

TARTU ÜLIKOOL
Arvutiteaduse instituut
Informaatika õppekava

Joosep Suuder

**Manimi abil loodud õppevideod põhikooli
matemaatikast**

Bakalaureusetöö (9 EAP)

Juhendaja: Kati Ain, PhD

Kaasjuhendaja: Toomas Krips, PhD

Manimi abil loodud õppevideod põhikooli matemaatikast

Lühikokkuvõte:

Käesoleva bakalaureusetöö eesmärk on luua matemaatika õppevideosid põhikooli õppeastmele kasutades Pythoni programmeerimiskeele Manim teeki. Valmis videod on mõeldud toetama põhikooli õppekavas õpetatavaid teemasid, kasutades visuaalseid elemente, et soodustada materjalide omandamist ja nendest arusaamist.

Töö sisaldab visuaalse õppe olemuse uurimist, eesmärkide ja videote tegemisel järgivate tingimuste paika panemist, kasutatud tehnoloogiate ülevaadet ning tulemuste analüüsi.

Võtmesõnad:

Python, Manim, visuaalne õppimine, YouTube, põhikool

CERCS: P175 Informaatika, süsteemiteooria, S281 Arvuti õpiprogrammide kasutamise metoodika ja pedagoogika

Educational Videos About Elementary School Mathematics Created With Manim

Abstract:

The purpose of this bachelor's thesis is to create mathematics educational videos for the elementary school level using the Manim library of the Python language. The created videos are meant to support the teaching of concepts that are handled in the primary school curriculum, using visual elements to aid in the memorization and understanding of the aforementioned.

This work explores the nature of visual learning, sets goals and requirements for the created videos, gives an overview of the used technologies and analyzes the results.

Keywords:

Python, Manim, visual learning, YouTube, primary school

CERCS: P175 Informatics, systems theory, S281 Computer-assisted education

Sisukord

Sissejuhatus	4
Mõisted ja terminid	5
1. Visuaalne õpe	6
1.1 Õppijate tüübid	6
1.2 Visuaalse õppe tähtsus	6
1.3 Visuaalne õpe matemaatikas	7
2. Eesmärkide ja nõuete kaardistamine	9
2.1 Lõputöö eesmärgid	9
2.2 Nõuded loodud animatsioonidele	10
3. Kasutatud tehnoloogiad ja tööprotsess	12
3.1 Manim	12
3.1.1 Manimi loomisest	13
3.1.2 Animatsioonide loomine Manimi abil	13
3.1.3 Manimi sisemine töö	17
3.2 VSCode liides Manim Sideview	18
3.3 iMovie	19
3.4 Ideest valmis videoni	20
4. Tulemused	22
4.1 Valminud videod	22
4.2 Tulemuste analüüs	24
5. Kokkuvõte	26
Viidatud kirjandus	27
Lisad	30
Lisa 1. Litsents	30

Sissejuhatus

Iga päevaga mängivad digiajastu arengud üha suuremat rolli kõikides meie eluvaldkondades ning sellest ei pääse ka õpetamine ja õppematerjalide loomine. Mida aeg edasi, seda varasemalt hakkavad lapsed internetti kasutama [1, 2]. Üheks võimalikuks sisuks, mida lapsed tänapäeval internetist tarbivad on õppematerjalid, mis ei pruugi aga alati olla kvaliteetsed, sest internetti võib sisu luua igäüks. Bakalaureusetöö eesmärk on luua põhikoolile eestikeelseid matemaatika õppevideoid, kasutades ära informaatika valdkonnas loodud tööriistu ja nende hüvesid. Lisaks õppevideote loomisele, tahetakse tagada nende kvaliteetsus ja relevantsus õppematerjalina. Eesmärgi täitmiseks on valitud Pythoni¹ programmeerimiskeele teek Manim, mis laseb luua koodi abil animatsioone ning on suunatud just matemaatika teemalistele animatsioonidele.

Teoreetilise osa esimene peatükk annab ülevaate visuaalsest õppest üldiselt, selle tähtsusest ja kasutamisest matemaatikas. Tuuakse välja üks õppijate liigitamise viis ning argument, miks visuaalne õppevorm on tähtis. Lisaks esitatakse näiteid sellest, kuidas traditsioonilises kooliõppes visuaalseid elemente kasutatakse ja mis on nende tugevad ja nõrgad küljed.

Teises peatükis sõnastatakse töö praktilise osa eesmärgid ning uuritakse, millised peaksid olema ühe efektiivse õppevideo tunnused. Vastavalt välja selgitatud tunnustele seatakse videotele tingimused, mille abil kindlustatakse videote kvaliteet õppematerjalidena.

Kolmandas peatükis antakse ülevaade sellest, milliseid tehnoloogiaid töö praktilise osa valmimise käigus kasutati ning selgitatakse, miks tehti just sellised valikud. Kolmanda peatüki esimene alapeatükk annab ülevaate Manim teegist, selle ajaloost ning sellest, miks Manim teek osutus parimaks valikuks matemaatiliste animatsioonide loomisel. Järgnevates alapeatükkides käsitletakse tähtsamaid tööriistu lisaks Manim teegile. Samuti põhjendatakse, miks sellised valikud tehti.

Viimaseks analüüsitakse töö tulemusi, tuuakse välja mõned olulisemad osad valminud videotest ning arutletakse võimalike edasiarenduste üle.

¹ Pythoni programmeerimiskeel. <https://www.python.org/>.

Mõisted ja terminid

Teek (ingl *library*) – Infoobjektide kogu, üldiseks korduvaks kasutamiseks.²

Renderdama (ingl *to render*) – Renderdamine on arvutiprogrammis toimuv protsess, mille käigus graafilisest mudelist genereeritakse kahemõõtmeline kujutis.³

Alfa põlvkond (ingl *Generation Alpha*) – Alfa põlvkonda kuuluvad inimesed, kes on sündinud aastatel 2010–2024.⁴

Varjutaja (ingl *shader*) – Varjutamine on 3-mõõtmelises arvutigraafikas väike programm või algoritmide komplekt, mis määrab ära objektide 3-mõõtmeliste pinnaomaduste kujutamise viisi sõltuvalt valgusallikast, selle kaugusest ja valguse langemisnurgast.⁵

² Pärineb andmekaitse ja infoturbe portaalist AKIT <https://akit.cyber.ee/>.

³ Pärineb Eesti Keele Instituudi Sõnaveebist <https://pre.sonaveeb.ee/>.

⁴ Pärineb termini autori, McCrindle M., artiklist, *Understanding Generation Alpha*.
<https://mccrindle.com.au/article/topic/generation-alpha/generation-alpha-defined>

⁵ Pärineb Eesti Keele Instituudi Sõnaveebist <https://pre.sonaveeb.ee/>.

1. Visuaalne õpe

Käesolevas peatükis kirjeldatakse õppijate tüüpe, antakse lühike ülevaade visuaalsest õppest ja selle mõjust õppijate tähelepanule, informatsiooni talletamisele ja õppematerjalidest aru saamisele. Arutletakse visuaalsete elementide kasutamise väärtuse üle ning uuritakse lähemalt visuaalsete õppevormide mõju ja kasutust matemaatika õppes.

1.1 Õppijate tüübid

Hetkeseisuga ei ole ühte kindlat aktsepteeritud teooriat õppijate tüüpide kohta, kuid on välja pakutud mitmeid erinevaid viise õpistiilide eristamiseks, millest enamusel esineb rohkem sarnasusi, kui erinevusi [3].

Üks võimalik õppijate tüüpide liigitus on: visuaalsed õppijad, kuulmispõhised õppijad ja kinesteetilised õppijad [4]. Visuaalsed õppijad õpivad kõige efektiivsemalt nähes, kuulmispõhised kuulates ning kinesteetilised õppijad tegutsedes [4].

Visuaalse õppimise põhimõte seisneb ideede ja informatsiooni edasi andmises, kasutades visuaalseid abivahendeid nagu graafikud, pildid, slaidid või videod, et hõlbustada mitte ainult meeldejätmist, vaid ka sügavamalt arusaama õpetatavatest kontseptsioonidest [5]. Kuigi õppijaid liigitatakse nende poolt eelistatud õppe tüüpide põhjal erinevalt, saavad visuaalsete elementide kasutusest kasu kõik erinevad õppijate tüübid [4, 6]. On oluline tähele panna, et ühe õppija tüübi tunnuste esinemine ei vähenda teiste õppija tüüpide tunnuste toetamise kasu – ideaaljuhul tuleks kasutada kõiki õppijate tüüpe toetavaid elemente [3].

