Panteli, C., Kylili, A., & Fokaides, P. A. (2020). Building information modelling applications in smart buildings: From design to commissioning and beyond A critical review. *Journal of Cleaner Production*, 265, 121766.
- Parish, Y. I. H., & Müller, P. (2001). Procedural modeling of cities. *Proceedings of the 28th Annual Conference on Computer Graphics and Interactive Techniques*, 301–308.
- Patow, G. (2012). User-Friendly Graph Editing for Procedural Modeling of Buildings. *IEEE Computer Graphics and Applications*, 32(2), 66–75. <https://doi.org/10.1109/mcg.2010.104>
- Patterson, B., & Ward, M. (2022). *Adaptive Procedural Generation in Minecraft*.
- Pauly, M., Mitra, N. J., Wallner, J., Pottmann, H., & Guibas, L. J. (2008). Discovering structural regularity in 3d geometry. *ACM Trans. Graph.*, 27(3), 43:1–43:11.
- Prusinkiewicz, P., & Lindenmayer, A. (1990). *The Algorithmic Beauty of Plants*. Springer-Verlag New York, Inc.
- Qin, J., Fang, C., Wang, Y., Li, G., & Wang, S. (2015). Evaluation of three-dimensional urban expansion: A case study of Yangzhou City, Jiangsu Province, China. *Chinese Geographical Science*, 25(2), 224–236. <https://doi.org/10.1007/s11769-014-0728-8>
- Qin, X., Mao, W., Hu, Z., Zheng, H., & Xu, X. (2023). Procedural modeling and layout method for a generic ancient Chinese city. *Multimedia Tools and Applications*, 83(16), 47021–47048. <https://doi.org/10.1007/s11042-023-16942-1>
- Richardson Pedro, M. D. (2006). Markov logic networks. *Machine Learning*, 62(1), 107–136. <https://doi.org/10.1007/s10994-006-5833-1>
- Ripperda, N. (2008). Determination of facade attributes for facade reconstruction. *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, 37(B3a), 285–290.
- Ripperda, N., & Brenner, C. (2009). Application of a formal grammar to facade reconstruction in semiautomatic and automatic environments. *Proc. of the 12th AGILE Conference on GIScience*, 1–12.
- Ritchie Ben; Goodman Noah D.; Hanrahan Pat, D. M. (2015). Controlling procedural modeling programs with stochastically-ordered sequential Monte Carlo. *ACM Transactions on Graphics*, 34(4), 105–111. <https://doi.org/10.1145/2766895>
- Ritchie, D., Mildenhall, B., Goodman, N. D., & Hanrahan, P. (2015). Controlling procedural modeling programs with stochastically-ordered sequential monte carlo. *ACM Transactions on Graphics (TOG)*, 34(4), 1–11.
- Rozenberg, G. (1997). Handbook of Graph Grammars and Computing by Graph Transformation: Volume 1: Foundations - Handbook of graph grammars and computing by graph transformation: volume I. foundations. In *NA: Vol. NA* (Issue NA). WORLD SCIENTIFIC. <https://doi.org/10.1142/3303>
- Ruizhongtai Qi, C. (2020). Deep learning on 3D data. *3D Imaging, Analysis and Applications*, 513–566.
- Saldana, M., & Johanson, C. (2013). PROCEDURAL MODELING FOR RAPID-PROTOTYPING OF MULTIPLE BUILDING PHASES. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, XL-5/W1, 205–210. <https://doi.org/10.5194/isprsarchives-XL-5-W1-205-2013>

