TARTU ÜLIKOOL
Arvutiteaduse Instituut
Informaatika õppekava

Rainer Kõiv

Tehisintellekti abiga joonistamine kasutades Stable Diffusion Img2Img mudelit

Bakalaureusetöö (9 EAP)

Juhendaja:
Ardi Tampuu, PhD

Tartu 2025

Tehisintellekti abiga joonistamine kasutades Stable Diffusion Img2Img mudelit

Lühikokkuvõte:

Bakalaureusetöö eesmärk oli luua interaktiivne demorakendus, mis võimaldab kasutajatel joonistada lihtsaid visandeid, mille alusel tehisintellekti mudelid tuvastavad objekte ja genereerivad neist valitud stiiliga täiustatud pildi. Demo on mõeldud kasutamiseks Delta õppehoone koridori väljapanekuna ning erinevatel üritustel tutvustamiseks. Töö koosneb teoreetilisest osast, kus tutvustatakse demo realiseerimiseks vajalikke tegevusi ja mudeleid, ning praktilisest osast, kus antakse ülevaade loodud lahendusest ning analüüsitakse tulemusi. Rakenduse kasutajaliides loodi Gradio raamistikus ning piltide töötlemiseks kasutati Florence-2-large, Stable Diffusion v1.5 ja SD ControlNet - Scribble mudeleid.

Võtmesõnad: Generatiivne tehisintellekt, lihtsad joonistused, demorakendus, Stable Diffusion, SD ControlNet, Florence, Gradio

CERCS: P175 Informaatika, süsteemiteooria, P176 Tehisintellekt

AI-Assisted Drawing Using the Stable Diffusion Img2Img Model

Abstract:

The goal of this bachelor's thesis was to develop an interactive demo application that allows users to draw simple sketches, from which AI models detect objects and generate enhanced images in a selected style. The demo is intended for display in the Delta academic building and for presentation at various events. The thesis consists of a theoretical part, introducing the components and models required to implement the demo, and a practical part, which provides an overview of the developed solution and analyzes the results. The application's user interface was built using the Gradio framework, and image processing was performed using the Florence-2-large, Stable Diffusion v1.5, and SD ControlNet – Scribble models.

Keywords: Generative AI, simple sketches, demo, Stable Diffusion, SD ControlNet, Florence, Gradio

CERCS: P175 Informatics, systems theory, P176 Artificial intelligence

Sisukord

Sissejuhatus

Alates 2022. aastast on tehisintellekt, eriti suured keelemudelid nagu ChatGPT, saanud väga palju tähelepanu. Lisaks keelelise info töötlemisele on kiiresti arenenud tehisintellekti võimekus töödelda ka teisi andmetüüpe. Näiteks on loodud pildigeneratsiooni mudelid, nagu DALL-E, Stable Diffusion ja Midjourney, mis suudavad lihtsate tekstikirjelduste ja olemasolevate piltide põhjal luua kvaliteetseid pilte. Kõnetuvastuse ja -sünteesi valdkonnas paistab silma OpenAI loodud Whisper[1], mis suudab kõnet töödelda mitmes keeles. Samuti on arendatud multimodaalseid mudelid, nagu GPT-4[2] ja Gemini[3], mis suudavad töödelda mitut sisendit korraga, genereerides näiteks viiba, heli ja pildi põhjal uue pildi (Sajid, 2024).

Selle bakalaureusetöö eesmärk on luua Delta koridori väljapanekuks interaktiivne süsteem, kus kasutajad saavad ekraanile joonistada lihtsaid visandeid või kritseldusi ning tehisintellekti mudel täiustab neid kunstiliste või fotorealistlike elementidega. Lisaks saavad kasutajad valida erinevaid stiile intuitiivse liidese kaudu. Töö tähtsus on näitamaks, et Tartu Ülikooli arvutiteaduse instituudis on võimalik teha huvitavaid projekte ning demos kasutatud tehnoloogia on aastal 2025 lihtsasti kättesaadav ja rakendatav. Valminud demo sihtrühmaks on noored, keda see tehnoloogia võiks inspireerida ja motiveerida instituuti õppima asuma. Lisaks võivad sihtrühma kuuluda instituudi vilistlased, praegused töötajad, tudengid ning külalised. Samuti on demo suunatud diplomaatilistele ja ärilistele külalistele, kellele on oluline näidata instituudi kaasaegsust ning kellele nähtud tehnoloogiad võivad pakkuda ideid koostööprojektideks.

Töö esimeses peatükis käsitletakse teoreetilise taustana difusioonimudeleid ning joonistuste tuvastamist ja töötlemist. Teises peatükis kirjeldatakse metoodikat ehk mida ja milleks kasutati ning kolmandas peatükis antakse hinnanguid ja analüüsitakse tagasisidet. Töö lisades on toodud demo töövoog, tehisintellekti kasutus, valminud programmi lähtekood, rohkem näiteid töödeldud joonistustest ja genereeritud piltidest ning küsitluse küsimused.

Käesoleva töö kirjutamisel kasutatud kohati tehisintellekti abi. Juba koostatud teksti töödeldi ja parandati juturoboti ChatGPT-4o abil. Kui juturobotit kasutati eraldi teksti loomiseks, siis on vastav osa välja toodud ja viidatud.

[1] Whisper: https://openai.com/index/whisper/
[2] GPT-4: https://openai.com/index/gpt-4-research/
[3] Gemini: https://www.coursera.org/articles/google-gemini-ai

1. Teoreetiline taust

Üks peamisi väljakutseid tehisnärvivõrkude täpsuse parandamisel on sisendite varieeruvus, mis raskendab mudelil sisendi korrektset tõlgendamist. Näiteks on trükitud teksti tuvastamisel (ingl *optical character recognition, OCR*) saavutatud väga kõrge täpsus isegi tasuta tööriistadega nagu Google Lens. Siiski jääb käsikirjalise teksti tuvastamine (ingl *handwritten text recognition, HTR*) keeruliseks ülesandeks, kuna käsikirjad varieeruvad inimese käekirja, stiili ja loetavuse poolest (Garrido-Munoz et al., 2025).

Sarnaselt käsikirjalise teksti mõistmisele on püstitatud ülesanne käsitsi joonistatud lihtsate piltide ehk kritselduste mõistmisest. See ülesanne võib osutuda keerulisemaks kui fotodelt objektide äratundmine, sest nii inimeste joonistusoskus kui ka mõte, kuidas mingit objekti visuaalsest kõige paremini kirjeldada, võib erineda. Seega on kritseldus huvitav ja keeruline sisendi tüüp, mis testib tehisintellekti võimet inimesi mõista. Demo loomine oli juhendaja poolt pakutud ülesanne töö autorile.

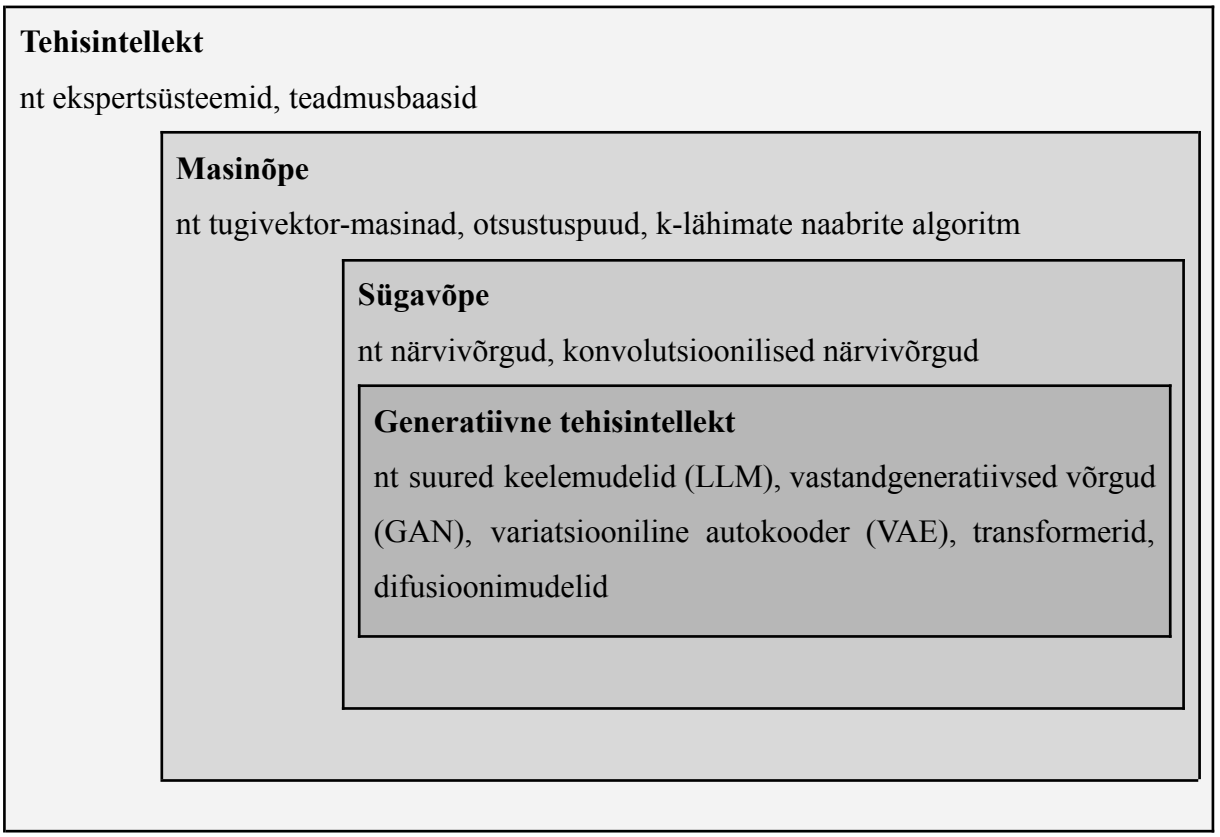
Käesolevas peatükis kirjeldatakse lähemalt generatiivset tehisintellekti ja tehisnärvivõrke, nende olemust ja tööpõhimõtet. Samuti saab ülevaate töös käsitletud stabiilse difusioonimudeli tööpõhimõttest ja selle jaoks loodud ControlNet arhitektuurist, mis võimaldas stabiilsel difusioonimudelil töödelda erinevaid sisendeid, antud juhul kasutaja abstraktset joonistust. Lisaks käsitletakse töös pildist teksti genereerivat mudelit ja juba olemasolevaid sarnaseid lahendusi.

1.1 Generatiivne tehisintellekt

Tehisintellekti mõiste võib tavainimese jaoks olla natukene hägune ja seda võidakse seostada ChatGPT juturobotiga. Seepärast tuleb esmalt täpsustada, mis üldse on generatiivne tehisintellekt ja kus see tehisintellekti valdkonnas täpsemalt asub. Generatiivse tehisintellekti kontseptsiooni on käsitletud erinevates hiljuti ilmunud artiklites (Banh & Strobel, 2023; Sengar et al., 2024). Banh ja Strobel (2023) on oma artiklis kirjeldanud, kuidas tehisintellekti alamkategooriad jagunevad ning järgnev lõik võtab selle kokku.

Tehisintellekti kasutatakse üldise mõistena, mis hõlmab erinevaid arvutuslikke algoritme. Need algoritmid suudavad tavaliselt lahendada ülesandeid, mis nõuavad inimlikku mõtlemist, näiteks naturaalse keele mõistmist, mustrite tuvastamist, valikute tegemist ja vigadest õppimist. Masinõppe alamkategooria tegeleb algoritmide loomisega, mis suudavad iseseisvalt andmetele tuginedes lahendada teatud ülesandeid ilma, et neid otseselt programmeeritakse.

Sügavõppe alamkategooria on masinõppe edasiarendus, mis tugineb närvivõrkudele, et töödelda keerukaid ja suuri andmehulki tänu mustrite tuvastamisele ning seoste tekitamisele. Närvivõrgud on mudelid, mis sarnanevad inimese aju struktuuriga ning koosnevad paljudest omavahel ühendatud kihtidest ja tehisneuronitest. Tänu närvivõrkudele on sügavõppega võimalik töödelda erinevaid andmeid ühemõõtmelistest signaalidest ja tekstist mitmemõõtmeliste piltide, videote ja helini. Sügavõppest on välja arenenud sügavad generatiivsed mudelid, mis suudavad genereerida uusi andmeid tuginedes olemasolevatele andmetele. Generatiivsete mudelite varasemad näited on Markovi peitmudel ja Bayesi võrgud. Närvivõrkude abiga on tänaseks välja arenenud generatiivne tehisintellekt. Joonisel 1 on toodud visuaalne kokkuvõte generatiivse tehisintellekti kategooria asukohast tehisintellekti valdkonnas.



Joonis 1. TI kontseptsioonid ja generatiivse TI asukoht selles valdkonnas (Banh & Strobel, 2023).

Sengar et al. (2024) on oma artiklis teinud põhjaliku ülevaate generatiivsest tehisintallektist (GenAI). Järgnev lõik on refereeritud nende artiklist.

GenAI on tehisintellekti alamkategooria, mis suudab genereerida uut sisu, kasutades selleks

generatiivseid mudeleid. Loodud sisu võib olla näiteks tekst, pilt või video. Need mudelid suudavad aru saada mustritest ja treeningandmete struktuurist, tänu millele on uus genereeritud sisu sarnane originaalse sisuga, aga mitte kunagi täpselt samasugune. GenAI võib jagada neljaks tuntumaks sisu genereerivaks viisiks: vastandgeneratiivsed võrgud (ingl *Generative Adversarial Networks, GAN*), transformeritel baseeruvad mudelid (ingl *Transformer-based Models, TRM*), variatsioonilised autokooderid (ingl *Variational Autoencoders, VAE)* ja difusioonimudelid (ingl *Diffusion Models, DM*).

Käesolevas töös keskendutakse peamiselt generatiivse tehisintellekti alamkategoorias difusioonimudelitele, mida kasutatakse joonistuse alusel pildi genereerimisel. Joonise sisu tuvastav või mõistev ja selle alusel teksti genereeriv mudel tugineb peamiselt transformeritele, mida tutuvustatakse lähemalt peatükis 1.4.