1.2 Visuaalse õppe tähtsus

Juba lasteaias kasutatakse visuaalset õpet, näiteks näpunukkude ja juturaamatute näol, et hoida ja pikendada laste tähelepanu ning süvendada õppematerjalide omandamist [7]. Visuaalne õppevorm muutub tehnoloogia ajastul järjest kättesaadavamaks ja seda ka üha nooremale eale. Näiteks leiti Soomes läbiviidud uuringus [1], et 64% 7-aastastest kasutab internetti. Vinter [2] leidis, et 61-st vastanud eelkooliealisest lapsest ei kasutanud enda sõnul arvutit regulaarselt vaid kaks, lisaks sellele märkisid fookusgruppides osalenud õpetajad, et huvi arvutite vastu näitavad lapsed välja juba kolmandal eluaastal. See viitab sellele, et aina enam on väärtust vabavaralistel

õppematerjalidel, mis on lihtsasti kättesaadavad. Sellele annab toetust ka Cimene jt. uuring [4], kus leiti, et alfa põlvkonna koolilapsed eelistavad visuaalseid õppevorme. Ühtlasi tehti kindlaks, et ka kuulamis põhisele õppevormile lisab väärtust visuaalne element. Lisaks leidis Bartan [7], et visuaalsed õppevormid hoidsid enim lasteaialaste tähelepanu ning olid informatsiooni edasi andmises kõige tõhusamad.

Visuaalsete vahendite kasutust traditsioonilises õppes on uurinud Irani jt. [8], kes leidsid, et enim kasutatud visuaalne õppevahend traditsioonilises õppes on tahvli peale joonistamine/kirjutamine. Tahvli kasutamise efektiivsusest visuaalse õppevahendina nendib ka Ostling [9], tuues välja, et tahvlile joonistamine viib õpetamise tempo õppijatele sobivamale kiirusele ning annab õppijatele võimaluse käsitsi märkmeid teha⁶ – see on ehe näide sellest, kuidas erinevaid õppevorme kasutades, saab võimaldada paremaid tingimused õppe toetamiseks.

1.3 Visuaalne õpe matemaatikas

Visuaalsete elementide kasutamine matemaatika õppematerjalides mõjub õpitulemustele positiivselt ja vähendab kognitiivset koormust õpilaste seas [10]. Animatsioonide kasutamist õppevahendina kirjeldab Gambari jt. artikkel [11], kus leiti, et animatsioonidel on matemaatiliste kontseptsioonide õppimisel positiivne mõju nii meelespidamisele kui ka omandamisele. Schoenherr jt. läbi viidud metaanalüüsis [12], mis hõlmas 41 uuringut, leiti, et visuaalsete elementide kasutamine matemaatika õppes andis paremaid tulemusi võrreldes nende mittekasutamisega. Lisaks leiti, et tehnoloogia roll visuaalsetes õppevahendites, s.t tehnoloogia abil valmistatud visuaalsed abivahendid, ei andnud erilist eelist võrreldes analoogmeetoditel loodutega [12].

Tehnoloogia tugevused ei pruugi alati väljenduda otseses võrdluses, vaid suurimat kasu näeme selles, et tehnoloogia abil loodud visuaalsed õppevahendid annavad võimaluse teha õppevahendid kättesaadavamaks, kui analoogmeetoditega on võimalik. Lisaks sellele annab tehnoloogia kasutamine meile võimaluse kiiremini ja efektiivsemalt luua kvaliteetseid õppevahendeid. Võttes näiteks selle lõputöö raames loodavad videod, siis saab neid vaadata lugematu arv kordi, panna

⁶ Käsitsi märkmete väärtusest: *The Pen Is Mightier Than the Keyboard: Advantages of Longhand Over Laptop Note Taking*. <https://journals.sagepub.com/doi/pdf/10.1177/0956797614524581>.

pausile ja muuta esitamise kiirust kiiremaks või ka aeglasemaks, mida traditsioonilises kooliõppes teha ei ole võimalik. Lisaks sellele võib olla käsitsi visuaalide loomine ajakulukas, eriti olukorras, kus edasi antavad kontseptsioonid on väga keerulised või vajavad dünaamilisust, mida tõdeb ka Ostling [9].

Eestikeelseid matemaatika videosid on juba varasemalt loodud. Suuremate vaatamiste arvuga, võib näiteks tuua YouTube'i kanalid KÕIK SELGEKS!⁷, MataMentor⁸, Mõtlematika⁹ ja Allar Veelmaa¹⁰. Uurides nende kanalite loodud sisu, võib märgata, et pea kõik videod on üles ehitatud mingi kindla ülesande lahendamise peale ning enamasti kasutatakse n.ö virtuaalset tahvlit, kuhu ülesannet lahendades kirjutatakse ja joonistatakse. Videosid põhikooli teemade kohta võime leida kõikidelt eelmainitud kanalitelt. See annab toetust väitele, et vabavaraalistel ja visuaalseid elemente kasutavatel õppevideotel on väärtust.

⁷ YouTube'i kanal KÕIK SELGEKS! <https://www.youtube.com/@k6ikselgeks>.

⁸ YouTube'i kanal MataMentor <https://www.youtube.com/@MataMentor>.

⁹ YouTube'i kanal Mõtlematika <https://www.youtube.com/@motlemaatika>.

¹⁰ YouTube'i kanal Allar Veelmaa <https://www.youtube.com/@allary>.

2. Eesmärkide ja nõuete kaardistamine

Käesolevas peatükis seatakse lõputöö eesmärgid, kaardistatakse nõuded animatsioonidele ning antakse ülevaade sellest, kuidas valminud videod ja töö käigus kirjutatud kood avalikkusele kättesaadavaks tehakse.

2.1 Lõputöö eesmärgid

Lõputöö üldine eesmärk on luua matemaatika õppematerjale põhikooli õppeastmele, mis annaksid vajalikku informatsiooni edasi kasutades animatsiooni, et toetada materjalide omandamist.

Loodud õppematerjalid peavad olema lihtsasti kättesaadavad, et neid oleks võimalik kasutada nii iseseisvaks õppeks kui ka igapäevases õppetöös. Sellest tulenevalt peaksid videod olema lihtsasti arusaadavad, et õpilane saaks neid iseseisvalt juurde õppimiseks kasutada, ilma välist abi kaasamata.

Lõputöö eesmärk ei ole asendada juba olemasolevaid õppematerjale, vaid luua materjale, mis toetaksid juba kasutatavaid vahendeid. Kuna YouTube'is on olemas õppematerjale, mida ka mainisime peatükis 1.3, mis abistavad kindlate ülesannete lahendamisel, peaksid lõputöö käigus loodud videod keskenduma pigem definitsioonide lahti seletamisele ja kasutama visuaalseid abivahendeid, et seda toetada. Eelnevatele eesmärkidele tuginedes püstitati järgmised tingimused lõputöö käigus valmivatele videotele:

1. videod peavad sisaldama ainult seda infot, mis on vajalik käsitletava teema edasi andmiseks;
2. videod peavad olema piisava pikkuse ja tempoga, et püüda ja hoida vaataja tähelepanu;
3. videod peavad olema lihtsasti kättesaadavad;
4. videode temaatika peab olema kooskõlas põhikoolis õpitavate temadega;
5. videotest peab keskenduma definitsioonide lahti seletamisele;
6. videod peavad sisaldama lisaks visuaalsele osale ka peale loetud teksti, et anda visuaalidele selgitust.

Videote kättesaadavuse, loendi 3. punkti, tagamiseks otsustati valminud materjalid laadida YouTube'i platvormile Creative Commons Attribution-NonCommercial (CC BY-NC) litsentsi all, mis annab loa videot muuta, adapteerida ja levitada koos viitamisega ainult mitteärielistel eesmärkidel [13].

Lisaks sellele otsustati videote käigus kirjutatud kood üles laadida avaliku koodihoidlasse, platvormil Github, et võimaldada edasiarendust. Ühtlasi lisati koodi kommentaare sinna, kus need vajalikud olid selguse loomiseks ning hoiti kood loetavana, kasutades võimalikult täpseid muutujate nimesid. Samuti tehti otsus hoida animatsioonid ühe video raames ainult *construct* meetodis. See tuleneb sellest, et koodi liigendamine erinevatesse meetodisse ei oleks oluliselt tõstnud koodi taaskasutatavuse taset ning oleks vähendanud koodi loetavust.

Videote temaatika valikud, loendi 4. punkt, tehakse põhikoolis käsitletavate teemade seast¹¹, et tagada videote kooskõla juba olemasolevate materjalidega. Temaatikat valides lähtuti sellest, et millised teemad saaksid enim kasu visuaalsetest elementidest.

2.2 Nõuded loodud animatsioonidele

Lõputöö eesmärkidest tulenevalt on vaja välja selgitada, milline on optimaalne video pikkus, et hoida vaataja tähelepanu, ning milline peaks olema peale loetud teksti tempo, et maksimaalselt toetada materjalide väärtust õppevahendina.