- Satari, M., Samadzadegan, F., Azizi, A., & Maas, H.-G. (2012). A multi-resolution hybrid approach for building model reconstruction from lidar data. *The Photogrammetric Record*, 27(139), 330–359.
- Scheler, C. (n.d.). *Image| Band 230*.
- Schwarz, M., & Müller, P. (2015a). Advanced procedural modeling of architecture. *ACM Trans. Graph.*, 34(4).
- Schwarz, M., & Müller, P. (2015b). Advanced procedural modeling of architecture. *ACM Transactions on Graphics (TOG)*, 34(4), 1–12.
- Schwarz Pascal, M. M. (2015). Advanced procedural modeling of architecture. *ACM Transactions on Graphics*, 34(4), 107–112. <https://doi.org/10.1145/2766956>
- Sensmeier, J. (2017). Harnessing the power of artificial intelligence. *Nursing Management*, 48(11), 14–19. <https://doi.org/10.1097/01.NUMA.0000526062.69220.41>
- Simon, L. (2011). *Procedural reconstruction of buildings: towards large scale automatic 3D modeling of urban environments*. Ecole Centrale Paris.
- Simon, L., Teboul, O., Koutsourakis, P., & Paragios, N. (2011). Random exploration of the procedural space for single-view 3d modeling of buildings. *International Journal of Computer Vision*, 93(2), 253–271.
- Singla, P., & Domingos, P. (2006). Entity resolution with markov logic. *Sixth International Conference on Data Mining (ICDM'06)*, 572–582.
- Smelik, R. M., Tutenel, T., Bidarra, R., & Benes, B. (2014a). A survey on procedural modelling for virtual worlds. *Comp. Graph. Forum*, 33(6), 31–50.
- Smelik, R. M., Tutenel, T., Bidarra, R., & Benes, B. (2014b). A survey on procedural modelling for virtual worlds. *Computer Graphics Forum*, 33(6), 31–50.
- Somanath, S., Naserentin, V., Eleftheriou, O., Sjölie, D., Wästberg, B. S., & Logg, A. (2024). Towards Urban Digital Twins: A Workflow for Procedural Visualization Using Geospatial Data. *Remote Sensing*, 16(11), 1939. <https://doi.org/10.3390/rs16111939>
- Št'ava, O., Beneš, B., Měch, R., Aliaga, D. G., & Křištof, P. (2010). Inverse procedural modeling by automatic generation of L-systems. *Computer Graphics Forum*, 29(2), 665–674.
- Stiny, G. (1975). *Pictorial and formal aspects of shape and shape grammars*. Birkhäuser.
- Stolcke, A., & Omohundro, S. (1994). Inducing probabilistic grammars by Bayesian model merging. *International Colloquium on Grammatical Inference*, 106–118.
- Sukkar, A. W., Fareed, M. W., Yahia, M. W., Abdalla, S. B., Ibrahim, I., & Senjab, K. A. K. (2024). Analytical Evaluation of Midjourney Architectural Virtual Lab: Defining Major Current Limits in AI-Generated Representations of Islamic Architectural Heritage. *Buildings*, 14(3), 786. <https://doi.org/10.3390/buildings14030786>
- Sun, C., Han, J., Deng, W., Wang, X., Qin, Z., & Gould, S. (2023). 3d-gpt: Procedural 3d modeling with large language models. *ArXiv Preprint ArXiv:2310.12945*.
- Talton, J. O., Lou, Y., Lesser, S., Duke, J., Měch, R., & Koltun, V. (2011). Metropolis procedural modeling. *ACM Trans. Graph.*, 30(2), 11:1–11:14.
- Talton, J., Yang, L., Kumar, R., Lim, M., Goodman, N., & Měch, R. (2012a). Learning design patterns with bayesian grammar induction. *Proceedings of the ACM UIST*, 63–74.
- Talton, J., Yang, L., Kumar, R., Lim, M., Goodman, N., & Měch, R. (2012b). Learning design patterns with bayesian grammar induction. *Proceedings of the 25th Annual ACM Symposium on User Interface Software and Technology*, 63–74.
- Tarsha-Kurdi, F., Landes, T., & Grussenmeyer, P. (2007). Hough-transform and extended ransac algorithms for automatic detection of 3d building roof planes from lidar data. *ISPRS Workshop on Laser Scanning 2007 and SilviLaser 2007*, 36, 407–412.