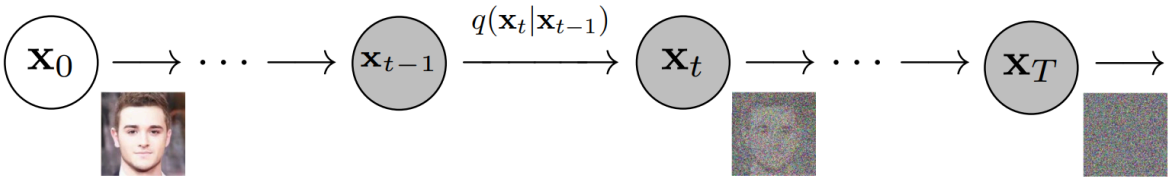
1.2 Difusioonimudelid

Nii nagu generatiivne tehisintellekt, on ka difusioonimudelid väga lai teema, mille kohta on ilmunud mitu põhjalikku teadusartiklit (Ho et al., 2020; Yang et al., 2024). Antud juhul on eesmärk anda ülevaade difusioonimudelite tööst pildi genereerimisel ja kuidas töös kasutatud stabiilne difusioonimudel nendest erineb. Järgnev lõik ja järgnevad kolm alalõiku on refereeritud ja põhinevad IBM artiklil (Bergmann & Stryker, 2024) ja seletaval videol (IBM Technology, 2025), kus kirjeldatakse difusioonimudeleid lähemalt. Matemaatilised valemid ja selgitus pärinevad Karagiannakos ja Adaloglou (2022) artiklist, mis kirjeldab difusioonimudelite matemaatilist tausta.

Difusioonimudelite idee põhineb vedelike diffusiooni füüsikal: kui veeklaasi tilgutada tilk värvi, siis värvi molekulid segunevad vee molekulidega ja kui nüüd tahta segust uuesti puhast vett saada, siis peab värvimolekulid mingil viisil veest eemaldama. Difusioonil põhinevaid närvivõrke treenitakse sügavõppe abil. Treeningu käigus muudavad närvivõrgud algse pildi iga piksli järk-järgult mürasemaks, kuni pilt koosneb täielikult mürast. Seejärel püüab mudel pöörddifusiooni abil müra eemaldada ja selle tulemusel vähem mürasemat pilti taastada. Selliselt treenitud närvivõrgu pöörddifusiooni haru suudab suvalisest mürast genereerida uusi andmepunkte, pilte, mis sarnanevad treenitud andmetega. Difusioonimudelitega piltide genereerimist võib jagada kolme etappi: pärisuunaline difusioon (ingl *forward diffusion*), tagasisuunaline difusioon (ingl *reverse diffusion*), tingimuslik difusioon (ingl *conditional diffusion*).

1.2.1 Pärisuunaline difusiooni etapp

Pärisuunalise difusiooni etapis lisatakse treeningpildile teatud ajahetkel ehk sammul Gaussi müra. Samme korratakse seni, kuni terve pilt koosneb mürast, ning seda protsessi kirjeldab joonis 2. Gaussi müra on normaaljaotusest saadud juhuslik väärtus, mis liidetakse igal sammul pildi iga piksli väärtusele. Müra lisamist saab kirjeldada Markovi ahelaga, milles järgmise oleku tõenäosus sõltub ainult hetkeolekust ja ei sõltu eelnenud muutuste jadast[4].



Joonis 2. Pärisuunaline difusiooni protsess, kus eelnevale pildile x_{t-1} lisatakse müra ning saadakse natukene mürasem pilt x_t. Protsess algab treeningpildist x_0 ja lõpeb mürast koosneva pildiga x_T (O'Connor, 2022).

Defineerime algse treeningpildi kui $x_{t=0}$, kus t on ajahetk ehk samm. Siis igal järgmisel sammul $t >= 1$ rakendatakse pildi x_{t-1} iga piksli igale värviväärtusele sõltumatult Gaussi müra ja saadakse natukene mürarohkem pilt x_t. Seda protsessi jätkates saadakse lõpuks täielikult mürast koosnev pilt x_T. Difusiooniprotsessi saab kokku võtta järgneva valemiga (O'Connor, 2022):

$$q(x_t|x_{t-1}) = N(x_t; \mu_t = \sqrt{1-\beta_t}x_{t-1}, \Sigma_t = \beta_t I), \tag{1}$$

kus x_t vastab mürasele pildile sammul t.

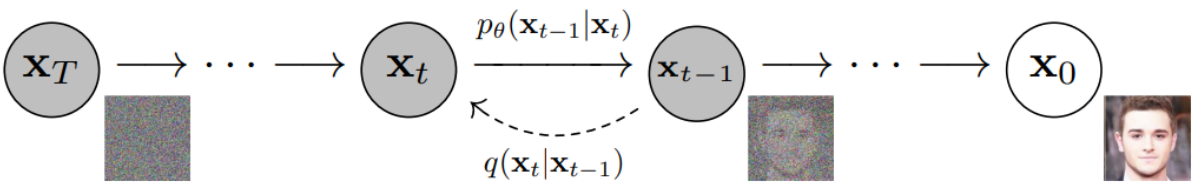
Iga samm on defineeritud tõenäosusfunktsiooniga $q(x_t|x_{t-1})$, mis tähendab, et hetkesed ennustatavad väärtused pildil x_t sõltuvad eelmistest väärtustest pildil x_{t-1}. See funktsioon on normaaljaotus, mis on defineeritud keskmise μ ja dispersiooni Σ kaudu. Valemis 1 on I ühikmaatriks, mis tähendab, et igale pikslile liidetakse keskmise ja standardhälbe põhjal iseseisvalt genereeritud juhuslik müra. Difusiooniprotsessi kiirust määrab dispersiooni planeerija (ingl *variance scheduler*), mis kontrollib normaaljaotuse keskmist ja

[4] Andmekaitse ja infoturbe portaal. Cybernetica AS. 2011 - 2023. https://akit.cyber.ee/

standardhälvet, kus kõrgem standardhälve tähendab, et suurema tõenäosusega valitakse suurem müra väärtus. Müra suurust määravat parameetrit tähistatakse β_t, kus β on alati positiivne väärtus sammul *t*.

1.2.2 Tagasisuunaline difusiooni etapp

Tagasisuunalise difusiooni etapis on difusioonimudeli eesmärk eelmise protsessi tulemina saadud vaid müra sisaldavast pildist järk-järgult müra eemaldada. Seda etappi kirjeldab joonis 3. Seda on võimalik teha treenides konvolutsioonilist U-Net tüüpi närvivõrku. Kui võtta mürane väljundpilt x_t suvalisel müra lisamise sammul t, siis mudel õpib, kuidas ennustada varasemal sammul sisendpildile x_{t-1} lisatud müra, mida eemaldada väljundpildilt x_t. Mudeli eesmärk on minimeerida keskmist ruutviga (ingl *mean squared error, MSE*) pöörddifusiooni abil ennustatud lisatud müra ja difusioonis tegelikult lisatud müra vahel. Seejärel saab mürasest pildist x_t maha lahutada ennustatud müra, mille tulemusel saadakse lähend vähem mürasele pildile x_{t-1}ühe võrra varasemal sammul $t - 1$.



Joonis 3. Tagasisuunaline difusiooniprotsess, kus närvivõrk ennustab pärisuunalises difusiooni etapis sisendpildile x_{t-1} lisatud müra väljundpildi x_t põhjal. Pildist x_t lahutatakse maha ennustatud müra ja saadakse vähem mürasem pilt x_{t-1} (O'Connor, 2022).

Selle protsessi jooksul õpib mudel, kuidas muuta pikslite mürased väärtused ühe sammu kaupa tagasi originaalseks väärtuseks. Sama arvutuslik mudel suudab ühe sammu võrra müra eemaldada nii täiesti müraselt pildilt kui ka peaaegu müravabalt pildilt. Kui selline müra eemaldamise funktsioon on hästi treenitud, siis neid samme palju korrates on tulemuseks loomuliku väljanägemisega pilt.

Juba treenitud pöörddifusiooni mudel suudab Gaussi mürast genereerida uusi pilte, mis sarnanevad originaaliga, aga pole täpselt samad. Matemaatiliselt võib seda protsessi kirjeldada valemiga (O'Connor, 2022):

$$p_\theta(x_{t-1}|x_t) = N(x_{t-1}; \mu_\theta(x_t, t), \Sigma_\theta(x_t, t)) \quad (2)$$

Valemis 2 ennustatakse $q(x_{t-1}|x_t)$ närvivõrguga p_θ, kus keskmine ja kovariatsioonimaatriks on parameetritega määratud.

1.2.3 Tingimuslik difusiooni etapp

Tingimusliku difusiooni puhul suunatakse mudeli pildi genereerimist tekstikirjelduse ehk viibaga. Juhusliku müra alusel uut pilti luues saadakse juhuslik pilt, mis sarnaneb treeningandmetega. Selleks, et genereerida viiba põhjal soovitud sisuga pilti, teisendatakse sisendtekst esmalt vektorruumi, mida mõistavad spetsiaalsed multimodaalsed mudelid, nagu OpenAI CLIP[5] või Microsofti Florence[6]. Need mudelid suudavad siduda keelelise tähenduse visuaalsete kujunditega ning toimivad sageli vahesammuna tekstist pilti genereerivate mudelite tööprotsessis. Tagasisuunalise difusioonimudeli treenimise käigus kasutatakse paariesitust pildist ja seda pilti või pildil olevat objekti kirjeldavast vektoresitusest. Mudel õpib käesoleva sammu juures mürase pildi ja teksti vektoresituse tunnuste põhjal, kui palju ja milliste mustritega müra eemaldada.

Tingimusliku difusiooni meetodeid saab jagada kaheks. Esiteks saab kasutada enesetähelepanu mehhanismi (ingl *Self-Attention Guidance, SAG*), mis aitab mudelil aru saada, kuidas mingid tekstiosad mõjutavad teatud pildiosa genereerimist. Teine võimalus on kasutada klassifitseerimisvaba juhendamist (ingl *Classifier-Free Guidance, CFG*), mis aitab suurendada teatud sõnade kaalu pildi genereerimisel. Kokkuvõttes õpib mudel sõnade tähenduse ja tagasisuunalise difusiooni sammude vahelisi suhteid, et mürasest pildist müra struktuuriliselt eemaldada ja uut pilti genereerida.

Enne difusioonimudeleid kasutati lihtsamaid ja vanemaid arhitektuure nagu vastandgeneratiivsed võrgud (GAN) ja autokooderid (VAE), kus tingimus koosneb objekti klassist. Difusioonimudelid on nendest võimsamad ja stabiilsemad. Tänapäeval on kõige tavalisem kasutusjuht, et tingimus seatakse keelemudeli abil kirjalikku kirjeldust kokku võttes vektorruumi vektoresitusse ning seda võetakse genereerimisel arvesse. Praegu on ühed

[5] CLIP: https://openai.com/index/clip/
[6] Florence: https://www.microsoft.com/en-us/research/project/projectflorence/

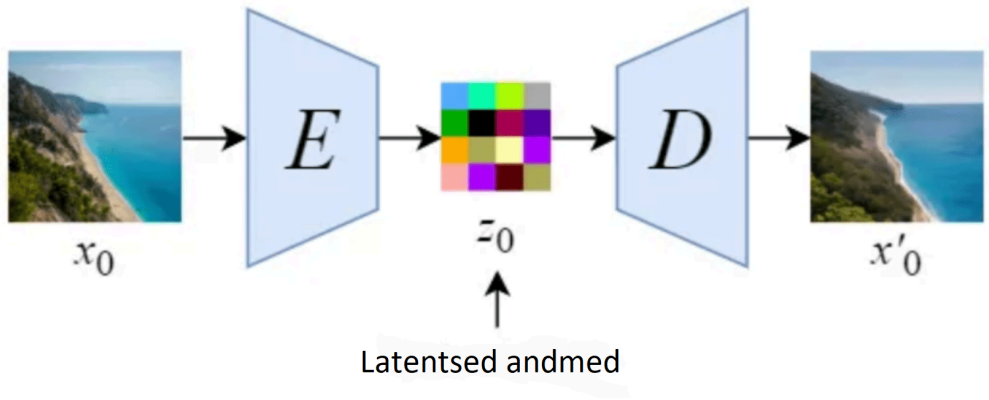
võimsamad difusioonimudelid Stability AI Stable Diffusion[7], OpenAI DALL-E[8] ja Midjourney[9].

Lisaks tekstist pildi genereerimise kasutusjuhule on difusioonimudelitel veel kasutusjuhte. Nendeks on pildist pilte genereerivad mudelid, puuduvate objektide pildile lisamine, heli ja video genereerimine. Käesolevas töös on soov lisaks tekstilisele genereerimise tingimusele seada ka ruumilise struktuuri tingimus ehk pilt.

1.2.4 Latentne difusioonimudel

Käesolevas töös kasutati latentset difusioonimudelit Stable Diffusion. Tavaliste difusioonimudelite puhul lastakse terve mürane pilt U-Net arhitektuurist läbi, mis on väga ajakulukas ja selle protsessi efektiivsemaks tegemise jaoks loodi Stable Diffusion, algselt tuntud kui Latent Diffusion Model (Aristimuño, 2023). Latentse difusioonimudeli kohta on ilmunud ka teadusartikkel (Rombach et al., 2022). Järgnevad lõigud kirjeldavad seda mudelit lähemalt ning põhinevad Ignacio Aristimuño (2023) artiklil ja Amazon Web Services artiklil (*What Is Stable Diffusion?*, s.a.).

Stable Diffusion erineb tavalisest difusioonimudelist selle poolest, et difusiooniprotsess toimub latentses ruumis (ingl *latent space*). See on võimalik tänu treenitud variatsioonilisele autokooderile (VAE), mis kodeerib täissuuruses pildi väiksemaks pildiks, mida pärisuunalises ja tagasisuunalises difusioonietapis töödelda ning pärast dekodeeritakse pilt täissuuruseks tagasi. Seda protsessi kirjeldab joonis 4.



Joonis 4. Kooder E muudab algse pildi väiksemaks, mida on efektiivsem töödelda. Dekooder D muudab töödeldud pildi täissuuruseks tagasi (*What Is Stable Diffusion?*, s.a.).

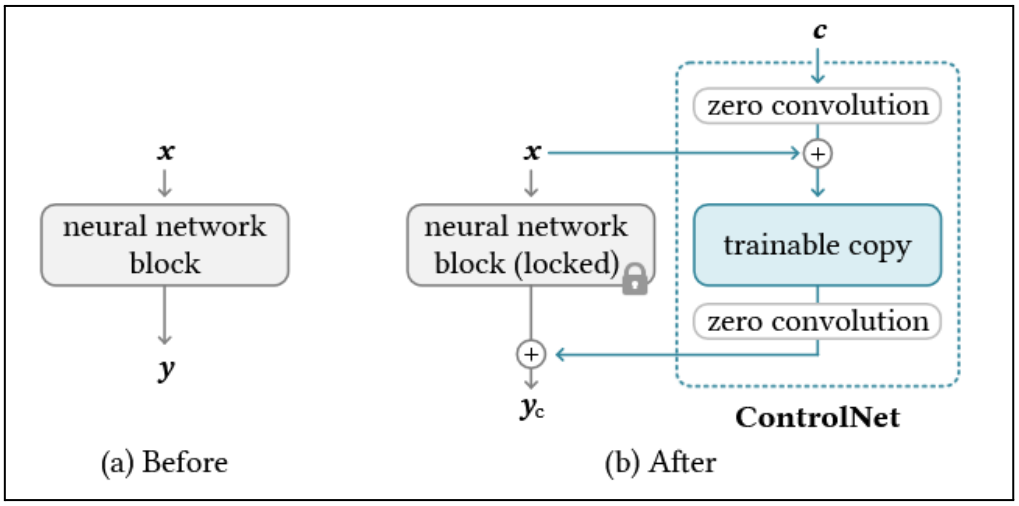
[7] Stable Diffusion: https://remaker.ai/stable-diffusion-img2img
[8] DALL-E: https://openai.com/index/dall-e-2/
[9] Midjourney: https://www.digitalartists.com/blog/midjourney-ai-art/

Näiteks muudetakse 512 × 512 pikslit pilt enne difusiooniprotsessi 64 × 64 pikslit pildiks, lisatakse ja eemaldatakse müra ning muudetakse tagasi 512 × 512 pikslit pildiks. Tänu kooderile on Stable Diffusion kiirem ja efektiivsem kui tavaline difusioonimudel.