Varasemalt läbiviidud uuringutes on leitud, et video pikkus mängib suurt rolli. Näiteks Guo jt. 2014. aastal läbiviidud empiirilises uurimuses, mis hõlmas 6,7 miljonit õppevideo vaatamise sessiooni, leiti, et optimaalne video pikkus on kuus minutit või alla selle [14]. Lisaks leiti, et kiirem peale lugemise tempo hoidis üldiselt õpilase tähelepanu paremini, kuid pandi ka tähele, et kiirem tempo ainuüksi ei anna paremat tulemust, vaid suurt rolli mängib ka pealelugeja entusiasm [14]. Sellele annab toetust Kolthofi läbi viidud uuring [15], kus leiti et enamus uuringus osalenud 144st Hollandi keskkooli õpilasest eelistas videosid, mille pikkus oli umbes kuus minutit. Lisaks sellele eelistas 66,7% õpilastest informaalset instruktori kõneviisi ning

¹¹ Põhikoolis käsitletud teemad. TaskuTark. <https://www.taskutark.ee/sisukord-5/>.

59,5% õpilastest entusiastlikku kõneviisi. Nendele leidudele tuginedes on võimalik täpsustada eelnevas peatükis sätestatud tingimusi:

- videod ei tohi olla pikemad kui kuus minutit;
- teksti pealelugemisel eelistada informaalset ja entusiastlikku kõneviisi.

Olles paika seadnud tingimused, uuritakse võimalike tehnoloogilisi valikuid töö praktilise osa teostamiseks, ning tuuakse välja, milliseid nendest videote tegemisel kasutusele võetakse.

3. Kasutatud tehnoloogiad ja tööprotsess

Selles peatükis tuuakse välja töö käigus kasutatavad tehnoloogiad ning antakse ülevaade sellest, kuidas ideest saab video. Lisaks sellele kirjeldatakse Manim teegi loomislugu ja seda kuidas Manim teek renderdab animatsioone.

3.1 Manim

Animatsioonide loomiseks läbi programmeerimise on olemas mitmeid raamistikke, näiteks Three.js¹², Blender¹³, GeoGebra¹⁴ ja Manim, millel on oma tugevused ja nõrkused.

Esimene neist, Three.js, on JavaScripti-põhine raamistik ja sobib hästi interaktiivsete veebirakenduste loomiseks, kuid sellega animatsioonide loomine, vastavalt lõputöös seatud eesmärkidele, nõuaks palju rohkem arendustööd kui Manim teegis. Näiteks LaTeX teksti ekraanile loomine on Three.js raamistikus oluliselt keerulisem kui Manim teegis. Lisaks sellele ei paku Three.js eeldefineeritud animatsioone, seega iga animatsioon peaks olema käsitsi kirjutatud.

Blender pakub kõrget visuaalset kvaliteeti, kuid selle keerukas kasutajaliides ja üldine fookus 3D-graafikale muudavad selle vähem sobivaks vahendiks matemaatilise sisu loomiseks.

GeoGebra on küll interaktiivne ja suunatud just matemaatikale, kuid animatsioonide loomise võimekuse poolest on GeoGebra piiratud, loodud jooniseid saab küll salvestada pildina, kuid videoformaati antud platvorm ei toeta. See välistab GeoGebra videote loomise vahendina.

Manim eristub eelnevatest just seetõttu, et on spetsiaalselt ehitatud matemaatiliste animatsioonide loomiseks. Manim võimaldab vektori põhise visualiseerimist, animatsioonide ajastamist ja objektide liigutamist ajas ja ruumis ning sellel on LaTeXi tugi [16]. Lisaks sellele on Manim teek ülesehitatud Pythoni programmeerimiskeelele, millega on töö autoril enim kogemust. Nendest asjaoludest lähtuvalt osutus Manim kõige sobivamaks valikuks nii oma funktsionaalsuse kui ka sihtotstarbe poolest.

¹² 3D visuaalide loomise JavaScripti teek Three.js. <https://threejs.org/>.

¹³ 3D loomise platvorm Blender. <https://www.blender.org/>.

¹⁴ Matemaatika visualiseerimise platvorm GeoGebra. <https://www.geogebra.org/>.

3.1.1 Manimi loomisest

Manim on loodud Grant Sandersoni [16], tuntud kui YouTube'i sisulooja 3Blue1brown, poolt, kes kasutas seda isiklikes projektides, luues mitmesuguseid matemaatika, füüsika ja arvuti-teaduse suunaga videosid. Nende videote eesmärk on lihtsamal kujul edasi anda antud valdkondade keerulisi kontseptsioone ning seda animatsioonide abil¹⁵.

Sandersoni YouTube'i kanali¹⁶ kasvades muutus tema loodud animatsioonide stiil aina ligi tõmbavamaks teistele animatsioonihuvilistele. Kuna Sandersoni loodud animatsiooni mootor oli raskesti paigaldatav ja mitte väga kasutajasõbralik, hakkas ta 2019. aastal töötama uue Python programmeerimiskeeles kirjutatud OpenGL¹⁷ baasil Manim teegi kallal [16].

Aastal 2021 löid Manimi ümber tekkinud kogukonnast¹⁸ pärit arendajad ManimCE (Manim Community Edition) [16], mida ka selle lõputöö raames kasutatakse. ManimCE eesmärk on muuta Manim kättesaadavamaks ja kasutajasõbralikumaks, võrreldes algse Sandersoni poolt loodud Manim teegi versiooniga. Kogu ManimCE projekt on avatud lähtekoodiga¹⁹. Lisaks on kogukond kirjutanud dokumentatsiooni ja juhised Manimi paigaldamiseks²⁰. Lihtsuse mõttes viidatakse töös kasutatavale ManimCE teegile kui Manimile, sest ManimCE on enim kasutatud Manim teegi versioon ning üldjuhul viitataksegi sellele kui Manimile.

3.1.2 Animatsioonide loomine Manimi abil

Animatsioonide loomiseks on vaja luua Manimi projekt. Seda saab teha käsurealt, kasutades käsku *manim init project my-project --default*, kus *my-project* tähistab projekti nime.

¹⁵ Pärineb Grant Sandersoni YouTube-i kanali kirjeldusest <https://www.youtube.com/@3blue1brown>.

¹⁶ Grant Sandersoni YouTube-i kanal <https://www.youtube.com/@3blue1brown>.

¹⁷ Rakendusliides OpenGL <https://www.opengl.org/>.

¹⁸ ManimCE teegi kommuuni poolt seatud arenduseesmärgid <https://github.com/orgs/ManimCommunity/projects/7>.

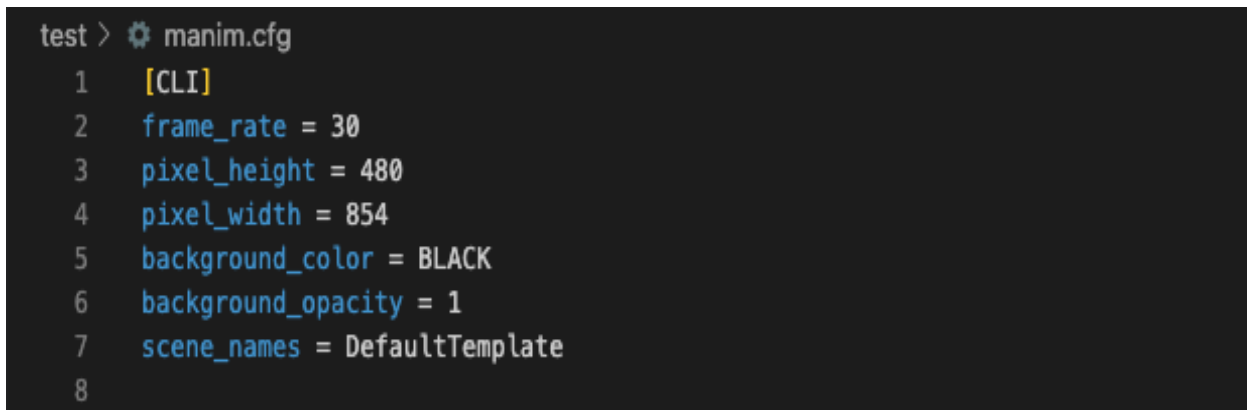
¹⁹ ManimCE projekti lähtekood <https://github.com/manimCommunity/manim>.

²⁰ Manim teegi dokumentatsioon <https://docs.manim.community/en/stable/index.html>.



Joonis 1. Manim projekti failipuu projekti loomise järgselt

Käsus defineeritud projekti nimi määrab ühtlasi ka Manimi poolt genereeritud faile sisaldava kausta nime. Joonisel 1 kannab see kaust nime *test*. Võtmesõna *--default* määrab, et projekt luuakse vaikeparameetritega. Käsu andmisel loodud kaust sisaldab kolme olulisemat osa: kaust *media*, Pythoni fail *main.py* ja projekti sätete fail *manim.cfg*, mida on samuti näha joonisel 1. Esimene neist, kaust *media*, sisaldab koodi jooksutamisel genereeritud videosid ja pilte. Fail *main.py* sisaldab koodi, mille abil meediat luuakse.