- Tarsha-Kurdi, F., Landes, T., & Grussenmeyer, P. (2008). Extended RANSAC algorithm for automatic detection of building roof planes from LiDAR data. *The Photogrammetric Journal of Finland*, 21(1), 97–109.
- Teboul, O., Kokkinos, I., Simon, L., Koutsourakis, P., & Paragios, N. (2011). Shape grammar parsing via reinforcement learning. *CVPR 2011*, 2273–2280.
- Teboul, O., Simon, L., Koutsourakis, P., & Paragios, N. (2010). Segmentation of building facades using procedural shape priors. *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 3105–3112.
- Tekavec, J., Lisec, A., & Rodrigues, E. (2020). Simulating Large-Scale 3D Cadastral Dataset Using Procedural Modelling. *ISPRS International Journal of Geo-Information*, 9(10). <https://doi.org/10.3390/ijgi9100598>
- Toshev, A., Mordohai, P., & Taskar, B. (2010). Detecting and parsing architecture at city scale from range data. *Computer Vision and Pattern Recognition*, 398–405.
- Tsochantaridis, I., Hofmann, T., Joachims, T., & Altun, Y. (2004). Support vector machine learning for interdependent and structured output spaces. *Proceedings of the Twenty-First International Conference on Machine Learning*, 104.
- Ullrich, T., Schinko, C., & Fellner, D. W. (2010). *Procedural modeling in theory and practice*.
- Usta, Z., Akin, A. T., & Cömert, Ç. (2023). Deep learning aided web-based procedural modelling of LOD2 city models. *Earth Science Informatics*, 16(3), 2559–2571. <https://doi.org/10.1007/s12145-023-01053-0>
- Usta, Z., Cömert, Ç., & Akin, A. T. (2024). An interoperable web-based application for 3d city modelling and analysis. *Earth Science Informatics*, 17(1), 163–179. <https://doi.org/10.1007/s12145-023-01167-5>
- Vanegas, C. A., Aliaga, D. G., & Benes, B. (2010). Building reconstruction using manhattan-world grammars. *Computer Vision and Pattern Recognition (CVPR)*, 358–365.
- Vanegas, C. A., Aliaga, D. G., Benes, B., & Waddell, P. A. (2009). Interactive design of urban spaces using geometrical and behavioral modeling. *ACM SIGGRAPH Asia 2009 Papers*.
- Vanegas, C. A., Garcia-Dorado, I., Aliaga, D. G., Benes, B., & Waddell, P. (2012). Inverse design of urban procedural models. *ACM Trans. Graph.*, 31(6), 168:1–168:11.
- Vanegas Daniel G.; Wonka Peter; Müller Pascal; Waddell Paul; Watson Benjamin, C. A. ; A. (2010). Modelling the appearance and behaviour of urban spaces. *Computer Graphics Forum*, 29(1), 25–42. <https://doi.org/10.1111/j.1467-8659.2009.01535.x>
- Wang, R. (2013). 3D building modeling using images and LiDAR: A review. *International Journal of Image and Data Fusion*, 4(4), 273–292.
- Watson, B., Müller, P., Veryovka, O., Fuller, A., Wonka, P., & Sexton, C. (2008). Procedural urban modeling in practice. *IEEE Computer Graphics and Applications*, 28(3), 18–26.
- Wonka Michael; Sillion François; Ribarsky William, P. W. (2003). Instant architecture. *ACM Transactions on Graphics*, 22(3), 669–677. <https://doi.org/10.1145/882262.882324>
- Wonka, P., Wimmer, M., Sillion, F., & Ribarsky, W. (2003). Instant architecture. *ACM Trans. Graph.*, 22(3), 669–677.
- Wu, A. N., Stouffs, R., & Biljecki, F. (2022). Generative Adversarial Networks in the built environment: A comprehensive review of the application of GANs across data types and scales. *Building and Environment*, 223, 109477.
- Wu, F., Yan, D.-M., Dong, W., Zhang, X., & Wonka, P. (2013). Inverse procedural modeling of facade layouts. *ArXiv Preprint ArXiv:1308.0419*.
- Wu, F., Yan, D.-M., Dong, W., Zhang, X., & Wonka, P. (2014). Inverse Procedural Modeling of Facade Layouts. *ACM Trans. Graph.*, 33(4). <https://doi.org/10.1145/2601097.2601162>