1.3 Joonistuse töötlemine ControlNet mudeli abil

Käesolevas töös ainult teksti põhjal piltide genereerimisest ei piisa, kuna eesmärk on kasutaja joonistusi täiustada. Selleks tuleb kasutada pildist pilti genereerivat mudelit, mille tekstilise tingimuse asemel on pilt. Järgnev lõik on kokkuvõte Zhang et al. (2023) teadustekstist, kus nad lõid mitu erinevat mudelit, mis võimaldavad stabiilsel difusioonimudelil arvestada erinevat tüüpi sisendpiltidega. Antud töö puhul piirdutakse lihtsaid joonistusi (ingl *sketch* või *scribble*) töötleva mudeliga ControlNet - Scribble.

ControlNet on närvivõrgu arhitektuur, mis võimaldab lisada tingimuslikku kontrolli olemasolevatele suurtele tekstist pilti genereerivatele difusioonimudelitele, antud juhul Stable Diffusion mudelile. Selleks, et säilitada olemasoleva suure mudeli kvaliteet ja võimekus, hoitakse selle parameetrid muutumatuna (ehk "lukustatuna") ning luuakse selle kodeerimiskihtidest treenitav koopia. Selline arhitektuur kasutab suurt eeltreenitud mudelit kui alusbaasi, millele saab lisada uusi juhitavaid sisendeid. Treenitav koopia ja algne mudel on ühendatud null-konvolutsioonikihtide kaudu, mis on alguses nullkaaludega ning muutuvad treeningu käigus järk-järgult. Sellise lahenduse eesmärk on vältida kahjuliku müra lisandumist juba olemasolevatesse sügavamatesse kihtidesse treeningu algfaasis ning kaitsta suurt eeltreenitud mudelit soovimatute muutuste eest. Näide on toodud joonisel 5.



Joonis 5. Suur närvivõrk enne ja pärast ControlNet mudeliga ühendamist, kus x on sisend, y väljund ja c tingimuslik vektor (Zhang et al., 2023).

Nende katsetest selgub, et nende loodud ControlNet mudelid suudavad suunata stabiilset difusioonimudelit erinevate tingimuslike sisendite, kaasa arvatud kasutaja lihtsate joonistuste, kaudu. ControlNet mudeli treenimise käigus asendasid autorid pool teksti viibast tühja sõnega, et mudel õpiks sisendpildiga puuduvat teksti asendama. Samuti suudab mudel genereerida ilma viibata pilte, kuid selleks peab mudel tuvastama sisendpildilt semantilist sisu.

ControlNet mudel kasutab abistavat HED (ingl *Holistically-Nested Edge Detection*) mudelit, mis suudab närvivõrkude abil tuvastada pildilt või joonistuselt tähtsamaid ääri ja objekti piirjooni (Hutson, 2023). Tänu sellele on võimalik ControlNet mudelit lisaks joonistustele kasutada ka näiteks fotodest joonistuste tegemiseks ning joonistustest uuesti pildi genereerimiseks.

1.4 Joonistuse tuvastamine ja kirjelduse genereerimine

Selleks, et demo kasutamise protsess oleks kiirem ja kasutaja ei saaks vigu teha, tuli viiba loomine teha automaatselt, ilma et kasutaja midagi sisestama peaks. Fotodelt saab objekte tuvastada ja klassifitseerida erinevate masinõppe lahendustega, näiteks Tensorflow ja Keras, millega on võimalik ka ise mudeleid luua. Kuid lihtsate ja abstraktsete joonistuste pealt on esemete tuvastamine või mitme esemega stseeni kirjeldamine palju keerulisem. Selleks on mõistlik kasutada suuri eeltreenitud närvivõrke, mis pakuvad rohkem võimalusi. Käesolevas töös testiti erinevaid mudeleid, nagu BLIP 1, BLIP 2, CLIP ja Florence-2, täpsemalt Florence-2-large. Nendest mudelitest sai antud ülesandega kõige paremini hakkama Florence-2 mudel, mille kohta on avaldanud Bin Xiao jt teadusartikli (Xiao et al., 2024). Järgnevad kaks lõiku on refereeritud ja kirjeldavad nende lahendust lähemalt.

Nende eesmärk oli luua universaalne mudel, mis suudaks lihtsate juhistega lahendada erinevaid tehisnägemisel põhinevaid ülesandeid. Selline mudel peab aru saama ruumilistest tasemetest (ingl *spatial hierarchy*) ja semantilisest granulaarsusest (ingl *semantic granularity*). Ruumiliste detailide puhul peab mudel mõistma nii pilditasemelist kontseptsiooni kui ka pikslitasemelisi detaile, näiteks asukoha määramine või segmenteerimine. Semantilise granulaarsuse puhul peab mudel suutma luua tekste kõrgetasemelistest pealdistest (ingl *caption*) kuni detailsete kirjeldusteni, näiteks mõnest võtmesõnast koosnev info või mitmelauseline kirjeldus pildi jaoks. Sellise multitegumtöö õppimine nõuab palju kvaliteetseid märgendatud andmeid, mille jaoks nad lõid lisaks FLD-5B andmestiku, mis koosneb 126 millionist pildist ja 5,4 miljardist märgendatud

andmepunktist.

Selle põhjal lõid nad alusmudeli *Florence*-2, mis on universaalne viibatehnikal põhinev arhitektuur erinevate tehisnägemise ja nägemis-keeleülesannete jaoks. Mudeli *Florence*-2 treenimiseks kasutasid nad jadast-jadasse arhitektuuri (ingl *sequence-to-sequence*). Mudel võtab juhisena sisendiks tekstiviiba ja suudab pildi põhjal sooritada erinevaid ülesandeid nagu objekti tuvastust, kirjelduse genereerimist, asukoha tuvastust või segmenteerimist. Mudel kodeerib sisendpildid vektoresituseks ja ühendab need teksti vektoresitustega. Seejärel töötleb neid transformeril põhinev multimodaalne kooder ja dekooder, et genereerida vastus.

Käesoleva töö algusfaasis oli idee kasutada Google'i poolt loodud mängu "Quick, Draw!" klasse, kuid see mäng ja andmestik on aastast 2016, mis on tehnoloogia poolest pigem vananenud. Tänaseks loodud nägemisel ja keelel põhinevad närvivõrgud suudavad lisaks klassifitseerimisele või objekti tuvastusele ka pealdist ja pikemat kirjeldust genereerida. Selline pikem, detailsem ja täpsem kirjeldus annab pildi genereerimisel paremaid tulemusi.

1.5 Sarnased lahendused

Kui internetist otsida tasuta generatiivse tehisintellekti kasutamise võimalusi, siis leiab palju lahendusi, mis genereerivad viibast või pildist ja viibast uue pildi. Samuti eksisteerivad erinevad lahendused, mis suudavad lihtsast must-valgest joonistusest ja teksti viibast uut pilti genereerida, näiteks ChatGPT juturobotis DALL-E 3[10], OpenArt lahendus[11], Canva lahendus[12], Leonardo AI lahendus[13], Stable Diffusion lahendus[14] jne.

Kuid veebilahendustega on mitu piirangut. Esiteks tihti peab kasutaja konto looma, et pilti genereerida. Teiseks võivad tasuta versioonis olla kasutuspiirangud ja kõik võimalused ei pruugi saadaval olla. Kolmandaks pakuvad need lahendused tavakasutaja jaoks palju parameetrite muutmise valikuid, mida kasutaja ei pruugi mõista ja nende muutmine võib kaua aega võtta. Samuti peab kasutaja alati viiba ise kirjutama. Neljandaks võib ühe pildi genereerimine aega võtta minut või kaks ja kui saadud tulemus pole sobiv ning kasutaja tahab midagi joonistusel muuta, siis peab ta jälle ootama. Käesolev töö üritab neid probleeme leevendada.

[10] DALL-E 3: https://openai.com/index/dall-e-3/
[11] OpenArt: https://openart.ai/apps/sketch-to-image
[12] Canva: https://www.canva.com/features/ai-sketch-and-draw/
[13] Leonardo AI: https://leonardo.ai/wiki/from-sketch-to-masterpiece/
[14] Stable Diffusion: https://remaker.ai/stable-diffusion-img2img

2. Metoodika

Käesolevas peatükis kirjeldatakse valminud programmi arhitektuuri ja komponente lähemalt. Komponendid on Gradio kasutajaliides, joonistusi pildiks muutev stabiilne difusioonimudel ja seda suunav kritselduste töötlemiseks mõeldud ControlNet mudel ning pildist kirjeldust genereeriv nägemis-keelemudel (ingl *visual language model, VLM*). Samuti antakse ülevaade, kuidas komponendid omavahel ühendati ja milliseid parameetrite väärtuseid tulemuste saamiseks kasutati ning milliseid riist- ja tarkvara võimalusi kasutati demo realiseerimiseks.

2.1 Töövoog ja arhitektuur

Käesolevas töös loodud demo koosneb kasutajaliidesest ja tagaliidesest. Rakendus kirjutati programmeerimiskeeles Python. Kasutajaliideses tehtud valikute põhjal saadetakse vastavad parameetrid tagaliidesesse, kus neid töödeldakse ning tulemused saadetakse kasutajaliidesesse. Töövoogu ja üldist arhitektuuri kirjeldab lisa 1 ning järgnev algoritmiline kirjeldus:

1. Kasutajaliides ja tagaliides avatakse.
2. Kasutaja joonistab pildi ja valib režiimi, stiili ning tausta.
3. Joonistusala andmed, pildi mõõtmed, valitud režiim, stiil ja taust edastatakse tagaliidesesse.
4. Kontrollitakse, kas kasutaja on joonistanud pildi:
    - Kui kasutaja on joonistanud, jätkatakse punktiga 5.
    - Kui pilti ei ole joonistatud, jätkatakse punktiga 7.
5. Joonistust töödeldakse, rakendades ControlNet ja HED mudelit, et tuvastada joonistuse piirjooned.
6. Sõltuvalt valitud režiimist:
    - **Objekti tuvastuse** korral tuvastatakse objekt Florence-2-large mudeli abil;
    - **Stseeni kirjelduse** korral genereeritakse lühike kirjeldav pealdis (*caption*) Florence-2-large mudeli abil;
    - **Suvalise stseeni** korral jäetakse kirjeldus tühjaks.
7. Moodustatakse lõplik viip:
    - Tuvasustatud objekti, valitud stiili ja tausta põhjal;
    - Või stseeni kirjelduse, valitud stiili ja tausta põhjal;
    - Või ainult stiili ja tausta põhjal.
8. Uue pildi genereerimiseks kasutatakse Stable Diffusion v1.5 ja ControlNet mudeleid,

lähtudes viibast ja töödeldud joonistusest (kui olemas).

9. Salvestatakse töödeldud joonistus ja genereeritud pilt.
10. Genereeritud pilt ja kirjeldus väljastatakse kasutajaliidesesse.

2.2 Kasutajaliides

Masinõppe ja tehisintellekti mudeleid on tihti keeruline teistega jagada ja testida, kuna mudelid nõuavad palju arvutusressurssi, mida pakuvad üksnes võimsad ning kallid graafikakaardid. Google Colab ja Jupyter Notebook seda mingil määral võimaldavad, kuid Google'i poolt pakutavatel ressurssidel on piirangud ning Notebook on mõeldud pigem koodi ja andmeteaduse tulemuste näitamiseks kui tavakasutajate jaoks. Selle probleemi lahendamiseks on loodud erinevaid kasutajaliideseid, mis võimaldavad veebibrauseris oma tehisintellektil põhinevat programmi demonstreerida, näiteks Gradio[15] ja Taipy[16].

Käesolevas töös kasutati kasutajaliidese loomisel Python paketti Gradio, kuna see on lihtsasti integreeritav tehisintellekti projektidega, põhineb Pythonil ning pakub eeldefineeritud komponente. Selle jaoks on loodud ka lõuendi komponent, mis oli antud töö puhul väga tähtis, kuna kasutajad saavad otse Gradio kasutajaliideses oma joonistusi luua. Lisaks võimaldab Gradio kasutada kohandatud CSS-i, et muuta liidese kujundust vastavalt vajadusele. Kasutajaliidesest suunatakse sisendid – joonis, režiim ehk tuvastamise viis, stiil, taust ja uue pildi mõõtmed – tagaliidesesse. Järgnev lõik on kokkuvõte Gradio loojate Abid et al. (2019) teadusartiklist, mis kirjeldab paketti lähemalt.

Gradio on avatud lähtekoodiga Python pakett, mis pakub masinõppe mudelite jaoks veebipõhist kasutajaliidest. Pakett sisaldab teeki, mis toetab levinumaid liideseid erinevate mudelite tööde jaoks, nagu piltidel, helil ja tekstil põhinevad mudelid. See teeb koostöö masinõppe teadlaste ja kindla valdkonna ekspertide vahel lihtsamaks, kuna valdkonna eksperdid ei pea omama informaatika alaseid teadmisi ja saavad kasutajaliidese kaudu mudelit testida ning mudelile kohe tagasisidet anda. Gradio töötab erinevate masinõppe raamistikega, näiteks Scikit-Learn, TensorFlow ja PyTorch mudelitega, ning võimaldab kasutajal genereerida jagamislinki, et mudeleid lihtsalt jagada. Teek pakub erinevaid lihtsaid võimalusi sisendeid muuta, näiteks pilte töödelda, teksti muuta, helile müra lisada ning videoid lühemaks lõigata. Samuti on võimalik kasutajaliidest kasutada Google Colab või Jupyter Notebook projektidega.

[15] Gradio veebileht: https://www.gradio.app/
[16] Taipy veebileht: https://taipy.io/

Kuigi Gradio eesmärk on pakkuda kasutajaliidest, kus mudel on kättesaadav veebi kaudu, siis käesoleva töö üheks tingimuseks oli, et programm töötaks ilma internetiühenduseta, mida Gradio ka vaikimisi võimaldab. Järgnevas lõigus kirjeldatakse käesolevas töös kasutatud Gradio võimalusi tuginedes Gradio loojate GitHub repositooriumi juhistele[17] ja dokumentatsioonile (*Gradio Documentation*, s.a.).

