Duckify

Week 4 review

Milestone

About the week 4

Using the **complete pipeline**, draw **AI generated contours** with a **selected GenAI solution**, on **a duck**, installed on a **stable support**, using **multiple colors**, while changing **colors automatically**

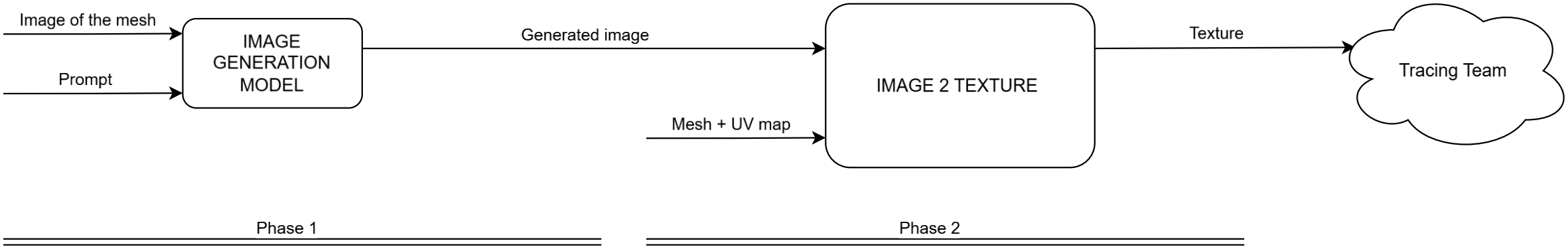
GenAI

- Assess MV-Adapter on Disco, perform fine-tuning and prompt engineering, and evaluate what it can reliably generate, identify its limitations, and determine whether it is a viable long-term solution. ☑

GenAI

- Integrate & Develop the AI Pipeline Solution with Disco to the global project pipeline : ☒
  - Why : Time constraint with the GenAi group
  - MV-Adapter docker: ok

Image 2 Texture pipeline



Black Forest Labs

FLUX KONTEXT 1
~32 GB VRAM

stability.ai

SDXL
~32 GB VRAM

Phase 1:Flux image generation solution

Prompt

Body and head: solid green.
Beak: solid yellow.
Paint a simple red pirate band across the head where the eyes are located. The band must be a flat painted stripe only and must not include a knot or cloth pieces. Paint a simple red diagonal stripe across the body to represent a pirate sash.

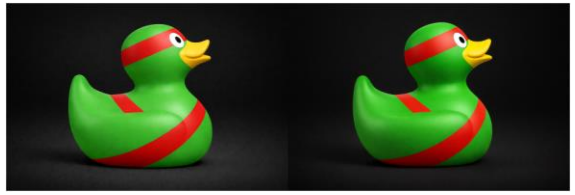
Negative Prompt

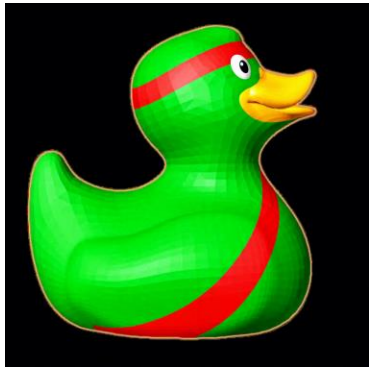
Strictly preserve the exact **proportions, orientation, silhouette, and shape** of the model.