- Yang, X., & Delparte, D. (2022). A Procedural Modeling Approach for Ecosystem Services and Geodesign Visualization in Old Town Pocatello, Idaho. *Land*, 11(8).
<https://doi.org/10.3390/land11081228>
- Yeguas, E., Muñoz-Salinas, R., & Medina-Carnicer, R. (2012). Example-based procedural modelling by geometric constraint solving. *Multimedia Tools and Applications*, 60(1), 1–30. <https://doi.org/10.1007/s11042-011-0795-0>
- Zhang, H., Xu, K., Jiang, W., Lin, J., Cohen-Or, D., & Chen, B. (2013). Layered analysis of irregular facades via symmetry maximization. *ACM Trans. Graph.*, 32(4), 104:1–10.
- Zhang, M., Wu, J., Liu, Y., Zhang, J., & Li, G. (2022). GIS Based Procedural Modeling in 3D Urban Design. *ISPRS International Journal of Geo-Information*, 11(10), 531.
<https://doi.org/10.3390/ijgi11100531>
- Zheng, Y., Fu, H., Cohen-Or, D., Au, O. K.-C., & Tai, C.-L. (2011). Component-wise controllers for structure-preserving shape manipulation. *Comp Graph. Forum*, 30(2), 563–572.

URL-ovi:

URL 1 <https://www.cs.purdue.edu/cgvlab/papers/aliaga/3DV16.pdf> (14.5.2025.)

URL 2 <https://www.researchgate.net/profile/Celso-Lapa/publication/296639888/figure/fig3/AS:335254394032130@1456942284272/Buildings-of-the-virtual-city-of-the-digital-game.png> (14.5.2025.)

URL 3 <https://www.cs.purdue.edu/cgvlab/papers/aliaga/sig16course.pdf> (14.5.2025.)

URL 4 https://geofhr-my.sharepoint.com/:f:/g/personal/rnikci_geof_hr/EtzgdtIh9HxIjsdgWgq6TRcBw_UcCzPWxA2WBsJjxydAKg?e=2lrpDx (14.5.2025.)

URL 5 https://geofhr-my.sharepoint.com/:w:/g/personal/rnikci_geof_hr/EVNI2QhkWSpDisPUT9z1I8cB52cXgndgeT-s2C1p07TeGw?e=GfQz4O (14.5.2025.)

URL 6 https://geofhr-my.sharepoint.com/:w:/g/personal/rnikci_geof_hr/EWYJSdzddodHmlMixn2CUk8BUH47_XKZOBxyYLZ7YbpXyw?e=BPTsDF (14.5.2025.)

URL 7 <http://www.ikg.uni-bonn.de/forschung/projektarchiv/measurefacade.html> (14.5.2025.)

URL 8. <https://www.arcgis.com/home/item.html?id=0f03323a2fd84da399a27250ceeb8d5e> (14.5.2025.)

URL 9. https://osmot.cs.cornell.edu/svm_cfg/current/svm_cfg_windows.zip (14.5.2025.)

URL 10 www.sketchfab.com (14.5.2025.)

URL 11 <https://en.wikipedia.org/wiki/L-system> (14.5.2025.)

URL 12 <https://hrcak.srce.hr/file/412432> (14.5.2025.)

URL 13 https://geofhr-my.sharepoint.com/:f/g/personal/rnikci_geof_hr/Ehi7Lhc8nrhGiz0Vl9vIhBMB1zExMVCKId2O9AFcDnoaQQ?e=vb8eiK (14.5.2025.)

URL 14 <https://doi.org/10.3390/info14070394> (14.5.2025.)

URL 15 <https://en.wikipedia.org/wiki/Ljubljana> (14.5.2025.)

URL 16 <https://www.danielgm.net/cc/> (14.5.2025.)

URL 17 <https://qgis.org/> (14.5.2025.)

URL 18 <https://plugins.qgis.org/plugins/OSMDDownloader/> (14.5.2025.)

URL 19 <https://doi.org/10.3390/info14070394> (14.5.2025.)

URL 20 https://geofhr-my.sharepoint.com/:f/g/personal/rnikci_geof_hr/EjDSM6PFoolMipwsmh0lKOoBicMxThckCJkcOYhGB48iWg?e=xMF8vN (14.5.2025.)

URL 21 <https://rapidlasso.com/lastools/> (14.5.2025.)

URL 22 <https://doi.org/10.3390/info14070394> (14.5.2025.)