Üheks võimaluseks kasutajaliidese loomisel on kasutada kõrgtasemelist *Interface* klassi, kuid antud töös oli vaja mahutada mitu komponenti ekraanile nii, et kasutaja ei peaks brauseris üles-alla kerima. Selleks pakub Gradio madalatasemelist lähenemist *Blocks* klassiga, millega on võimalik komponente paigutada lehel soovitud asukohale. Klassidega *Row* ja *Column* on võimalik komponente paigutada üksteise alla või kõrvale. Käesolevas töös loodud kasutajaliides on jagatud kaheks: vasakul pool asuvad sisendid ja paremal pool väljundid.

Sisendite poolel kasutati töös järgnevaid komponente:

- *Sketchpad* klassi, et luua lõuend, millele kasutaja saab joonistada. Komponendi mõõtmed on 800 × 600 ning lõuendi mõõtmed on 768 × 512. Töös kasutatud Stable Diffusion v1.5 mudelit oli treenitud piltidega, mille resolutsiooniks oli 512 × 512 pikslit ning testimise käigus selgus, et genereeritud piltide kõrgus ei olnud suurem kui 512 pikslit, kuigi lõuendi kõrgus oli. Sellest tulenevalt jäi lõuendi ja genereeritud piltide kõrguseks 512 pikslit ning kuna ekraanil oli laiusesse ruumi üle, siis suurendati laius 768 pikslile, mis ka genereeritud piltidel kajastus;
- *Radio* klass kolme režiimi valikuga joonistuse tuvastamiseks. See lubab teha ainult ühe valiku korraga.
- Kaks *Dropdown* klassi genereeritava pildi stiili ja tausta valikute menüüde jaoks;
- Sisendite all on nupu komponent, mis annab lõuendi, režiimi, stiili ja tausta parameetritena vastavale tagaliidese funktsioonile.

Väljundite ekraanipoolel asuvad järgnevad komponendid:

- *Image* klassi objekt, mis tagastab kasutajale tehisintellekti poolt täiendatud pildi (peamine väljund);
- *Textbox* klassi objekt, mis tagastab kasutajale info tema joonistuse alusel automaatselt loodud stseenikirjelduse (toetav, seletav info);
- Ruumi kokkuhoiu huvides lisati autori ja juhendaja kontaktandmed nende komponentide alla.

[17] Täpsemad juhised: https://github.com/gradio-app/gradio

Kasutajaliidese teksti suurust ja kujundust muudeti kohandatud CSS-faili abil.

2.3 Mudelid

Tagaliideses on vaja kasutaja joonistus muuta generatiivse tehisintellekti abiga vastava stiiliga pildiks. Pilte töötlevaid tehisintellekti mudeleid on nüüdseks juba palju ja need on integreeritud ka juturobotitesse, näiteks ChatGPT puhul töötleb pilte DALL-E. Antud töö ühe tingimusena pidi demo töötama kohalikus arvutis, seega oli vaja leida mudel, mida on võimalik alla laadida ja kasutada ka ilma internetita. Juhendaja pakutud teema kirjelduses oli näide kasutada stabiilse difusiooni pildist pilti genereerivat mudelit ning autor otsustas kasutada Stability AI stabiilseid difusioonimudeleid, sest need on enim kasutatavad ja hästi dokumenteeritud. Kuna paljud pildist pilti genereerivad mudelid vajavad viipa, siis oli vaja leida ka mudel, mis kirjeldaks ja genereeriks kasutaja joonistuse põhjal viiba.

2.3.1 Joonistuste töötlemine baasmudelit suunava ControlNet mudeliga

Esiteks oli vaja leida mudel, mis suudab lihtsaid kasutaja joonistusi töödelda. Paljud mudelid, mis genereerivad teksti ja pildi või lihtsalt teksti põhjal uue pildi, ei suuda arvestada abstraktsete must-valgete piltidega.

Käesolevas töös implementeeriti ControlNet - Scribble mudel toetudes loojate juhendile (Zhang & Agrawala, 2023). Töös kasutatud eeltreenitud ControlNet mudel on “lllyasviel/sd-controlnet-scribble” ja seda abistav meetod, mis töötleb kasutaja joonistust, on HED (ingl *Holistically-Nested Edge Detection*). Soovitatavalt tuleb mudelit kasutada koos üldisema pildist pilti genereeriva Stable Diffusion v1-5 mudeliga, millest lähemalt järgmises peatükis. Mudeli kasutamiseks tuleb esmalt paigaldada joonist töötlev abistav mudel (*controlnet_aux*) ning vajalikud paketid (*diffusers, transformers, accelerate*). Seejärel tuleb vajalikud moodulid importida ja mudelid alla laadida.

Töö käigus testiti ka teist ControlNet Scribble SDXL mudelit[18], mis töötab Stable Diffusion XL baasmudeliga. Kuid nende mudelitega võttis ühe pildi genereerimine aega üle minuti, mistõttu need antud kasutusjuhuks ei sobinud.

[18] ControlNet Scribble SDXL HuggingFace repositoorium: https://huggingface.co/xinsir/controlnet-scribble-sdxl-1.0

2.3.2 Piltide genereerimine eeltreenitud latentse difusioonimudeliga

Kuna ControlNet mudel töötab kõige paremini Stable Diffusion v1.5 baasmudeliga, siis seda töös ka kasutati. Töö käigus testiti teisi mudeleid, nagu Stable Diffusion XL, aga nendega oli piltide genereerimine umbes kümme korda aeglasem ja hinnanguliselt kvaliteet väga palju ei paranenud. Stable Diffusion v1.5 implementeeriti loojate kasutusjuhendi (Rombach et al., 2024) järgi, mis on koos täiendava infoga (*Image-to-image*, s.a.) saadaval Hugging Face repositooriumist. Järgnevad lõigud kirjeldavad kasutatud Stable Diffusion v1.5 mudelit tuginedes juhendile.

Töös kasutati koos ControlNet mudeliga RunwayML repositooriumi mudelit "runwayml/stable-diffusion-v1-5". Toru (ingl *pipeline*) abil kombineeritakse Stable Diffusion ja ControlNet mudelid ning toru peamised sisendid on kasutaja joonistus ja viip. Viip pannakse kokku eeldefineeritud stiilist ja taustast ning automaatselt genereeritud objektist või pildi kirjeldusest. Eeldefineeritud stiile ja taustu kirjeldatakse lähemalt peatükis 2.6. Lisaks ka genereeritava pildi mõõtmed, et need oleksid kasutaja joonistuse mõõtmetega samad. Töö käigus seati toru osadele lisaks mitu parameetrit:

- Tugevuse (ingl *strength*) parameeter väärtusega 0,35 vahemikust 0,0–1,0, mis määrab, kui palju sarnaneb genereeritud pilt mudelile etteantud pildiga. Mida väiksem väärtus, seda sarnasemad pildid on ja mida suurem väärtus, seda erinevamad pildid on. Antud juhul ei tohtinud väärtus väga suur olla, muidu ei võetud kasutaja joonistust üldse arvesse;
- Sammude arvu (ingl *number of inference steps*) parameeter väärtusega 30. See parameeter on seotud tugevuse parameetriga, mis määrab müra lisavate sammude arvu. Antud juhul 30 × 0,35 ehk 10,5 sammu müra lisamiseks originaalsele pildile ja 10,5 sammu mürast pildi genereerimiseks. Selle väärtuse suurendamisel genereerib mudel kvaliteetsemaid pilte, aga on seetõttu aeglasem, seega pidi autor leidma tasakaalu;
- Juhendamise kaalu (ingl *guidance scale*) parameeter väärtusega 7,5. Kõrge väärtus tähendab, et viiba osakaal on pildi genereerimisel suur ning väike väärtus tähendab, et viipa ei võeta pildi genereerimisel palju arvesse. Antud juhul oli automaatselt genereeritud viip üsna tähtis;
- ControlNet juhendamise kaalu (ingl *controlnet conditioning scale*) parameeter väärtusega 0,9 vahemikust vahemikust 0,0–1,0. See parameeter määrab, kui palju arvestatakse tingimuslikke sisendeid, kus väike väärtus ei arvesta neid ja suur väärtus

arvestab neid täielikult. Antud juhul oli tähtis, et mudel neid arvestaks, aga oleks ka natukene loov.

- Eeldefineeritud negatiivne viip (ingl *negative prompt*), mis kirjeldab, milliseid objekte peaks mudel pildi genereerimisel vältima ning selleks on loodud mitu viipa (Rodriguez, 2025). Universaalsest negatiivsest viibast võeti mõned, kuna mudelil on teksti sisendi pikkuse piirang ja negatiivse viiba väärtuseks sai "*deformed, mutated, malformed, extra limbs, extra fingers, extra hands, distorted, bad proportions, disfigured, missing limbs, fused fingers, floating limbs, broken anatomy, low quality, blurry, pixelated, ugly, glitch, error, duplicate, collage, jpeg artifacts*".

Töö kirjutamise ajal on Stable Diffusion v1.5 repositooriumis (Rombach et al., 2024) kirjas, et repositooriumit nimega RunwayML enam ei toetata ja mudeli "runwayml/stable-diffusion-v1-5" asemel tuleks kasutada mudelit "sd-legacy/stable-diffusion-v1-5" või "stable-diffusion-v1-5/stable-diffusion-v1-5". Kuigi repositooriumit ei toetata, on mudel siiski laialdaselt kasutuses. Seda näitasid igakuised miljonid allalaadimised Hugging Face lehelt. Kuna alla laaditud mudelid töötavad lokaalselt ka edaspidi, siis otsustati mudel sellisel viisil alles jätta, sest tööpõhimõtte poolest töötavad Stable Diffusion v1.5 mudelid samamoodi, olenemata repositooriumist.

2.3.3 Joonistuse sisule automaatsete kirjelduste loomine

Käesolevas töös testiti erinevaid eeltreenitud tehisnägemisel põhinevaid mudeleid, nagu CLIP 1 ja CLIP 2 mudelid, BLIP mudel ja Florence 2 mudelil põhinevaid Florence-2-large ja Florence-2-base-PromptGen-v1.5 mudelid. Üheks probleemiks suurte nägemis-keelemudelite puhul on, et neid on treenitud päris piltidega, mitte lihtsate joonistustega, ning praegusel kasutusjuhul ei pruugi need kasutaja joonistusi alati täpselt tuvastada. Testimise käigus selgus, et osad võimsamad mudelid genereerisid viipa liiga aeglaselt ja genereeritud viip kirjeldas lihtsat joonistust liiga detailselt. Näiteks sisaldas iga viip väljendit "valge joonistus mustal taustal", mida realistliku pildi genereerimisel ei tohiks arvesse võtta. Selle tõttu olid parimad valikud CLIP 1 ja Florence-2-large mudelid, mis kirjeldasid joonistatud stseeni üldisemalt, aga ikkagi piisavalt täpselt. Nendest mudelitest sai valitud just Florence-2-large, kuna näiteks joonistatud linnu puhul genereeris Florence-2-large viibaks linnu, kuid CLIP 1 pakkus üldisemalt, et tegu on loomaga. Samuti oli Florence-2-large mudel töö algusfaasis juhendaja poolt soovitatud. Töös implementeeriti Florence-2-large toetudes loojate Hugging Face repositooriumi juhendile (Xiao & Haiping, 2024). Järgnev lõik kirjeldab protsessi lähemalt.

Antud repositooriumis on Microsofti poolt loodud Florence-2-large implementeeritud toetudes transformer teegile. ControlNet mudeliga sai transformer teek installitud, seega nüüd oli tarvis vajalikud moodulid importida ja mudel ning samanimeline sisendi töötleja "microsoft/Florence-2-large" alla laadida. Seejärel tuli määrata ühesõnaline viip, mis suunab mudelit kindlat ülesannet lahendama. Selles töös kasutati kahte tüüpi viipasid: objektituvastus (ingl *object detection*) ehk "<OD>" ja kirjeldus (ingl *caption*) ehk "<CAPTION>". Objektituvastus töötab kõige paremini, kui joonistatakse üks lihtne objekt, kuna see tagastab ühe sõna või sõnapaari. Kirjeldus tagastab ühe lause, seega sobib see kõige paremini kirjeldama mitme objektiga joonistust. Lisaks on võimalik genereerida detailsemaid kirjeldusi, aga antud juhul ei muutnud need tulemusi paremaks. Pärast viiba loomist tuli viip ja pilt sisendi töötlejale anda, mis omakorda anti sisendparameetriena mudelile. Nende põhjal genereerib ja tagastab mudel genereeritud ID-d ehk tekstisisendi tokeniseeritud kujud. Lõpuks genereerib protsessor ID-de põhjal teksti ja teeb selle süntaktiliselt korrektseks.

Kuna kasutaja joonistus on lihtne must-valge kritseldus, siis paratamatult leidus genereeritud viibas väljendeid nagu must-valge (*A black and white*) ja tumedal taustal (*on a black background*). Seetõttu oli oluline kirjeldust järeltöödelda enne kui see pildiloomele viibana sisendiks anda. Värvivalikut kirjeldavad väljendid ja üksikud sõnad nagu must, valge, joonis, kritseldus, taust ja kontuur tuli eemaldada pärast kirjelduse loomist. Mõnikord võib olla sama sõna ühe pildi genereerimisel ebavajalik, aga teise puhul vajalik ning seda ei tohiks eemaldada. Lisaks ei ole genereeritud kirjeldus alati samasugune, sellepärast ei olnud ka antud juhul ühte kindlat viisi, et sobimatuid väljendeid ja sõnu alati vältida või eemaldada. Selline lähenemine töötas piisavalt hästi ja ei teinud programmi märgatavalt aeglasemaks ning genereeritud viip ei mõjutanud genereeritud pilti halvasti.

2.4 Kasutatud riist- ja tarkvara

Demo loodi Windows operatsioonisüsteemiga arvutis, millel oli 16 GB mälu ja GeForce RTX 3060 Ti graafikakaart ning kogu protsess ühe pildi genereerimiseks võttis aega 10–13 sekundit. Demo seati üles Ubuntu operatsioonisüsteemiga natukene võimsamal masinal, millel oli GeForce RTX 3080 graafikakaart ja mis suutis tervet protsessi teostada 7–10 sekundiga. Joonistamiseks kasutati XP-Pen Artist Pro 15.6 graafikatahvlit ja pliiatsit.