Joonis 2. Manim projekti konfiguratsiooni fail

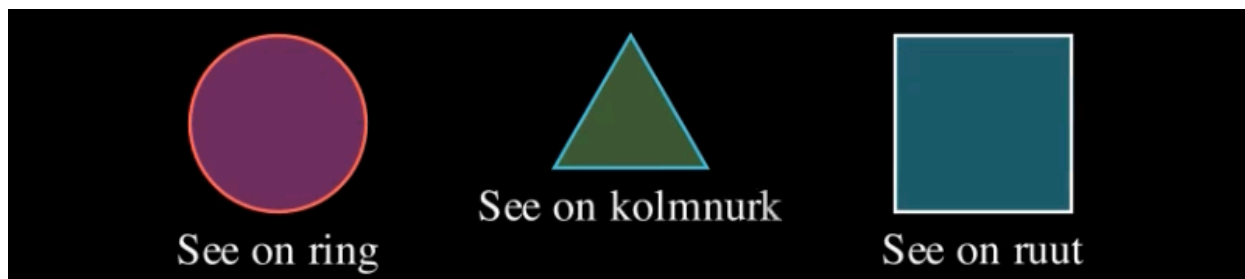
Kolmas oluline osa on sätete fail *manim.cfg*, kus saab muuta üldiseid Manimi parameetreid. Näiteks üksikpiltide arvu sekundis (ingl *frame rate*), genereeritud video või pildi mõõtmeid ja tausta värvi. Vaikeparameetritega loodud projekti sätete faili sisu on näha joonisel 2.

Kood on Manimis üles ehitatud stseenide põhimõttel. Iga stseen on klass, mis pärineb ülemklassist *Scene* ja kõik animatsioonid kirjutatakse selle klassi *construct* meetodis või sellest kutsuavate abifunktsioonide sees.

Manim pakub eeldefineeritud matemaatilisi objekte (Manim teegis tuntud kui *Mobject*), näiteks geomeetrilisi kujundeid *Circle*, *Square*, *Triangle*, kuid ka muid animatsioonide loomisel vajaminevaid objekte, näiteks *Text*, *MathTex* vastavalt tavalise teksti ja matemaatilistes valemities esineva teksti jaoks. Objekte saab grupeerida *VGroup* abil, et kõikidele grupis olevatele objektidele rakendada samu animatsioone/stiilimuutusi.

Manim kasutab Descartes'i koordinaatsüsteemi²¹ ning objekte on võimalik positsioneerida näiteks *move_to* funktsiooni abil või *next_to* abil mingi eelnevalt olemasoleva objekti suhtes. Lisaks pakub Manim erinevaid funktsioone objektide ekraanile kujutamiseks ja ekraanilt kustutamiseks. Näiteks *Create*, *Write* loomiseks ja *FadeOut* või *Unwrite* ekraanilt kaotamiseks.

Veel pakub Manim funktsioone ajastamiseks. Näiteks *self.wait(n)*, mis ootab n sekundit enne animatsiooniga edasi liikumist. Kogu info Manimi poolt pakutavate funktsioonide kohta võib leida dokumentatsioonist²².



Joonis 3. Üksikpilt Manim teegi abil loodud lihtsast animatsioonist, peale objektide loomist.

Kasutades ära eelmainitud Manim teegi pakutavaid funktsioone ja objekte, on võimalik luua animatsioone. Manim teegi poolt loodavate animatsioonide illustreerimiseks on välja toodud lihtne animatsioon, mille kuvatõmmist on näha joonisel 3 ning mille lähtekood on näha joonisel 4.

²¹ Descartes'i koordinaatsüsteem https://mathinsight.org/cartesian_coordinates.

²² Manim teegi dokumentatsioon <https://docs.manim.community/en/stable/index.html>.

Kood defineerib kolme geomeetrilise kujundi – kolmnurga ehk *triangle*, ringi ehk *circle* ja ruudu ehk *square* – ning igaühe kohta käiva selgitava teksti muutujad. Seejärel grupeeritakse objektid ja tekstid vastavalt ning joonistatakse need ekraanile kasutades *Create* ja *Write* funktsioone.

```
1  from manim import *
2  class DefaultTemplate(Scene):
3      def construct(self):
4
5          circle = Circle().move_to(LEFT * 4)
6          circle.set_fill(PINK, opacity=0.5)
7
8          circleText = Text('See on ring', font_size=36).next_to(circle, DOWN)
9
10         triangle = Triangle()
11         triangle.set_fill(GREEN, opacity=0.5)
12
13         triangleText = Text('See on kolmnurk', font_size=36).next_to(triangle, DOWN)
14
15         square = Square().move_to(RIGHT * 4)
16         square.set_fill(BLUE, opacity=0.5)
17
18         squareText = Text('See on ruut', font_size=36).next_to(square, DOWN)
19
20         objects = VGroup(circle, square, triangle)
21         texts = VGroup(circleText, squareText, triangleText)
22
23         self.play(Create(objects), Write(texts))
24
25         self.wait(2)
26
27         self.play(FadeOut(objects), Unwrite(texts))
```

Joonis 4. Manim teegi abil loodud lihtsa animatsiooni lähtekood

Peale kahe sekundi möödumist kaotatakse need ekraanilt kasutades *FadeOut* ja *Unwrite* funktsioone.

Objektide positsioneerimiseks on kasutatud *next_to* ja *move_to* käske. Funktsioon *next_to* vajab kahte argumenti: varasemalt defineeritud objekti, mille suhtes selle funktsiooni poolt mõjutatud objekti positsiooni muuta ning suunda, kuhu poole funktsioon positsioneeritava objekti paigutab. Võtmesõnad *LEFT* ja *DOWN* on Manim teegis eeldefineeritud väärtused, mis tähistavad ühte ühikut tasandil mingis kindlas suunas [16]. Teist positsioneerimise funktsiooni, *move_to*, on kasutatud, et paigutada objekte tasandi nullpunkti suhtes ette antud väärtuse võrra. Nullpunkt asub Manim teegi poolt kasutatud tasandil täpselt ekraani keskel [16].

3.1.3 Manimi sisemine töö

Järgnevas alapeatükis toodud informatsioon Manimi sisemisest tööst pärineb Manim teegi arendaja dokumentatsioonist [17].

Manimi struktuuris on neli olulisemat komponenti:

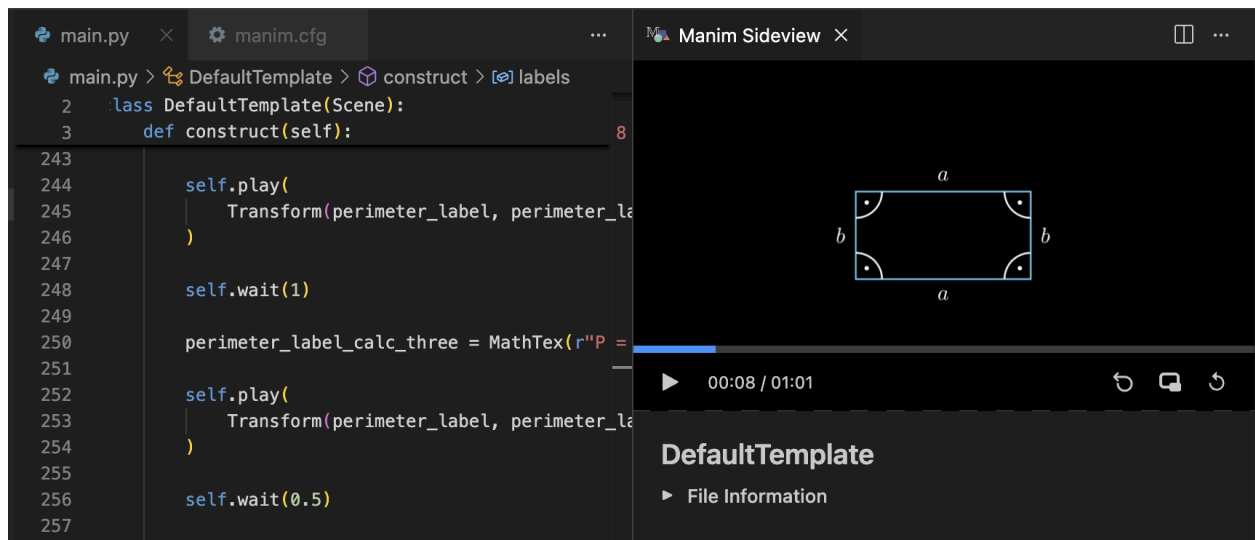
- objektid (ingl *Objects*) – eeldefineeritud objektid, millel on muudetavad atribuudid ning need tähistavad erinevaid matemaatilisi objekte või visuaalseid elemente. Näiteks *Text*, *Triangle*, *Polygon*, *Circle*;
- animatsioonid – funktsioonid, mida saab rakendada objektidele, et neid liigutada, muuta nende atribuute, luua neid ekraanile või neid sealt ära kaotada. Näiteks *Create*, *FadeOut*, *Rotate*, mis võtavad objekte argumentina ning *move_to*, *next_to*, mida rakendatakse objektidele;
- stseenid – klassid, mis hoiavad endas objekte ja nende animatsioone koos ajastuse informatsiooniga;
- varjutajad – igal objektil on määratud varjutaja failid, mis määravad ära objekti kujutamisi. Igal objektil peab olema vähemalt tipu varjutaja ja fragmendi varjutaja ning valikuliseks on geomeetria varjutaja.