URL 23 <https://www.sidefx.com/products/houdini/> (14.5.2025.)

URL 24 <https://www.esri.com/en-us/arcgis/products/arcgis-cityengine/> (14.5.2025.)

URL 25 <https://unity.com/> (14.5.2025.)

URL 26 <https://www.autodesk.com/products/maya/overview> (14.5.2025.)

URL 27 <https://www.blender.org/> (14.5.2025.)

URL 28 <https://www.geopipe.ai/about> (14.5.2025.)

URL 29 <https://www.pny.com/en-eu/professional/software/nvidia-omniverse-prod>
(14.5.2025.)

URL 30 <https://www.mapbox.com/> (14.5.2025.)

URL 31 <https://cesium.com/> (14.5.2025.)

URL 32 <https://github.com/domlysz/BlenderGIS> (14.5.2025.)

URL 33 <https://cementconcrete.org/building-construction/functional-components-building-structure/3246/> (14.5.2025.)

URL 34 <https://github.com/prochitecture/bpypolyskel> (14.5.2025.)

URL 35 <https://doi.org/10.3390/app14188345> (14.5.2025.)

Životopis

Rexhep Nikçi rođen je 04.10.1967. u Peći. On je državljanin Republike Kosovo. Nakon završene osnovne škole u Peći, nastavlja školovanje u srednjoj Geodetskoj školi u *Gjakovë* (Đakovici). U akademskoj godini 1987./1988. upisuje Geodetski Fakultet u Zagrebu. Obranom diplomskog rada pod mentorstvom prof.emer.dr.sc. Zdravka Kapovića 1994. stječe akademski naziv diplomirani inženjer geodezije.

Potom odlazi na odsluženje vojnog roka u Jastrebarskom, a potom se 1994-1999 *Vermessungsbüro Zeiss*, a 1999-2007. radi u *Vermessungsbüro Widman*. Nakon toga od 2007. do danas radi u vlastitom geodetskom uredu ovlaštenog inženjera geodezije *Vermessungsbüro Nikçi*.

Član je Bavarske komore ovlaštenih inženjera geodezije (*Ingenieurverband Geoinformation und Vermessung Bayern e.V.*) od 2006. Objavio je u koautorstvu 5 radova u časopisima uz aktivno sudjelovanje na više skupovima i konferencijama. Od 1995. živi i radi kao geodet u Münchenu. Od 2001. godine djeluje na projektima, seminarima i predavanjima na Kosovu s temama iz geodeziju i geoinformatike. Oženjen je i ima dvoje djece.

Popis objavljenih radova u časopisima:

Nikçi, R., Župan, R., & Racetin, I. (2024). Geovisualization of Buildings: AI vs. Procedural Modeling. *Applied Sciences* (2076-3417), 14(18).

Župan, R., Vinković, A., **Nikçi, R.**, & Pinjatela, B. (2023). Automatic 3D building model generation from airborne LiDAR data and OpenStreetMap using procedural modeling. *Information*, 14(7), 394.

Nikçi, R., Nikçi, F., Shehu, F., & Župan, R. (2022). Prijedlozi poboljšanja automatizacije modeliranja krovova postupcima proceduralnog modeliranja. *Geodetski list*, 76(3), 195-212.

Nikçi, R. (2019). Land consolidation for the future regulation of the Kosovo's land. *Journal of Natural and Technical Sciences*, 2(47), 333-344.

Nikçi, R. (2018). Regulation of agricultural land as a factor for agricultural development. *Journal of Natural and Technical Sciences*, 2(47), 139-146.

Nikçi, R. (2006). Untersuchung der Höhengenaugigkeit von GPS-Messungen im Flusstal der Großen Laaber. *VDV-Magazin*, 5, 385-389.

Na skupovima i konferencijama:

Nikçi, R., Župan, R., & Znaor, M. (2022). Procedural modelling of buildings and visualization. *International Multidisciplinary Scientific GeoConference: SGEM*, 22(6.2), 591-598.

Grgić, Š., Grgić, I., & **Nikci, R.** (2019). Mogući geodetski doprinos minimaliziranju konflikata tijekom građenja. In 12. simpozij ovlaštenih inženjera geodezije-Etika u struci (pp. 98-106).