Projekti alguses kasutati koodi kirjutamiseks ja erinevate komponentide testimiseks Jupyter Notebook tarkvara. Selleks, et demo töötaks kohalikus arvutis, mindi üle Visual Studio Code (VS Code) arenduskeskkonda. Töös kasutati VS Code keskkonda integreeritud programmeerimiskeele Python 3, täpsemalt 3.12.7 versiooni. Demos kasutatava

graafikakaardi kasutamiseks toetuti juba loodud videomaterjali juhenditele Windows (SL7 Tech, 2023) ja Ubuntu (Abstract programmer, 2023) operatsioonisüsteemides. Videote põhjal paigaldati CUDA 12 arvutusplatvormi jaoks PyTorch versioon 2.5.1. Seejärel installiti Nvidia CUDA Toolkit versioon 12.6 (12.8 Ubuntul) ning Nvidia cuDNN versioon 8.9.7 CUDA 12.x jaoks.

2.5 Tehisintellekti kasutamine töö teostamisel

Töös kasutati tehisintellekti juturoboti ChatGPT abi, et koodi kirjutamisel selgitada tekkinud vigade tähendust ja saada soovitusi nende parandamiseks. Näiteks küsiti tagaliideses torule antavaid sobilikke parameetrite väärtuseid, et tulemusi parandada. Samuti küsiti esiliideses Gradio kohandatud CSS-i loomise kohta, kuna komponentide suuremaks muutmise vajadus tuli demo üleval oleku ajal ja seda pidi kiiresti tegema. Pärast genereeritud soovitust kohandati stiililehte vastavalt vajadusele.

Tagaliideses pandi kokku viip tuvastatud objektist või stseenist, stiilist ja taustast. Kasutajaliideses pidi olema menüü stiili ja tausta valikute jaoks ning kuna neid on palju ja need peavad olema hästi sõnastatud, siis kasutati nende genereerimiseks tehisintellekti abi.

Kasutajale pakutavate stiilide nimekiri genereeriti viibaga "*I'm making a front-end AI app in Gradio where I let the user draw something on the screen which the AI can improve. I also want to let the user choose a style for the picture, could you give me the common styles that AI picture generators use?*" (ChatGPT versioon 4o). Selle tulemusel valiti välja kõige erinevamad 11 stiili: *"Realism", "Photorealistic CGI", "Impressionism", "Surrealism", "Pop Art", "Pixel Art", "Sketch & Ink Art", "Futurism", "Gothic", "Minimalism", "Anime"*.

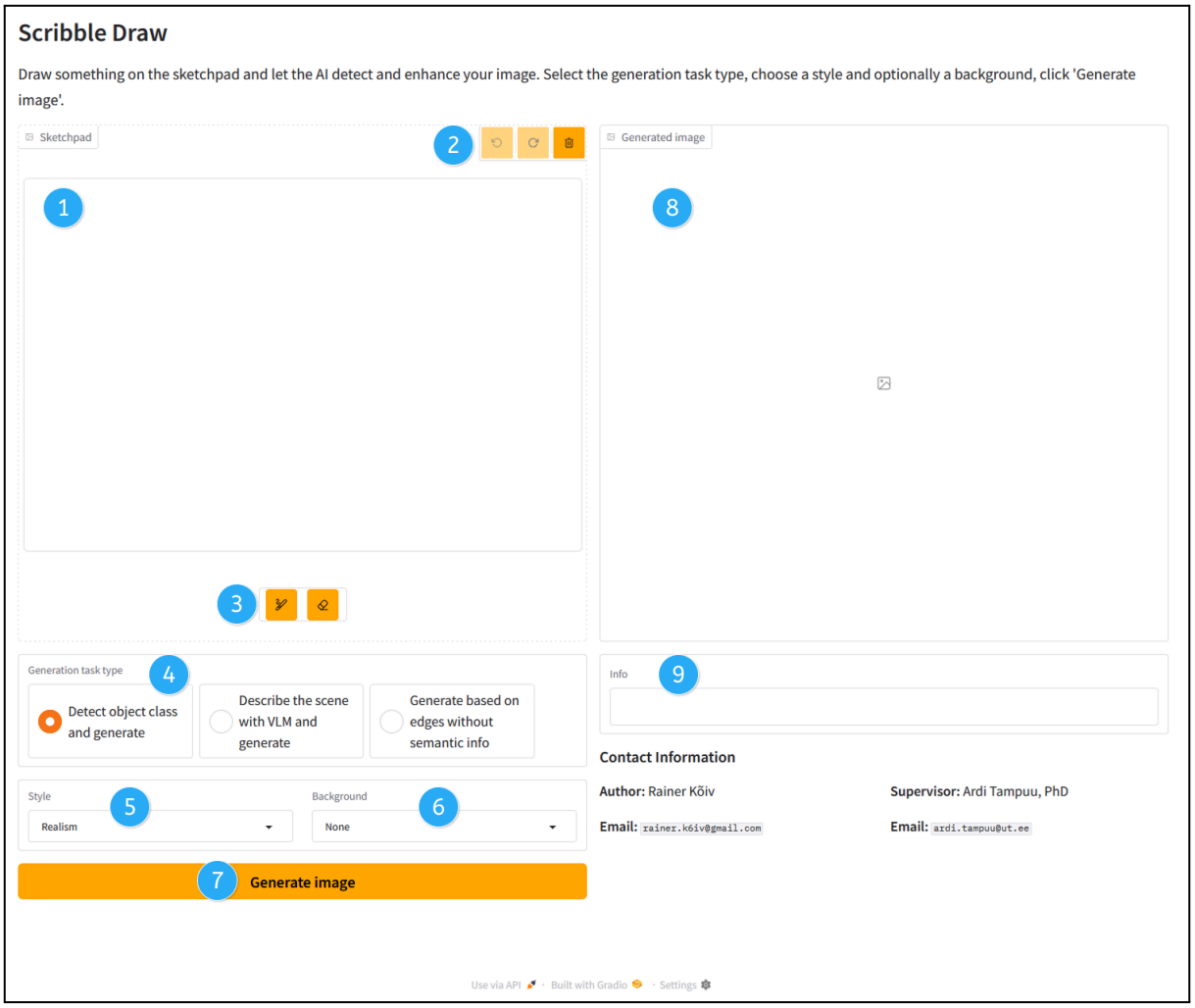
Igale stiilile oli vaja genereerida ka täpsemad sõnastused, mis viipa lisatakse, ning need leiti viiba "*I also need a prompt for each one. For example if the user selects realism style, then the drawing and a more detailed prompt related to realism is given to the image generator.*" abil (ChatGPT versioon 4o). Selle tulemused on toodud lisas 2. Vahetult pärast stiili valikuid genereeriti tausta valikud ja täpsemad sõnastused viibaga "*Give me background prompts too*", mis arvestas eelnevat konteksti (ChatGPT versioon 4o). Selle tulemusel saadi ja valiti neli tausta valikut – *"Natural", "Urban", "Studio Lighting", "Fantasy"*. Neile lisati juurde ka tühi sõne, kuna tausta valik pole nii tähtis kui stiil, siis oli taust valikuline. Täpsemad sõnastused on samuti toodud lisas 2.

3. Tulemused

Käesolevas peatükis antakse ülevaade valminud kasutajaliidesest ja programmi kui terviku tulemustest. Samuti antakse hinnang alammoodulite ehk mudelite tulemustele ja kirjeldatakse kasutajatelt tagasiside kogumist ning saadud tagasisidet. Lisas 3 on toodud valminud programmi lähtekood.

3.1 Loodud kasutajaliides

Töö üheks tähtsamaks osaks oli kasutajaliides, kuna selleks, et kasutaja joonistust töödelda, peab kasutajal olema mugav ja lihtne oma joonistust luua. Joonisel 7 on toodud valminud kasutajaliides.



Joonis 7. Valminud kasutajaliides.

Järgnevalt on kirjeldatud kasutajaliidese funktsionaalsed osad:

1. Joonistusala ehk lõuend.
2. Nupud “Võta tagasi”, “Tee uuesti” ja “Puhasta joonistusala”.
3. Nupud “Pliiats” ja “Kustutuskumm”. Nupule vajutades saab pliiatsi või kustutuskummi suurust muuta.
4. Genereerimise režiim.
5. Menüü stiili valikuks.
6. Menüü tausta valikuks.
7. Nupp pildi genereerimiseks.
8. Ala, kuhu ilmub genereeritud pilt.
9. Ala, kuhu ilmub info genereeritud pildi kohta, näiteks tuvastatud objekt või stseeni kirjeldus.

Autori arvates on loodud kasutajaliides piisavalt lihtne ja mugav ka neile, kes puutuvad sellega kokku esimest korda.

Kasutajaliidese tugevused:

- Kasutaja ei pea sisestama teksti – kogu tegevus toimub puutetundliku ekraani abil;
- Valikuid on vähe, mis aitab vältida segadust;
- Protsessi saab lihtsalt ja loogilises järjekorras läbi teha;
- Kasutajal on erinevad võimalused oma joonistust muuta või täiustada.

Kasutajaliidese puudused:

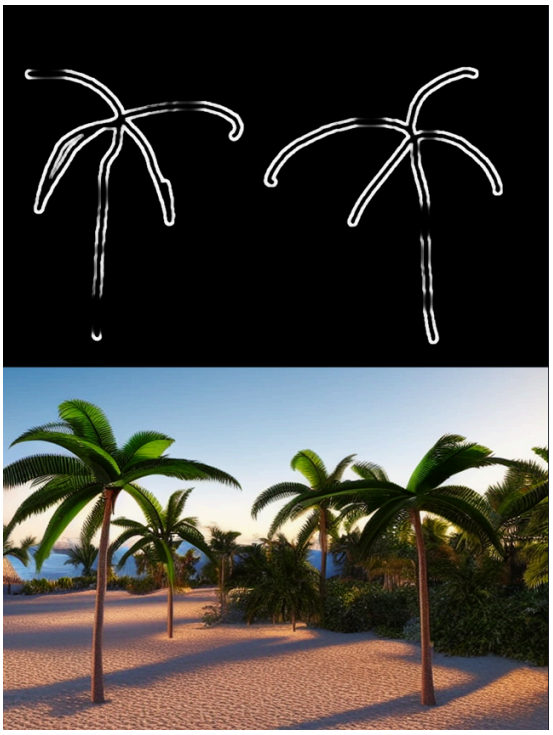
- Väiksematel ekraanidel ei pruugi liides hea välja näha.
- Tagasiside põhjal võiks lisada võimaluse pilt kas saata e-postile või laadida alla QR-koodi abil.
- Soovitati suurendada joonistusala ja genereeritud pildi suurust.
- Kui hiirekursor on aktiivne ja liigutakse joonistusalalt välja, siis hiirekursor jääb aktiivseks ja joonistamine jätkub – see on *sketchpad*-komponendist tulenev, autorist sõltumatu tehniline piirang.

Lisaks oli kasutajakogemus graafikatahvli ja sellega kaasas oleva pliiatsiga ebamugav, aga arvutis hiirega või puutetundliku ekraaniga probleeme polnud.

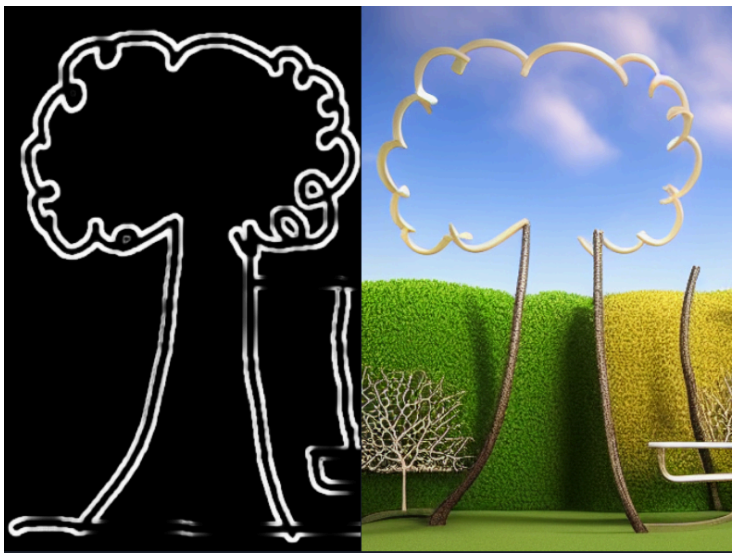
3.2 Programmi kui terviku tulemused ja hinnangud

Mudelid Stable Diffusion v1.5 ja ControlNet töötasid nii nagu olid kirjeldatud. Natukene probleemne koht oli ControlNet mudelit abistava HED mudeliga, mis tuvastas ühe joone mõlemad ääred ning selle tulemusel tekitas kaks joont ühe asemel. Väiksemate piltide puhul

tuleb see rohkem kasuks, näiteks kui kasutaja tahab ühe joone abil joonistada puutüve, siis joone kaks äärt võimaldavad seda hästi kujutada, nagu on toodud joonisel 8. Suuremate piltide puhul tekkisid selle tõttu mõnikord hõljuvad objektid ja jooned, näiteks nagu näha joonisel 9. Selle parandamiseks peab kasutaja oma joonistust kas muutma või laskma rakendusel proovida uuesti pilti genereerida. Samuti aitab selle vastu joonistatud objekti seest värviga täitmine, siis toimib HED nii nagu peab ja tuvastab ainult objekti välimised jooned. See on ka üks põhjus, miks kasutaja joonistust töödeldi HED mudeliga ja ei antud img2img mudeli sisendiks kohe töötlemata joonistus (käsitledes jooni piirjoontena), kuna siis saab kasutaja oma tulemust rohkem suunata – kas ta tahab joonistada lihtsate joontega või objekte ka seest ära täita.



Joonis 8. HED mudeli töödeldud väljund ja selle põhjal genereeritud pilt nn väiksema joonistuse puhul.



Joonis 9. HED mudeli töödeldud väljund ja selle põhjal genereeritud pilt nn suurema joonistuse puhul, objekt seest täitmata.

Tulles pildi genereerimise valikute juurde, siis kõige rohkem mõjutas genereeritud pilti valitud režiim.

- **Klassi tuvastamise režiimis** on nägemis-keelemudeli ülesanne jooniselt tuvastada üks kindel objekt. Sisendiks on töödeldud ehk äärte tuvastusega tehtud pilt ja viip, mis pannakse kokku tuvastatud objektist, stiilist ja taustast. Selline lähenemine sobis kõige rohkem piltidele, millel oli kujutatud üks lihtne objekt. Kuid siin tekkisid tuvastamisel ka mõned probleemid, näiteks mis on üks lihtne objekt, kas inimese nägu või silm. Kasutajad ei pruugi alati teada, kuidas asjad taustal töötavad ja mõnikord joonistati mitu objekti või olid joonistused mudeli jaoks liiga abstraktsed, et sealt ühte objekti õigesti tuvastada. Autori hinnangul töötas objektituvastus ootuspäraselt, suurem osa objekte tuvastati õigesti, ning režiim võiks ka tulevikus alles jääda, et pakkuda kasutajale rohkem võimalusi.
- **Stseeni kirjelduse režiimis** on nägemis-keelemudeli ülesanne joonist kirjeldada ja selle põhjal genereerida üks kirjeldav lause ehk pealdis. Sisendiks on töödeldud pilt ja viip, mis pannakse kokku genereeritud pealdisest, stiilist ja taustast. Selline lähenemine sobis kõige paremini piltidele, millel oli kujutatud mitu objekti, kuna siis loodi objektide põhjal stseeni kirjeldus. Hinnanguliselt andis see režiim üldiselt kõige paremad tulemused, kuna joonise kohta genereeriti rohkem infot olenemata, kas joonistati üks või mitu objekti. Kuid ka siin võib mudel objekte valesti tuvastada ja selle tulemusel genereerida teistsuguste objektidega pildi. Joonisel 10 on toodud näide, kus objekti või stseeni on valesti tuvastatud.