Manim kasutab ModernGL²³ Pythoni teegi abil OpenGL baasil renderdamise graafikatoru. Igal stseni aja sammul kutsutakse *OpenGLRenderer* klassi *render* meetodit, mis omakorda kutsub *update_frame* meetodit ning kirjutab joonistatud üksikpildi genereeritavasse faili. Meetod *update_frame* käib tsükliliselt üle iga objekti, kutsudes nendel välja *render_object* meetodit, mis joonistab vastavalt objekti varjutajatele selle ekraanile. Eelmainitud *render_object* meetod vastutab andmete eeltöötlemise eest enne, kui andmed saadetakse ModernGLi graafikatorusse. Kui kõik renderdamise etapid on algoritmi poolt edukalt läbitud, siis töö lõpetatakse ning fail salvestatakse.

²³ ModernGL koodihoidla <https://github.com/moderngl/moderngl>.

3.2 VSCode liides Manim Sideview

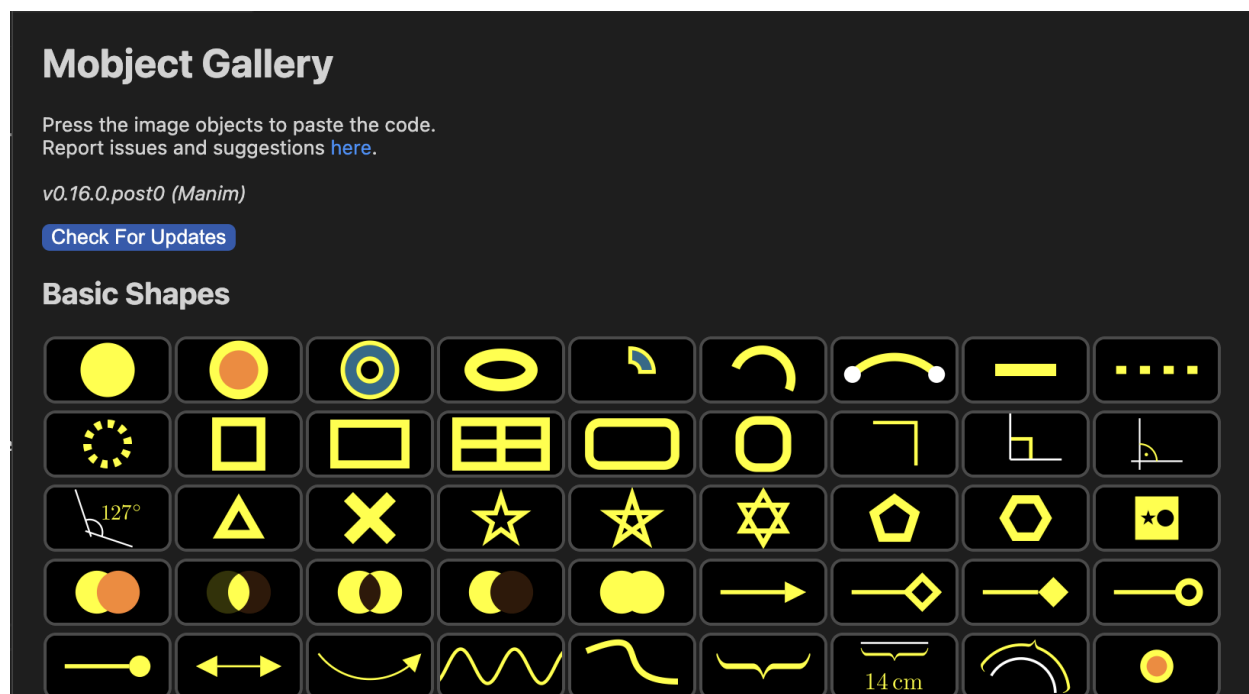
Manim teegi kasutamise teeb veelgi kasutajasõbralikumaks VSCode arenduskeskkonna rakendusliides Manim Sideview²⁴, mis võimaldab Manim teegi renderdatud animatsioonide kuvamist otse arenduskeskkonnas. Manim Sideview kasutamine teeb töövoogu mugavamaks, sest käsitsi käsurealt Manimi renderduse väljakutsumise asemel renderdab ja kuvab rakendusliides video uude arenduskeskkonna aknasse. Videot saab arenduskeskkonnas edasi kerida ja pausile panna, millest on arendusprotsessi jooksul palju abi. Manim Sideview rakendusliidese tööd on võimalik näha Joonisel 5.



Joonis 5. Manim Sideview rakendusliides töös.

Lisaks pakub Manim Sideview objektide galeriid, kust saab ühe nupuvajutusega koodi lisada eeldefineeritud lihtsamaid geomeetrilisi kujundeid ja visuaalseid elemente. Osa objektide galeriist on näha joonisel 6.

²⁴ Arenduskeskkonna VSCode rakendusliides Manim Sideview
<https://marketplace.visualstudio.com/items?itemName=Rickaym.manim-sideview>.



Joonis 6. Manim Sideview mobjektide galerii.

Töö käigus ei leidnud mobjektide galerii koodi kirjutamisel suurt kasutust ning pigem kasutati seda funktsionaalsust, et meelde tuletada mobjektide päiste definitsioone.

3.3 iMovie

Videote monteerimiseks otsustati kasutada Apple'i standardpaketi tarkvara iMovie't, mis on saadaval ainult macOS operatsioonisüsteemi kasutatavatel arvutitel ja iOS operatsioonisüsteemiga telefonidel. Peamiselt valiti just see rakenduse, sest tegu on vabavaraga ning alternatiivide soetamine on väga kulukas. Näiteks VEGAS Pro eest tuleb maksta 179 eurot ühekordselt [18] ja Adobe Premiere Pro puhul 26,64 eurot kuus [19]. Samuti on nimetatud tarkvarad küllaltki keerulise ülesehitusega ja seetõttu ka ajakulukamad õppida ja kasutada.

Kuna antud lõputöö jaoks vajamineva monteerimistarkvara funktsioonid piirdusid videote lõikamise, aeglustamise ja kiirendamise ning heli töötlemisega, siis ei olnud vajadust niivõrd detailse ja keeruka töötluse järele. See-eest iMovie on kasutajasõbralik ja efektiivne, võimaldades keskenduda videote sisulisele poolele, mitte erinevatele tehnilistele töötlusvõtetele.

Ühtlasi on antud keskkond MacOS'i süsteemidega hästi optimeeritud, mistõttu töötas see sujuvalt ning vähendas montaažiprotsessis esinevaid tõrkeid. Lisaks võimaldab iMovie hõlpsalt videoid erinevates formaatides salvestada ning kohandada nende kvaliteeti vastavalt sihtplatvormile, milleks antud juhul oli YouTube.

Tänu tarkvara kasutajasõbralikkuse ja helitöötuse võimalustele oli võimalik luua professionaalseid videoid, mistõttu oli iMovie sobiv ja tõhus valik töö eesmärkide saavutamiseks.

3.4 Ideest valmis videoni

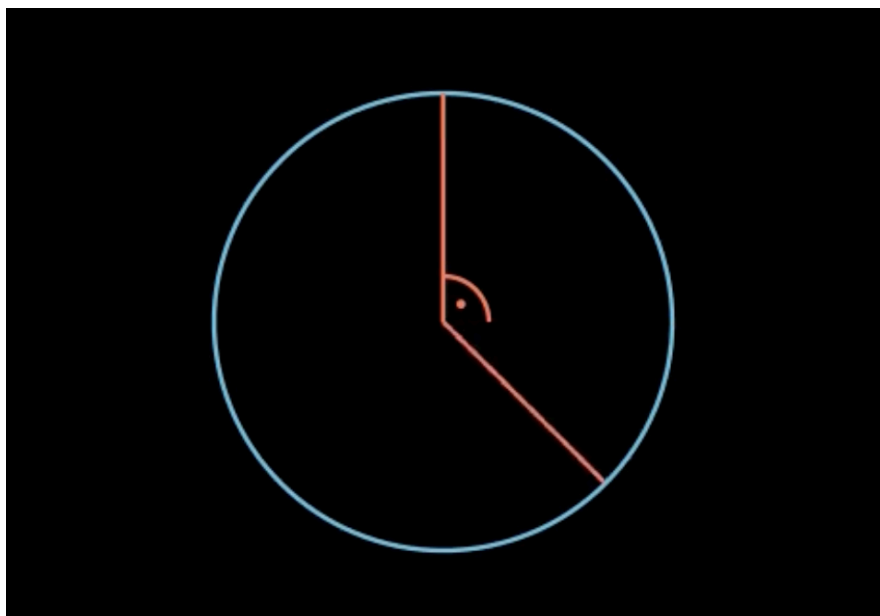
Videote tegemisel otsustati võtta esmaseks temaatikaks põhikooli geomeetria. Sellest tulenevalt sai esimese video temaks kolmnurk.