- **Ilma semantilise infota genereerimise režiimis** ei tuvastatud nägemis-keelemudeli abil joonistuselt midagi. Selles režiimis võeti sisenditeks töödeldud joonistus ja tekstiviip, mis koosnes ainult eeldefineeritud stiili ja tausta tekstidest, ning nende põhjal genereeriti pilt, mis arvestas joonistuse äärtega. See lähenemine sobis hästi üldiste stseenide loomiseks, kus ei soovitud konkreetset objekti, vaid pigem terviklikku kujutist. See režiim demonstreerib Stable Diffusion v1.5 ja ControlNet mudelite korrektselt toimivat koostööd, säilitades kõik sisestatud jooned ning luues visuaalselt piisavalt kvaliteetne tulemus.

Joonis 10. Nägemis-keelemudel on tuvastanud “linnu” asemel “looma”.

Visuaalne-keelemudel Florence-2-large sai antud ülesannetega üsna hästi hakkama, tuvastades erinevaid objekte ja kirjeldades stseene. Põhilised puudused olid, et mudel ei saanud väga detailse ja kirju pildi kirjeldamisega hakkama ning üsna tihti tuvastati arusaamatutelt piltidelt loom, aga mudel ei täpsustanud, milline loom.

Pildi genereerimine Stable Diffusion v1.5 mudeliga töötas väga hästi. Mõned näited töödeldud sisendpiltidest ja genereeritud piltidest on toodud joonisel 11 ning rohkem näiteid leiab lisast 4. Tegu pole väga võimsa mudeliga, näiteks ei suuda mudel genereerida koherentset teksti, kvaliteetseid nägusid ning fotorealistlikke pilte [**V21**]. Küll aga on mudel piisavalt kiire, et lihtsatest joonistustest kasutaja jaoks talutava ooteajaga pilte genereerida. Samuti aitas ControlNet mudel joonistatud tekstile piirjooni lisada, mis säilitas joonistustel olnud tekste ka lõplikel piltidel. Genereeritud piltide proportsioonid võisid kohati olla imelikud, kuna see tulenes kasutaja abstraktse joonistuse tõlgendamisest, mis võib mudeli jaoks keeruline olla. Selle probleemi leevendamiseks saaks ControlNet mudeli mõju vähendada, kuid siis läheks osa kasutaja joonistusest kaduma.

Joonis 11. Töödeldud kasutaja joonistused ja genereeritud pildid.

Kokkuvõtteks võib öelda, et kõik mudelid said ülesandega üsna hästi hakkama. Ootamatu tulemuse põhjuseks oli tõenäoliselt tuvastusmoodul ehk nägemis-keelemudel, mis abstraktset joonistust valesti tuvastas või kirjeldas.

3.3 Avalik demo

Lahenduse esmane versioon oli kasutamiseks väljas kuupäevadel 11.03.2025 kuni 15.03.2025 Delta õppehoones. Demo seati üles Delta töötajatele ligipääsetavale alale – kolmanda korruse Vaba Lava lähedale kohviruumi. Demo kohviruumis on toodud joonisel 12. Kuupäevadel 11.03.2025 kuni 14.03.2025 asus see kohviruumis ning kuupäeval 15.03.2025 oli see vilistlaspäevaga seoses teisel korrusel külastajatele kasutada.

Joonis 12. Demo Vaba Lava lähedal kohviruumis 11.03.2025.

Demoperioodi jooksul küsiti kasutajatelt tagasisidet mitmete demolahenduse aspektide kohta, kasutades Google Forms keskkonda. Täpsed küsimused on toodud lisas 5. Samuti küsiti kasutajatelt kohapeal tagasisidet. Küsimuste vastustele reageeriti jooksvalt ning tehti uuendusi, kogutud vastuseid analüüsiti ning tulemused on esitatud peatükis 3.4.

Avaliku demoperioodi jooksul tõsteti kasutajaliidese komponente ümber ja muudeti seda pidevalt. Jooksvalt saadud tagasisides tuli välja probleem, et nupud ja nende ikoonid ning

tekst olid liiga väikesed, samuti oleks võinud joonistusala olla suurem, kuna ekraanil oli ruumi üle. Lisaks oli algne tekst segane ja kasutajad ei saanud täpselt aru, mida režiimi nupud teevad, ning info genereeritud pildi kohta ei olnud informatiivne. Kõiki neid probleeme sai demo ülevaloleku ajal parandatud. Lisaks otsustati muuta nuppude värv oranžiks, et ühtlustada kasutajaliidese välimust Gradio vaiketeemaga, mis põhineb valge ja oranži värvikombinatsioonil.

3.4 Kasutajate tagasiside

Demo kõrval oli võimalik QR-koodi abil vastata Google Forms tagasiside küsimustikule. Kõige suurem huvi oli demo vastu esimestel päevadel, mil suurem osa tagasisidest ka saadi.

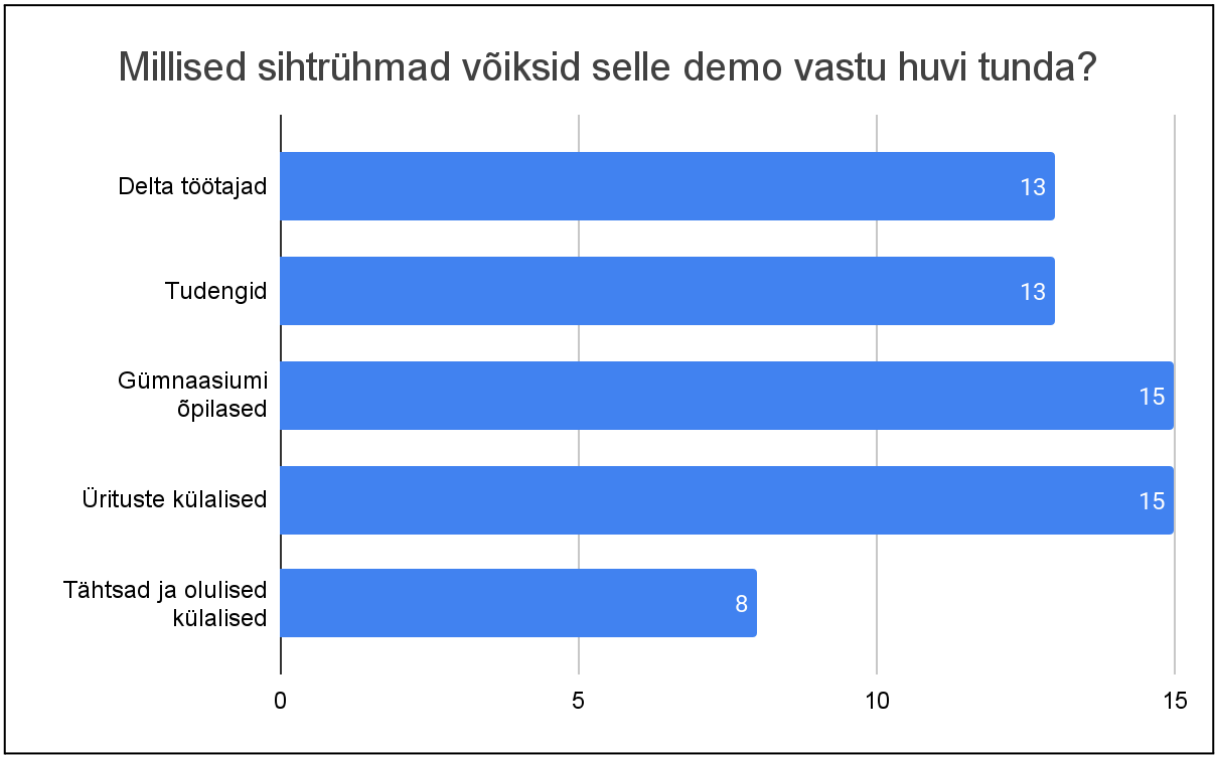
Küsimustiku eesmärk oli saada tagasisidet kasutajaliidesele ja genereeritud pildi kvaliteedile. Lisaks sisaldas see küsimusi, et teada saada ning valideerida juhendaja arvamust, kes võiks olla antud demo sihtrühm ja kui demo Deltasse üles jätta, mis eesmärki see täidaks. Samuti oli võimalik avatud vastusena soovitusi anda. Kokku vastas küsimustikule 18 inimest. Kasutajaliidest uuendati jooksvalt laekunud tagasiside põhjal, ning seega tuleb tagasiside interpreteerimisel arvestada, et osa sellest ei pruugi enam olla ajakohane.

Joonisel 13 on toodud küsimustikule vastanute arvamus antud demo kohta.

Joonis 13. Küsimuse “Kas käesolev demo oleks huvitav ja sobiks Delta koridori?” vastused.

Joonise põhjal võib öelda, et 13 inimest arvasid, et demo oleks huvitav ja sobiks püsivalt Delta koridori ülespanekuks. 4 inimest arvasid, et demo vajab veel parandusi, aga sobiks Delta koridori. Ning 1 inimene arvas, et demo ei sobiks püsivalt Deltasse, kuid seda võib üritustel näidata. Kokkuvõtvalt arvati, et käesolev demo on huvitav ja sobiks püsivaks eksponaadiks Delta koridori.

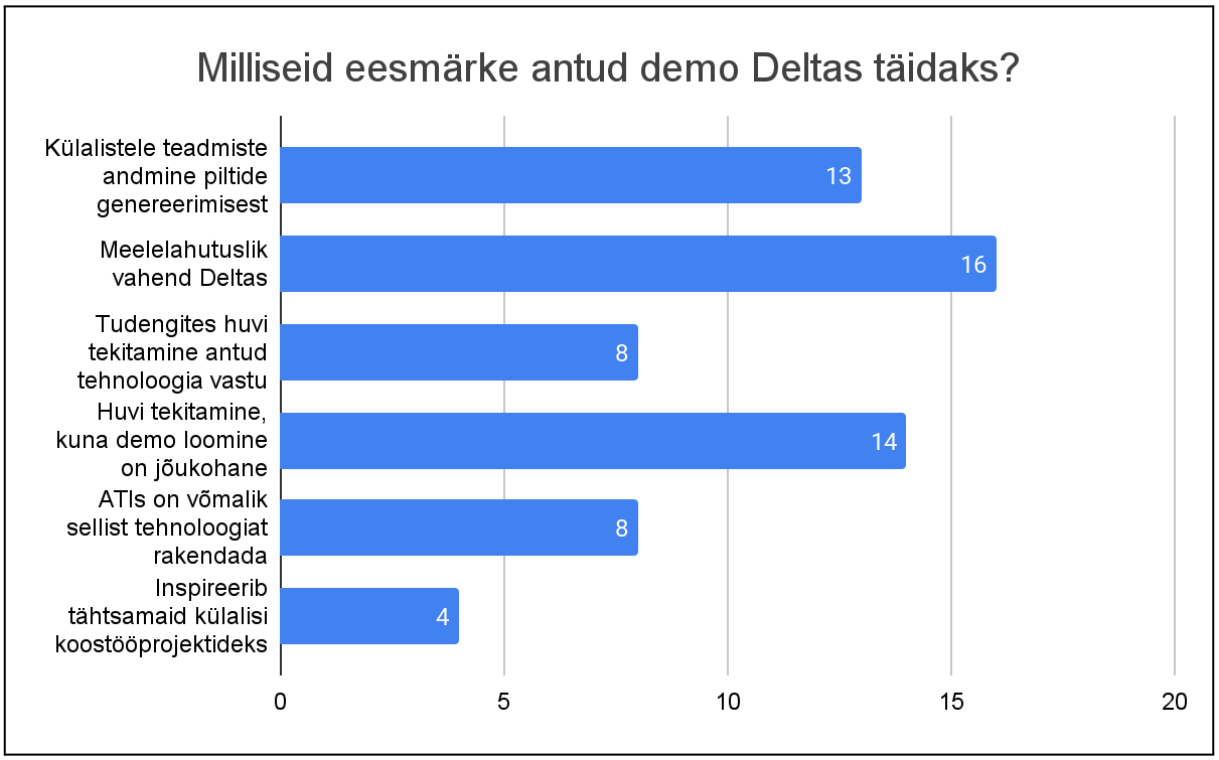
Joonisel 14 on toodud küsimustikule vastanute arvamused seoses, kes võiksid olla käesoleva demo sihtrühmad.



Joonis 14. Küsimuse "Millised sihtrühmad võiksid selle demo vastu huvi tunda?" vastused.

Vastuste põhjal peeti peamisteks sihtrühmadeks gümnaasiumiõpilasi ning vilistlaspäeva või muude ürituste külastajaid. Lisaks sobiksid sihtrühmadeks ka tudengid ja Delta töötajad. Vähem kui pooled vastanutest arvasid, et potentsiaalseteks sihtrühmadeks võiksid olla ka olulisemad külalised, nagu äripartnerid või väliskülalised. Kokkuvõtvalt võib öelda, et demo kasutajad hindasid seda huvipakkuvaks väga laiale hulgale Delta külastajatest.

Joonisel 15 on toodud küsimustikule vastanute arvamused sellest, mis võiks olla demo eesmärk.



Joonis 15. Küsimuse “Milliseid eesmärke antud demo Deltas täidaks?” vastused.

Vastuste põhjal võib järeldada, et enamik inimesi peab demot sobivaks meelelahutuslikuks vahendiks Deltasse. Samuti leiti, et demo näitab, et sellise rakenduse loomine on tudengitele täiesti jõukohane ning nad suudaksid selle edukalt ellu viia. Lisaks pakuks demo Delta külastajatele hea ülevaate tehisintellekti võimekusest pildigeneratsiooni vallas. Natuke alla poole vastanutest leidis, et demo võiks äratada õpilastes huvi antud tehnoloogia vastu ning näitab, et arvutiteaduse instituudil on suutlikkus selliseid lahendusi rakendada. Kõige vähem arvati, et demo võiks inspireerida tähtsamaid külalisi (nt äripartnereid või väliskülalisi) koostööprojektide algatamiseks. Kokkuvõtvalt võib öelda, et lahendust peeti keskmiselt pigem meelelahutuslikuks ning vähem tehnoloogilist tipptaset demonstreerivaks.

Küsimustikus paluti kasutajatel hinnata genereeritud pildi kvaliteeti skaalal 1–10. Enamik hinnanguid jäi vahemikku 8–10 (10 vastajat) ja sellest võib järeldada, et üldiselt oli pildi genereerimise mudel piisavalt võimas, et kvaliteetseid pilte genereerida. Üljäänud 8 vastajat andsid hinnangu vahemikus 6–7 ning vahemikus 1–5 hinnanguid ei olnud. Sellest järeldub, et mudeli genereeritud pildid ei olnud just kõige paremad, kuid samas ei olnud need ka väga ebakvaliteetsed. Kokkuvõttes võib öelda, et genereeritud pildi kvaliteeti peeti üldiselt piisavaks.

Lisaks paluti vastajatel hinnata, mil määral nende joonistus mõjutas lõpp-tulemust. Enamik hinnanguid koondus skaalavahemikku 7–10 (11 vastajat), mis viitab üldisele rahulolule.

Keskmise hinnangu (4–6) andis 5 vastajat ning ühtegi väga madalat hinnangut (1–3) ei esinenud. See viitab sellele, et kasutajad tundsid enamasti, et nende joonistusel oli mõju loodud pildi kujunemisele.

Kuna kasutajaliidest täiendati jooksvalt, siis küsimus, mis on seotud režiimi, tausta ja stiili valikute arusaadavusest, on aegunud. Sellel hetkel arvati, et režiimivalikud on vajalikud, kuid pole väga arusaadavad. Stiili ja tausta valikud on arusaadavad ja piisavad. Samuti sooviti arvamust, kas kasutaja võiks pilte alla laadida, millele 16 vastanutest arvasid, et võiks ning 2 vastanut ei pidanud seda vajalikuks. Sellegipoolest otsustati selline võimalus jätta implementeerimata, kuna see polnud väga vajalik mõne lihtsa pildi jaoks.

Kogutud vabavormiliste vastuste põhjal järeldub, et suurimaks probleemiks oli kasutajaliidese visuaalne osa ning selle komponentide paigutus, sealhulgas nuppude suurus ja asetus, samuti selge eesmärgi ja sõnumi puudumine. Neid kitsaskohti parandati jooksvalt. Samuti tehti tagasisides mitmeid ettepanekuid lisafunktsionaalsuse osas, näiteks võimalus pilti alla laadida või saata e-postile. Positiivselt hinnati liidese lihtsust ja loovust toetavat ülesehitust. Ülejäänud kasutajaliidese valikud olid samuti arusaadavad ja kasulikud. Tagasisidest tulid ka välja autorist sõltumatud probleemid, nagu graafikatahvli ja pliiatsi ebamugavus ning *sketchpad*-komponendi eripära, kus hiirekursor jääb aktiivseks, kui liigutakse joonistusalalt välja.

2.5 Rakenduse võimalikud edasiarendused

Käesolev lahendus töötab piisavalt hästi, et täiustada kasutaja lihtsaid joonistusi tehisintellekti abil. Mõned edasiarendused oleksid näiteks mitmevärviliste joonistuste loomine ja nende põhjal piltide genereerimine, kuna praegu on võimalik joonistada ainult musta värviga valgele taustale. Rohkem genereeritud väljundpilte samas või erinevates stiilides, et kasutajal oleks suurem valik. Reaalajas tulemuste eelvaade, kus kasutaja näeb genereeritud pilti juba joonistamise käigus ning tehtud täiendused kajastuvad jooksvalt. Joonistuste salvestamine ja hiljem muutmise võimalus, et kasutajad saaksid oma pilte ka hiljem muuta.

Samuti oleks võimalik uurida võimsamaid ja kvaliteetsemaid pilte genereerivaid mudeleid ning testida erinevaid nägemis-keelemudeleid, et tuvastamist paremaks teha. Kuna Stable Diffusion mudeli sõnalisel viibal on piirang 77 tookenit, siis on võimalus stiilide ja taustade kirjeldusi täiustada nii, et need oleksid piisavalt lühikesed ja informatiivsed.

Kokkuvõte

Bakalaureusetöö eesmärk oli luua Delta õppehoone koridori väljapanekuks interaktiivne demo, kus tehisintellekt abil täiustatakse kasutaja lihtsaid joonistusi.

Töös keskenduti tehisintellekti alamkategoorias generatiivsele tehisintellektile ja täpsemalt difusioonimudelitele. Lisaks kirjeldati, kuidas saab lihtsaid kasutaja joonistusi töödelda ning nende põhjal tekstilist infot genereerida.

Töö raames loodi olemasolevate tehisintellekti mudelitega demoprogramm, kus kasutaja saab Gradio raamistikus loodud kasutajaliidese kaudu joonistada lihtsaid visandeid. Joonistuse, stiili, tausta ning režiimi – objekti tuvastuse, kirjelduse loomise või ilma semantilise infota genereerimise – põhjal genereeritakse uus täiustatud pilt. Joonistustelt objektide tuvastamiseks ja kirjelduste genereerimiseks kasutati Florence-2-large nägemis-keelemudelit. Joonistuse töötlemiseks kasutati Stable Diffuison ControlNet mudelit ja piltide genereerimiseks ControlNet ning Stable Diffusion v1.5 mudeleid.

Valminud programmile andsid kasutajad avaliku demo ajal tagasisidet Google Forms küsimustiku kaudu. Tagasiside põhjal tehti parandusi ja täiendusi ning üldiselt jäädi demoga rahule. Mudelid said oma ülesandega hästi hakkama ja genereeritud piltide kvaliteet oli piisav. Demo sihtrühmadeks peeti peamiselt õpilasi, tudengeid, Delta töötajaid ja ürituste külalisi. Peamised demo eesmärgid Deltas oleks olla meelelahutuslik vahend, pakkuda teadmisi piltide genereerimisest ja tekitada huvi sellise tehnoloogia vastu.

Viidatud kirjandus

Abid, A., Abdalla, A., Abid, A., Khan, D., Alfozan, A., & Zou, J. (2019). *Gradio: Hassle-Free Sharing and Testing of ML Models in the Wild* (No. arXiv:1906.02569). arXiv. https://doi.org/10.48550/arXiv.1906.02569 (11.05.2025).

Abstract programmer (Director). (2023, aprill 15). *How to install PyTorch on Ubuntu 22.04 with Nvidia graphics card* [Videosalvestus]. https://www.youtube.com/watch?v=c0Z_ItwzT5o (11.05.2025).

Aristimuño, I. (2023, november 28). An Introduction to Diffusion Models and Stable Diffusion. *Marvik*. https://blog.marvik.ai/2023/11/28/an-introduction-to-diffusion-models-and-stable-diffusion/ (11.05.2025).

Banh, L., & Strobel, G. (2023). Generative artificial intelligence. *Electronic Markets*, *33*(1), 63. https://doi.org/10.1007/s12525-023-00680-1 (11.05.2025).

Bergmann, D., & Stryker, C. (2024, august 21). *What are Diffusion Models? | IBM*. What Are Diffusion Models? | IBM. https://www.ibm.com/think/topics/diffusion-models (11.05.2025).

Garrido-Munoz, C., Rios-Vila, A., & Calvo-Zaragoza, J. (2025). *Handwritten Text Recognition: A Survey* (No. arXiv:2502.08417). arXiv. https://doi.org/10.48550/arXiv.2502.08417 (11.05.2025).

Gradio dokumentatsioon. (s.a.). Gradio Documentation. https://www.gradio.app/docs (11.05.2025).

Ho, J., Jain, A., & Abbeel, P. (2020). *Denoising Diffusion Probabilistic Models* (No. arXiv:2006.11239). arXiv. https://doi.org/10.48550/arXiv.2006.11239 (11.05.2025).

Hutson, G. (2023, märts 29). Creating doodles with HED detection and ControlNet | Python-bloggers. *Creating Doodles with HED Detection and ControlNet | Python-Bloggers*. https://python-bloggers.com/2023/03/creating-doodles-with-hed-detection-and-controlnet/ (11.05.2025).

IBM Technology (Director). (2025, jaanuar 30). *Diffusion Models for AI Image Generation*

[Videosalvestus]. https://www.youtube.com/watch?v=x2GRE-RzmD8 (11.05.2025).

Image-to-image. (s.a.). https://huggingface.co/docs/diffusers/using-diffusers/img2img (11.05.2025).

Karagiannakos, S., & Adaloglou, N. (2022, september 29). *How diffusion models work: The math from scratch*. AI Summer. https://theaisummer.com/diffusion-models/ (11.05.2025).

O'Connor, R. (2022, mai 12). *Introduction to Diffusion Models for Machine Learning*. Introduction to Diffusion Models for Machine Learning. https://assemblyai.com/blog/diffusion-models-for-machine-learning-introduction (11.05.2025).

Rodriguez, R. (2025, aprill 27). *180+ Best Stable Diffusion Negative Prompts with Examples*. 180+ Best Stable Diffusion Negative Prompts with Examples. https://www.aiarty.com/stable-diffusion-prompts/stable-diffusion-negative-prompt.htm (11.05.2025).

Rombach, R., Blattmann, A., Lorenz, D., Esser, P., & Ommer, B. (2022). *High-Resolution Image Synthesis with Latent Diffusion Models* (No. arXiv:2112.10752). arXiv. https://doi.org/10.48550/arXiv.2112.10752 (11.05.2025).

Rombach, R., Blattmann, A., Lorenz, D., Esser, P., & Ommer, B. (2024, september 6). *Stable-diffusion-v1-5/stable-diffusion-v1-5 · Hugging Face*. https://huggingface.co/stable-diffusion-v1-5/stable-diffusion-v1-5 (11.05.2025).

Sajid, H. (2024, juuli 16). *Top 10 Multimodal Models* [Blog]. Top 10 Multimodal Models | Encord. https://encord.com/blog/top-multimodal-models/ (11.05.2025).

Sengar, S. S., Hasan, A. B., Kumar, S., & Carroll, F. (2024). Generative artificial intelligence: A systematic review and applications. *Multimedia Tools and Applications*. https://doi.org/10.1007/s11042-024-20016-1 (11.05.2025).

SL7 Tech (Director). (2023, aprill 10). *How to setup NVIDIA GPU for PyTorch on Windows 10/11* [Videosalvestus]. https://www.youtube.com/watch?v=r7Am-ZGMef8 (11.05.2025).

Zhang, L., & Agrawala, M. (2023, veebruar 24). *Lllyasviel/sd-controlnet-scribble · Hugging Face*. https://huggingface.co/lllyasviel/sd-controlnet-scribble (11.05.2025).

Zhang, L., Rao, A., & Agrawala, M. (2023). *Adding Conditional Control to Text-to-Image Diffusion Models* (No. arXiv:2302.05543). arXiv. https://doi.org/10.48550/arXiv.2302.05543 (11.05.2025).

What is Stable Diffusion? - Stable Diffusion AI Explained - AWS. (s.a.). Amazon Web Services, Inc. https://aws.amazon.com/what-is/stable-diffusion/ (11.05.2025).

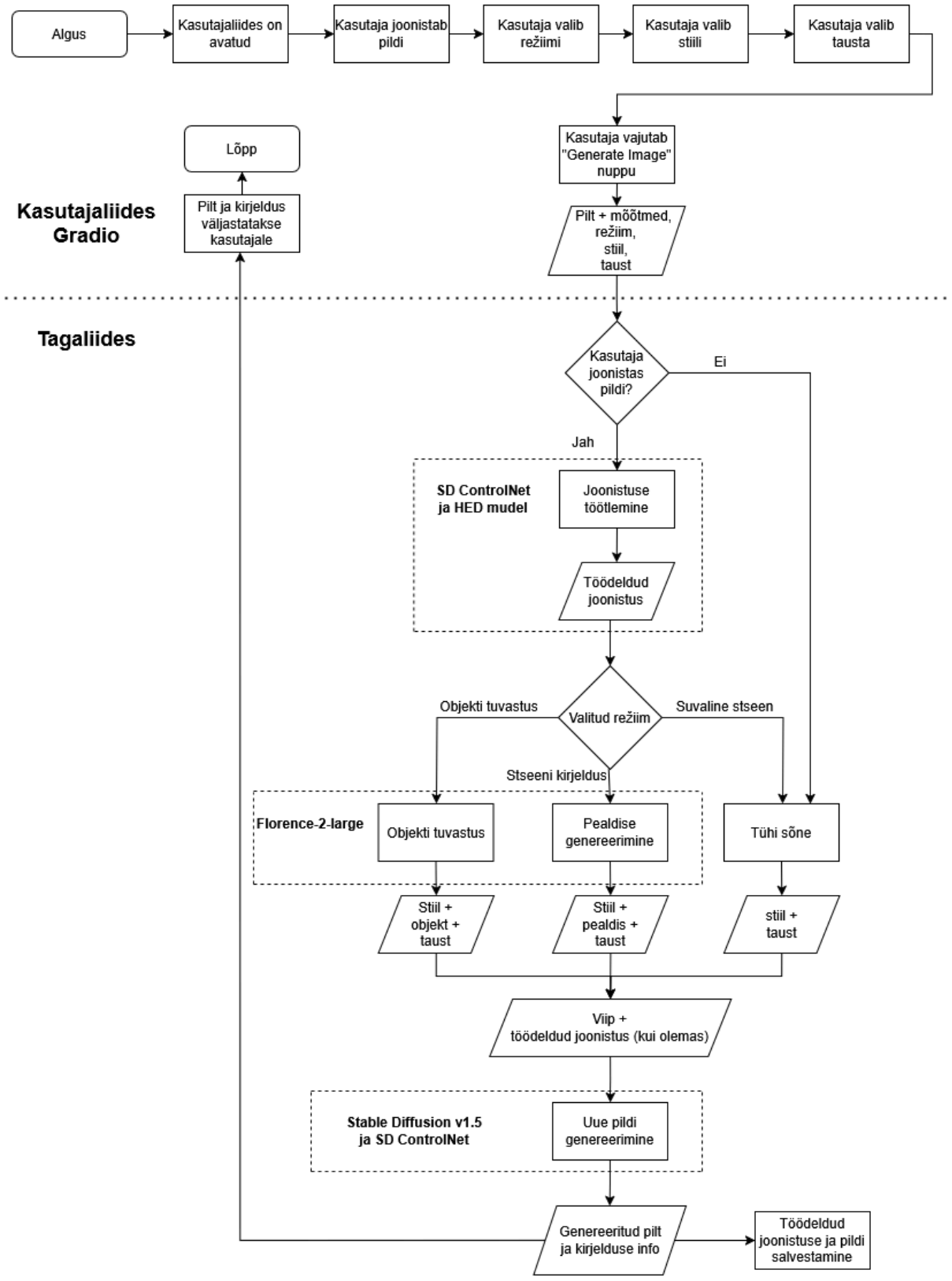
Xiao, B., & Haiping, W. (2024, juuni 15). *Microsoft/Florence-2-large · Hugging Face*. https://huggingface.co/microsoft/Florence-2-large (11.05.2025).