Kui teema oli valitud, siis esimese sammuna kaardistati ära ideed ja kontseptsioonid, mida videos katta taheti. Kaardistamise puhul jälgiti, et välja pakutud ideed kattuksid teemadega, mida õpetatakse põhikooli matemaatika õppekavas. Ideede kaardistamise samm on pigem eeltöö ning täpsemad teemad, mida katma peab, selgusid kogu protsessi vältel. Näiteks kolmnurga video puhul osutusid esmasteks teemadeks kolmnurga pindala ja übermõõdu valem ning reegel, et kolmnurga sisenurkade summa eukleidilises geomeetrias on 180° . Video loomise käigus aga selgus nendest ideedest tulenevalt, et peab katma ka üldisemat informatsiooni kolmnurga külgede ja nurkade kohta.

Sellele etapile järgnes koodi kirjutamine, mille eesmärk oli saada valmis prototüüpvideo, mille põhjal pealeloetav tekst kirjutada.

Koodi kirjutamisel olid suurimateks katsumusteks Manim teegi eripärad, mis tekitasid töövoogu ettenägematu olukordi. Näiteks, kui luua kahe lõigu vahele nurga objekt ning seejärel antud nurga haarasid pöörata, siis ei suurene nurka tähistava kaare pikkus nurga suurenemisega vastavuses. Kui nurka suurendada samas animatsioonis, siis oleks esmalt lõigud pöördunud ning seejärel alles suurenenud nurka tähistav kaar. See probleem lahendati nii, et loodi teised lõigud ja

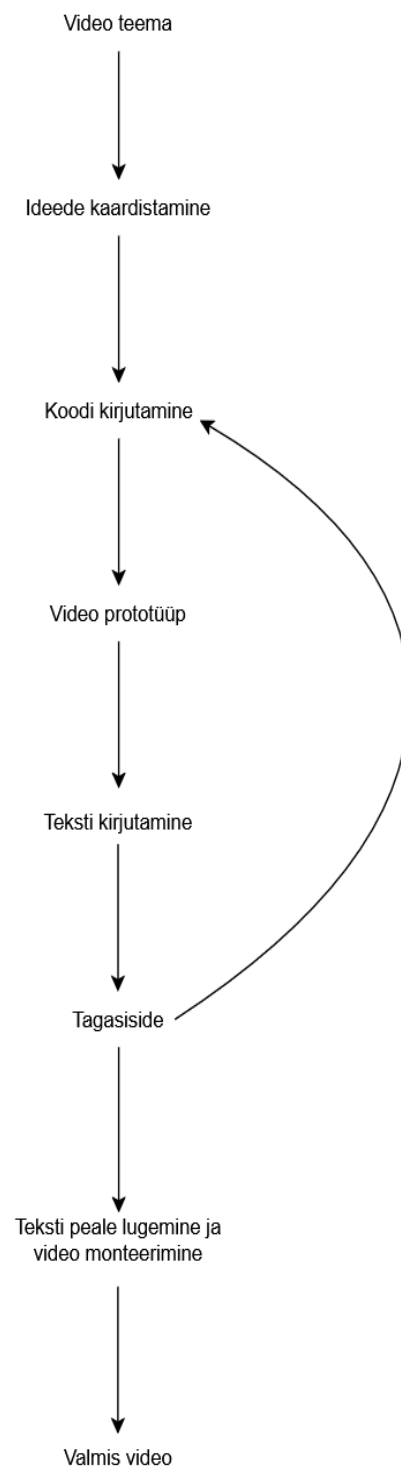
nurk nende vahele, mis asetsesid esimeste lõikude ja nurga lõpppositsioonis. Antud objektid olid nähtamatud ning hetkel, kui pöörati esimesi lõike, kasutati *Transform* funktsiooni, mis teisendas esimese nurga teiseks nurgaks. See lõi illusiooni, et nurk suureneb koos lõikude liikumisega. Kirjeldatud probleemi on näha jooniselt 7.



Joonis 7. Koodi kirjutamisel tekkinud Manim teegi eripära

Kui video prototüüp ja esmane tekst olid valmis, saadeti video juhendajale ja kaastudengitele tagasisidestamiseks. Saadud tagasiside põhjal tehti prototüüpi ja kirjutatud teksti vajalikud muudatused ning saadeti uuesti üle vaatamiseks.

Kui video ja tekst olid piisavalt kvaliteetsed, algas video monteerimine ja teksti pealelugemine, mida tehti samaaegselt. Pealelugemiseks kasutati iMovie sisseehitatud pealelugemise funktsiooni, mis võimaldab seda teha otse monteerimistarkvarast. Kuigi Manim teek pakub ajastamise funktsiooni, tehti lõplik ajastamine iMovie tarkvaras. Seda sellepärast, et koodi kirjutamise hetkel ei olnud teada, kui pikad pealeloetud teksti heliklipid on. Manim teegi pakutud



Joonis 8. Video valmimise protsess

ajastamist aga kasutati, et lisada animatsiooni pause, mis tegid monteerimise töövoogu mugavamaks.

Kõige lõpuks laeti valminud video üles YouTube'i ning muudeti video pealkiri, kirjeldus ja pisipilt vastavaks video temale. Joonis 8 on loodud illustreerimaks tööprotsessi etappe.

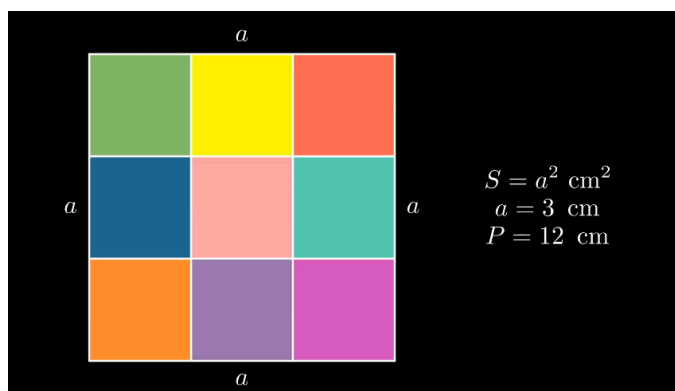
4. Tulemused

Selles peatükis kirjeldatakse valminud videosid, analüüsitakse tulemusi ning tuuakse välja võimalikud edasiarenduse võimalused.

4.1 Valminud videod

Lõputöö käigus valmis 4 videot: “Mis asi see kolmnurk on?”²⁵, “Mis asi see ruut on?”²⁶, “Mis asi see ring on?”²⁷ ja “Mis asi see ristkülik on?”²⁸. Bakalaureusetöö käigus loodud sisu pikkus kokku on 10 minutit ja 23 sekundit ning kõik videod on üles laetud töö autori YouTube’i kanalile²⁹.

Ruudu videos toodi välja ruudu definitsioon, diagonaali ja übermõõdu pikkuse valemid ja viimaseks pindala valem. Pindala valemi puhul toodi välja intuiitivne viis, kuidas mõelda pindala arvutamisest. Seda tehti läbi ruudu külje visualiseerimise väiksemate osadena, mida on näha joonisel 9. Lisaks sellele, toodi übermõõdu juures parema arusaama loomiseks välja idee, et kui laotada ruudu kõik küljed ühte ritta nii, et saadakse üks lõik, siis selle lõigu pikkus on võrdne ruudu übermõõduga. Selle idee visualiseerimist on näha joonisel 10. Lisaks toodi välja diagonaali pikkuse valemi tuletuskäik Pythagorase valemist.



Joonis 9. Pindala arvutamise visualiseerimine

²⁵ Valminud video “Mis asi see kolmnurk on?” <https://www.youtube.com/watch?v=uJfxr4mdaoA>.