Xiao, B., Wu, H., Xu, W., Dai, X., Hu, H., Lu, Y., Zeng, M., Liu, C., & Yuan, L. (2024). Florence-2: Advancing a Unified Representation for a Variety of Vision Tasks. *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 4818–4829. https://doi.org/10.1109/CVPR52733.2024.00461 (11.05.2025).

Yang, L., Zhang, Z., Song, Y., Hong, S., Xu, R., Zhao, Y., Zhang, W., Cui, B., & Yang, M.-H. (2024). *Diffusion Models: A Comprehensive Survey of Methods and Applications* (No. arXiv:2209.00796). arXiv. https://doi.org/10.48550/arXiv.2209.00796 (11.05.2025).

Lisad

Lisa 1. Programmi arhitektuur ja töövoog



Lisa 2. Tehisintellektiga genereeritud stiili ja tausta kirjeldused

Stiili kirjeldused:

1. **Realism:** *"A highly detailed and realistic depiction of {object}, with accurate lighting, shadows, and textures. The colors are true-to-life, and the composition looks like a professional photograph or a classical realistic painting."*.
2. **Photorealistic CGI:** *"A hyper-realistic CGI rendering of {object}, with highly detailed textures, perfect lighting, and a polished, modern 3D appearance."*.
3. **Impressionism:** *"A beautiful impressionist painting of {object}, with soft brushstrokes, vibrant colors, and a dreamy atmosphere. The lighting is natural, capturing a fleeting moment with artistic motion and a focus on light and color blending."*.
4. **Surrealism:** *"A surreal and dreamlike interpretation of {object}, featuring unexpected, bizarre, and otherworldly elements. The scene is imaginative and mysterious, blending reality with fantasy in a way that defies logic."*.
5. **Pop Art:** *"A bold and colorful pop-art version of {object}, inspired by Andy Warhol and Roy Lichtenstein. The colors are bright and saturated, with thick outlines and a comic book or advertisement-style aesthetic."*.
6. **Pixel Art:** *"A retro pixel-art rendition of {object}, with a low-resolution 8-bit or 16-bit aesthetic. The image consists of small square pixels, giving it a nostalgic video game look with bright, limited colors."*.
7. **Sketch & Ink Art:** *"A refined high-quality sketch of {object}, with expressive linework and subtle shading. The drawing maintains a hand-drawn feel, similar to concept art or ink illustrations. Crisp, confident lines use cross-hatching or stippling for depth, resembling expert concept art or ink illustrations."*.
8. **Futurism:** *"A high-tech, futuristic interpretation of {object}, with neon lights, sleek metallic surfaces, and a sense of movement. The artwork features advanced technology, cyberpunk elements, and a futuristic cityscape."*.
9. **Gothic:** *"A dark and moody gothic illustration of {object}, with high contrast lighting, intricate Victorian-inspired details, and a sense of mystery. The atmosphere is eerie and dramatic, evoking gothic horror themes."*.
10. **Minimalism:** *"A minimalist and clean depiction of {object}, with simple shapes, flat colors, and little to no extra detail, creating a modern, stylish aesthetic."*.
11. **Anime:** *"A vibrant and expressive anime-style illustration of {object}, featuring smooth shading, large expressive eyes, and dynamic character design. The colors are*

bright, with detailed backgrounds and action-oriented composition, inspired by Japanese animation.".

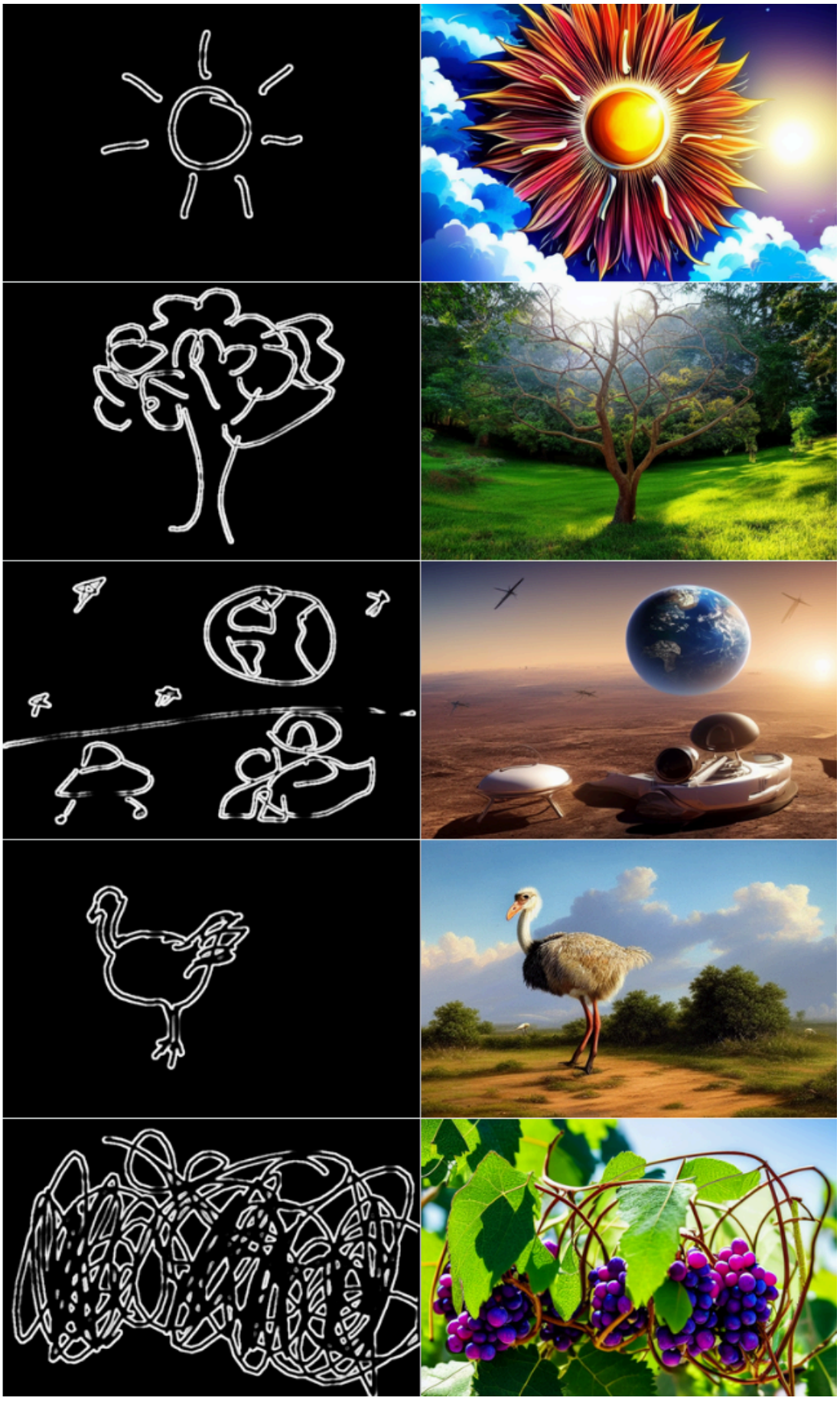
Tausta kirjeldused:

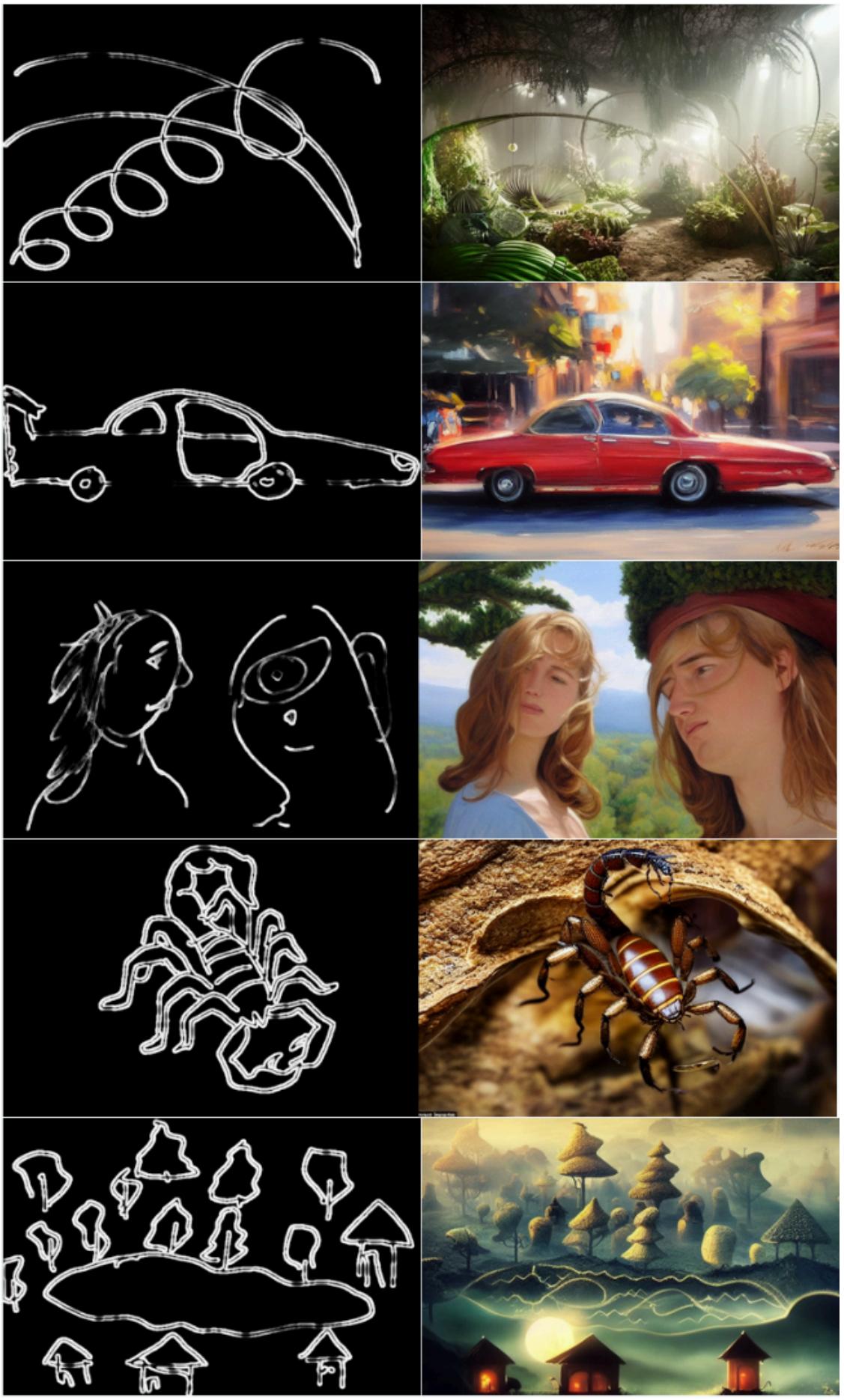
1. **None:** *""*.
2. **Natural:** *"A natural environment with lush greenery, blue skies, and soft sunlight, evoking a peaceful and organic feel."*.
3. **Urban:** *"An urban setting with architectural elements, providing a sense of city life."*.
4. **Studio Lighting:** *"A controlled lighting environment, highlighting the subject professionally."*.
5. **Fantasy:** *"A dreamy, imaginative background with a surreal or otherworldly atmosphere."*.

Lisa 3. Valminud programmi lähtekood

Programmi lähtekood on leitav GitHub veebikeskkonnas: https://github.com/RainerKoiv/ScribbleDraw.

Lisa 4. Rohkem näiteid kasutaja joonistustest ja genereeritud piltides





Lisa 5. Tagasiside küsimused

Küsimus 1:

Do you think this demo would be an interesting exhibit in the hallways of Delta?

Vastusevariandid:

- Yes, it could be in the hallway like other existing demos (magic mirror, ID card demo)
- It needs more work, but it could be in the hallway like other existing demos
- It is sufficient quality to be displayed during open-doors events and exhibitions, but not as a permanent demo
- This demo is not good or entertaining enough

Küsimus 2:

Which guests of Delta do you think this demo would interest or be entertaining for?

Vastusevariandid:

- Employees of Delta
- Students
- High-school students
- Guided tour guests during events such as Alumni day
- VIP guests such as business representatives, foreign guests

Küsimus 3:

If this demo was placed permanently in Delta, what goals would it serve in your opinion?

Vastusevariandid:

- Educate guests of Delta about the power of image generation technology (assuming they do not know enough about this)
- Make the space in Delta more entertaining
- Attract interest in this technology in potential students by looking interesting
- Attract interest in students by showing it is achievable at their competence level (at Bsc level)
- Demonstrate that our institute is capable of applying such technologies (if this demonstration is needed)
- Generate collaboration ideas in VIP guests by demonstrating the technology

Küsimus 4:

From a scale of 1 to 10, how would you rate the quality of the generated image?

Vastusevariandid:

Valik skaalal 1–10 (kus 1 = väga halb kvaliteet, 10 = väga hea kvaliteet)

Küsimus 5:

Did your drawing affect the generated output to a satisfactory degree?

Vastusevariandid:

Valik skaalal 1–10 (kus 1 = joonistusel polnud üldse mõju, 10 = joonistus mõjutas tulemust täiel määral ja rahuldavalt)

Küsimus 6:

What is true about the current set of options the user can pick?

Vastusevariandid:

- Generation task type ("specific object"/"specific scene"/"random scene") is useful
- Generation task type ("specific object"/"specific scene"/"random scene") options are understandable
- There are too many image style (realistic, ...) options and it is confusing
- The "background" options are sufficient
- The "background" options are understandable

Küsimus 7:

Give feedback on the current UI, is it too simple, too complex, confusing (what aspect in particular)? What could make it better?

Vastusevariandid:

Vabavormiline vastus

Küsimus 8:

Do you think there should be a download option (both scribble and generated image)?

Vastusevariandid:

- yes
- no

Lisa 6. Litsents

Lihtlitsents lõputöö reprodutseerimiseks ja üldsusele kättesaadavaks tegemiseks

Mina, **Rainer Kõiv**,

1. annan Tartu Ülikoolile tasuta loa (lihtlitsentsi) minu loodud teose

   Tehisintellekti abiga joonistamine kasutades Stable Diffusion Img2Img mudelit,

   mille juhendaja(d) on **Ardi Tampuu**,

   reprodutseerimiseks eesmärgiga seda säilitada, sealhulgas lisada Tartu Ülikooli digitaalarhiivi kuni autoriõiguse kehtivuse lõppemiseni;

2. annan Tartu Ülikoolile loa teha punktis 1 nimetatud teos üldsusele kättesaadavaks Tartu Ülikooli veebikeskkonna, sealhulgas digitaalarhiivi kaudu Creative Commonsi litsentsiga CC BY NC ND 4.0, 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.

Rainer Kõiv
15.05.2025