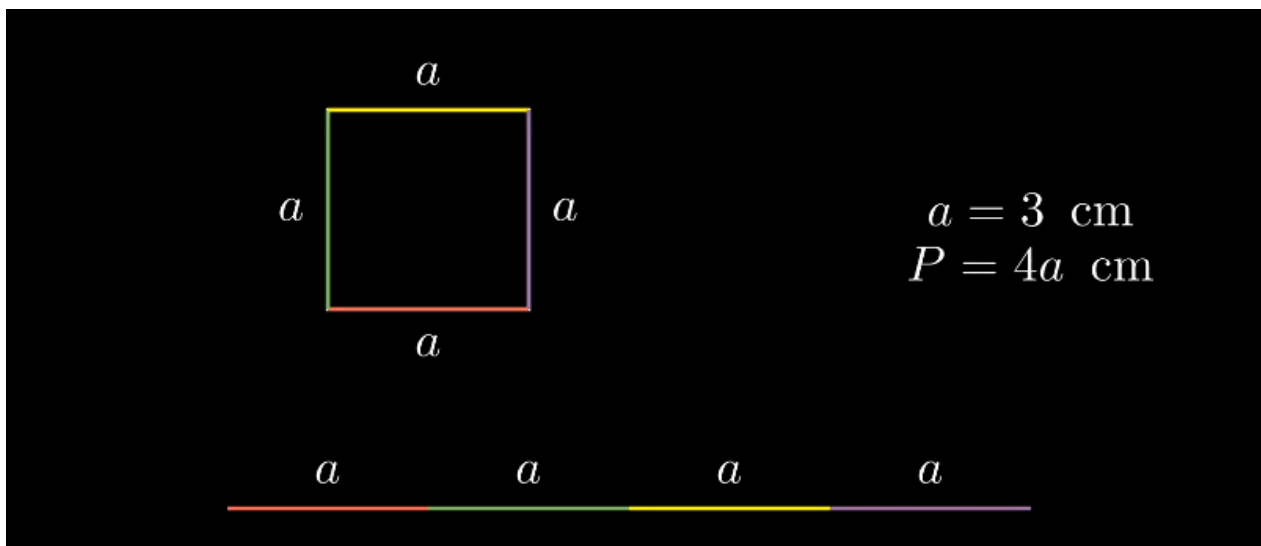
²⁶ Valminud video “Mis asi see ruut on?” <https://www.youtube.com/watch?v=tvxci001zgM>.

²⁷ Valminud video “Mis asi see ring on?” <https://www.youtube.com/watch?v=3b81bB1ln2Q>.

²⁸ Valminud video “Mis asi see ristkülik on?” https://www.youtube.com/watch?v=_adtHPSXHHk.

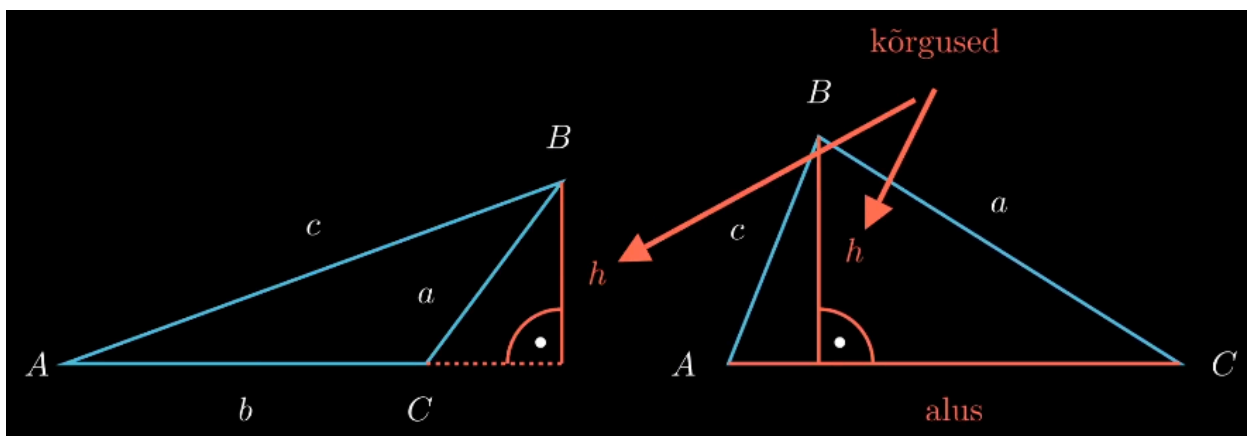
²⁹ YouTube’i kanal, kuhu bakalaureusetöö käigus valminud videod laeti

<https://www.youtube.com/@Informatemaatika>.



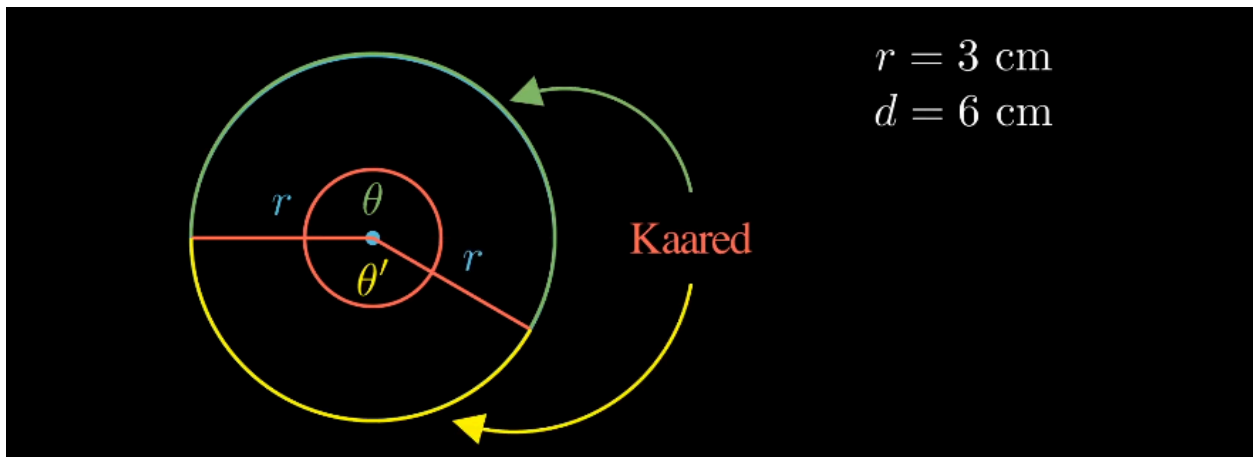
Joonis 10. Übermõõdu arutamise visualiseerimine.

Kolmnurga videos toodi samuti esile selle definitsioon, kolmnurga pindala valem ning fakt, et eukleidilises geomeetrias on kolmnurga sisenurkade summa 180° . Lisaks sellele toodi välja nurkade tähistamine ja kõrguse definitsioon. Kõrguse definitsiooni välja toomise juures loodud visuaali on näha joonisel 11.



Joonis 11. Kolmnurga kõrguse definitsiooni käsitlemisel loodud visuaal.

Ringi videos toodi välja nii ringi kui ka ringjoone definitsioon, ringi pindala ja übermõõdu valem ning sektori definitsioon ja pindala. Lisaks sellele selgitati, mis on ringi raadius ja diameeter ning tuuakse välja sektori kaare definitsioon. Sektorite ja nende kaarte välja toomist on näha joonisel 12.



Joonis 12. Ringi sektorid ning nende kaarte visualiseerimine

Bakalaureusetöö käigus loodud viimases, riskülikut käsitlevas, videos toodi välja risküliku definitsioon, pindala ja übermõõdu valem ning diagonaali valem. Sarnaselt ruudu videole loodi illustreeriv näide risküliku übermõõdu arvutamisest.

4.2 Tulemuste analüüs

Alustades videote tegemisega oli üks esimestest eesmärkidest teha selgeks, kui pikk ja kulukas videote tegemise protsess võib olla. Algselt loodeti valmis saada arvuliselt suurem hulk videosid, kuid esimese video valmimise käigus selgus, et videote tegemine võtab oodatust kauem aega.

Seda piiravat faktorit arvestades otsustati keskenduda loodud videote kvaliteedile, mitte kindlale videote arvule. Videote kvaliteet kindlustati eelneva uurimusega sellest, milliste tunnustega õppevideo hoiab kõige paremini õppijate tähelepanu ning millele peaks keskenduma, et luua maksimaalset väärtust õppematerjalina.

Videote tegemisel said bakalaureusetöö teises peatükis sätestatud tingimused täidetud. Kõik videote pikkused jäid alla kuue minuti, on lihtsasti kättesaadavad YouTube'i platvormil ja ei sisalda üleliigset infot, mida antud video teemade edasi andmisel vaja ei ole. Lisaks sellele kindlustati, et peale loetud tekst oleks korrektne ja kooskõlas põhikoolis käsitletavate matemaatika teemadega.

Kokkuvõttes valmis neli videot, mis kõik on YouTube'i platvormil kättesaadavad. Lisaks on videote lähtekood avalikult saadaval lõputöö Githubi koodihoidlas³⁰. Kokku kirjutati töö käigus 1697 rida koodi³¹, mis on kommenteeritud, et luua selgust ning jagada kood olulisemate animatsiooni faaside järgi.

Kõige iseenesest mõistetavamaks edasiarenduse võimaluseks on luua videoid teistest põhikoolis käsitletavatest matemaatika teemadest. Lisaks saaks luua videotele subtiitreid, et neid oleks võimalik kasutada õppimiseks ka ilma helita. Suureks kasuks nii uute kui ka juba loodud videote puhul oleks läbi viia fookusgrupid põhikoolis õppivate lastega, et paremini aru saada, millised videote aspektid sihtgrupi õpet kõige paremini toetavad.

Arutlus selle üle, kas loodud õppematerjalid on olemasolevatest lahendustest paremad, ei ole relevantne, kuna loodud õppematerjalid ei täida sama eesmärki, mis juba olemasolevad YouTube'i matemaatika õppematerjalid. Loodud õppematerjalides ei keskenduta kindlate ülesannete lahendamisele, vaid definitsioonide lahti seletamisele, kasutades visuaalseid elemente.

³⁰ Bakalaureusetöö Githubi koodihoidla <https://github.com/jsngnie/MathVideosWithManim>.

³¹ Number pärineb Githubi koodihoidla panustajate statistika vaatest.

5. Kokkuvõte

Töös loodi põhikooli matemaatika teemalised õppevideod kasutades Pythoni programmeerimiskeele Manim teeki. Eesmärgiks oli valmistada visuaalselt selged ja arusaadavad õppematerjalid, mis toetaksid olemasolevaid õppematerjale ning oleksid iseseisvaks õppimiseks kergesti kasutatavad.

Töö teoreetilises osas kirjeldati visuaalse õppe väärtust ning selle mõju õppijatele. Lisaks arutleti visuaalsete elementide rakendamise üle matemaatikas ning uuriti, millised eestikeelsed video vormis matemaatika õppematerjalid juba eksisteerivad. Tutvustati kasutatud tehnoloogiaid ning toodi välja, miks just sellised valikud tehti. Lisaks anti ülevaade praktilises osas kasutatavast Manim teegi loomeloost, selle pakutavatest funktsionaalsustest ning seletati lahti, kuidas Manim töötab et luua koodist animatsioone. Bakalaureusetöö käigus kasutatud tehnoloogiateks osutusid iMovie, Manim teek ja VSCode liides Manim Sideview. Selgitati tööprotsessi alates idee sõnastamisest kuni lõpliku video valmimiseni ja kirjeldati millised olid raskemad kohad selles. Viimaseks anti ülevaade bakalaureusetöö käigus loodud videotes käsitletud temadest ja toodi välja võimalikud edasiarenduse võimalused, milleks on subtiitrite loomine, muude põhikoolis käsitlevate teemade katmine uutes videotes ja fookusgruppide läbiviimine põhikooli õpilastega.

Bakalaureusetöö praktilise osa käigus valmis neli videot, mille kvaliteedi ja matemaatilisele korrektsuse kindlustamiseks rakendati eelnevalt uuritud heade õppematerjalide tavasid.

Valminud videote lähtekood on saadaval Githubi koodihoidlas³² ning videod ise töö autori YouTube'i kanalil³³. Töös toodi esile, et kuigi videote tootmine osutus ajamahukamaks, kui esialgu hinnatud, saavutati eesmärk luua kvaliteetsed ja õpetlikud videomaterjalid, mis on vastavuses põhikooli matemaatika õppekavaga.

³² Bakalaureusetöö Githubi koodihoidla <https://github.com/jsngnie/MathVideosWithManim>.

³³ YouTube'i kanal, kuhu bakalaureusetöö käigus valminud videod laeti <https://www.youtube.com/@Informatemaatika>.

Viidatud kirjandus

- [1] Pääjärvi S. Media use among 7–11-year-old children and their experiences on media education. *Finnish Society on Media Education*, 2011.
<https://mediakasvatus.fi/wp-content/uploads/2018/06/ISBN978-952-67693-2-5.pdf> (05.12.2024)
- [2] Vinter K. Digitaalse ekraanimeedia tarbimine 5-7-aastaste laste seas ja selle sotsiaalne vahendamine Eestis: Pedagoogiline vaatekoht. Tallinna Ülikooli kasvatusteaduste instituudi doktoritöö. 2013.
https://genire.ut.ee/wp-content/uploads/sites/128/k._vinter_doktoritoo_2013.pdf (27.02.2013).
- [3] Romanelli F., Bird E., Ryan M. Learning Styles: A review of theory, application, and best practices. *American Journal of Pharmaceutical Education*, 2009, vol. 73, no. 1, p. 9.
- [4] Cimene F. T. A., Mamburao M. L., Plaza Q. B., Nitcha H. Q., Somalipao M., Raña E. J. M., Baseo E. S., Siao Q. E. A., Mauna A. A., Cimene D. R. A. Generation Alpha Students' Behavior as Digital Natives and their Learning Engagement. *Psychology & Education: A Multidisciplinary Journal*, 2024, vol. 27, no. 3, pp. 258-273.
- [5] Mercadal T. Visual Learning. *Salem Press Encyclopedia*, 2021.
<https://research-ebSCO-com.ezproxy.utlib.ut.ee/c/qlurcm/viewer/html/fz5silj1zz> (05.12.2024).
- [6] Brown T., Zoghi M., Williams B., Jaberzadeh S., Roller L., Palermo C., McKenna L., Wright C., Baird M., Schneider-Kolsky M., Hewitt L., Sim J., Holt T.-A. Are learning style preferences of health science students predictive of their attitudes towards e-learning? *Australasian Journal of Educational Technology*, 2009, vol. 25, no. 4.
- [7] Bartan M. The use of storytelling methods by teachers and their effects on children's understanding and attention span. *Southeast Asia Early Childhood Journal*, 2020, vol. 9, no. 1, pp. 75–84.

- [8] Irani F. H., Afshar N. R., Parvizi, M., Barati, Z. The Impact of Blackboards or Whiteboard on language learning. *English Teacher*. 2020. no. 1, pp. 17-25.
- [9] Ostling M. Drawing on the Board. *Religious Studies and Theology*. 2019. no. 1 - 2. pp. 64-79
- [10] Yung H. I., Paas F. Effects of Computer-Based Visual Representation on Mathematics learning and cognitive load. *Journal of Educational Technology & Society*, 2015, vol. 18, no. 4, pp. 70–77.
- [11] Gambari A. I., Falode C. O., Adegbenro D. A. Effectiveness of computer animation and geometrical instructional model on mathematics achievement and retention among junior secondary school students. *European Journal of Science and Mathematics Education*, 2014, vol. 2, no. 2, pp. 127–146.
- [12] Schoenherr J., Strohmaier A. R., Stanislaw S., Learning with visualizations helps: A meta-analysis of visualization interventions in mathematics education. *Educational Research Review*. 2024. no. 45.
- [13] Creative Commons. <https://creativecommons.org/licenses/by-nc/4.0/> (10.05.2025).
- [14] Guo P. J., Kim J., Rubin R. How video production affects student engagement: an empirical study of MOOC videos. 2014. *L@S '14: Proceedings of the First ACM Conference on Learning*. Association for Computing Machinery (ACM). <https://doi.org/10.1145/2556325.2566239>
- [15] Kolthof A. A. Design guidelines for instructional videos in secondary mathematics Education: exploring student and teacher preferences. University of Twente Faculty of Behavioural, Management and Social Sciences Department of Instructional Technology masters thesis. 2021. https://essay.utwente.nl/86436/1/Kolthof_MA_BMS.pdf (23.06.2021)

[16] Manim. About Manim. <https://docs.manim.community/en/stable/faq/installation.html>
(10.04.2025)

[17] Github. Manim Developer Documentation.
[https://github.com/ManimCommunity/manim/wiki/Developer-documentation-\(WIP\)](https://github.com/ManimCommunity/manim/wiki/Developer-documentation-(WIP))
(20.04.2025)

[18] Vegas Creative Software. VEGAS Pro.
<https://www.vegascreativesoftware.com/us/vegas-pro/> (28.04.2025).

[19] Adobe. Premiere Pro. <https://www.adobe.com/ee/creativecloud/plans.html> (28.04.2025)

Lisad

Lisa 1. Litsents

Lihtlitsents lõputöö reprodutseerimiseks ja üldsusele kättesaadavaks tegemiseks

Mina, **Joosep Suuder**,

1. annan Tartu Ülikoolile tasuta loa (lihtlitsentsi) minu loodud teose
“**Manimi abil loodud õppevideod põhikooli matemaatikast**”,

mille juhendaja on Kati Ain, PhD ning kaasjuhendaja Toomas Krips, PhD

reprodutseerimiseks eesmärgiga seda säilitada, sealhulgas lisada digitaalarhiivi DSpace kuni autoriõiguse kehtivuse lõppemiseni.

2. Annan Tartu Ülikoolile loa teha punktis 1 nimetatud teos üldsusele kättesaadavaks Tartu Ülikooli veebikeskkonna, sealhulgas digitaalarhiivi DSpace kaudu Creative Commons'i litsentsiga CC BY NC ND 3.0, mis lubab autorile viidates teost reprodutseerida, levitada ja üldsusele suunata ning keelab luua tuletatud teost ja kasutada teost ärieesmärgil, kuni autoriõiguse kehtivuse lõppemiseni.
3. Olen teadlik, et punktides 1 ja 2 nimetatud õigused jäävad alles ka autorile.
4. Kinnitan, et lihtlitsentsi andmisega ei riku ma teiste isikute intellektuaalomandi ega isikuandmete kaitse õigusaktidest tulenevaid õigusi.

Joosep Suuder

10.05.2025