model

Black Forest Labs

3 min



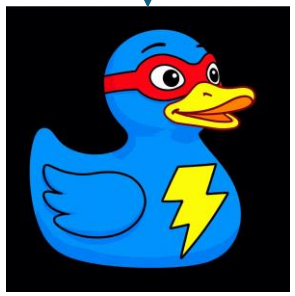


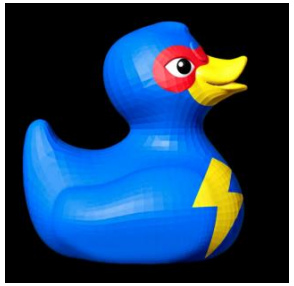
Above average

Average

Phase 1: Benchmark

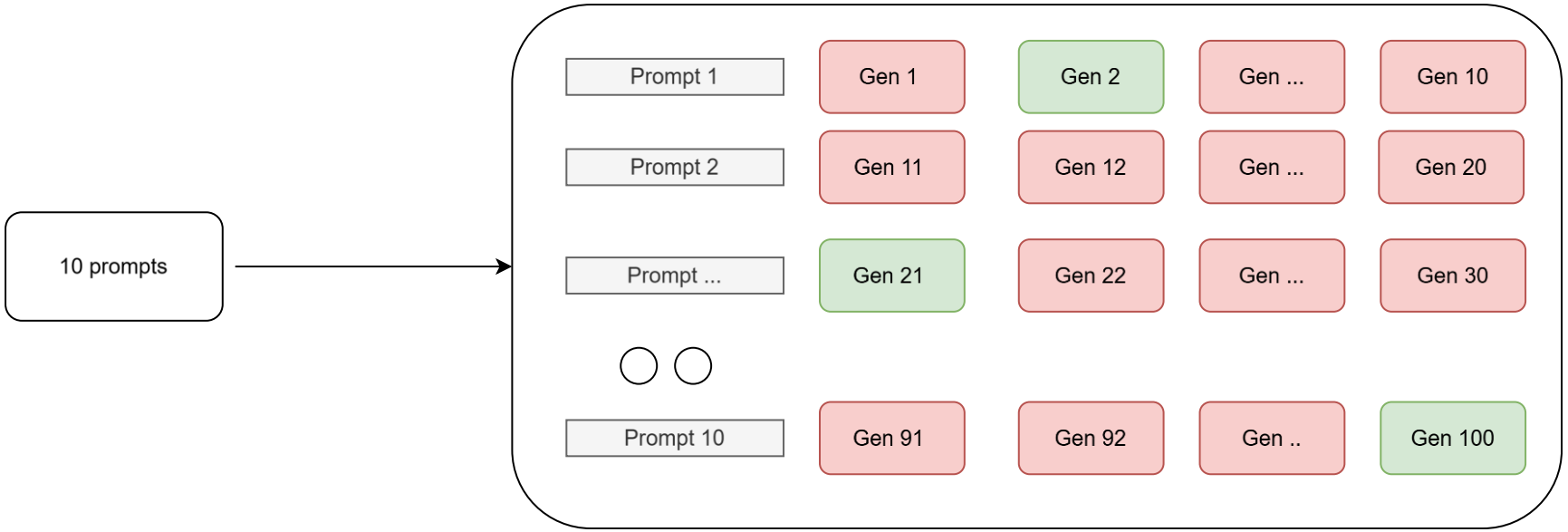
Design:
The entire body and head are solid blue.
The beak is solid yellow.
Add a simple red superhero mask around the eyes that wraps around the head.
Paint a large yellow lightning bolt symbol on the side of the chest.





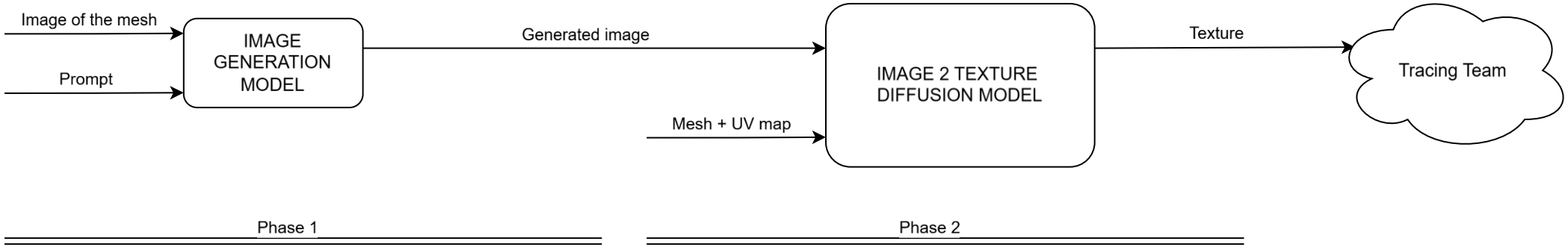
Criterion	Result	Comment
Base shape preservation	Good / Medium / Poor	Does the generated image respect the original duck silhouette and proportions?
Prompt adherence	Good / Medium / Poor	Does the image follow the requested design instructions?
Color limit respected	Yes / No	Are only the allowed colors used?
Solid colors only	Yes / No	Are the colored areas flat, without gradients or texture effects?
Surface paint plausibility	Good / Medium / Poor	Do the graphics look painted on the surface rather than added as accessories or extra geometry?
Image quality	Good / Medium / Poor	Is the render visually clean ?
Suitable for Phase 2	Yes / No	usable as input for the next phase?

Phase 1 - Conclusion



- With on premise solution, 1 image generation takes up to 3 min
- With less than 20% success rate, it can take up to 15 min to generate 1 acceptable image.

Image 2 Texture pipeline



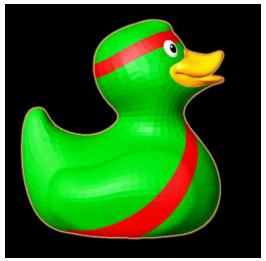
Black Forest Labs

FLUX KONTEXT 1
~32 GB VRAM

stability.ai

SDXL
~32 GB VRAM

Phase 2: Flux generated image are not reliable



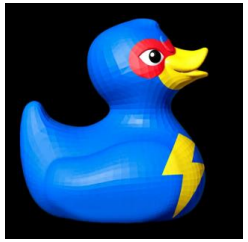
- Background leakage

- More complex pattern, more severe leakage

Phase 2: Single image input creates inconsistencies

Multiview

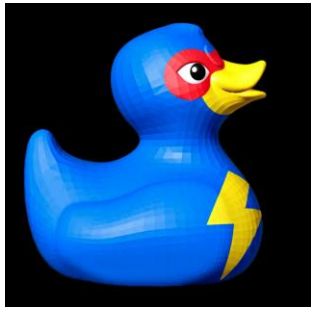
3D Render



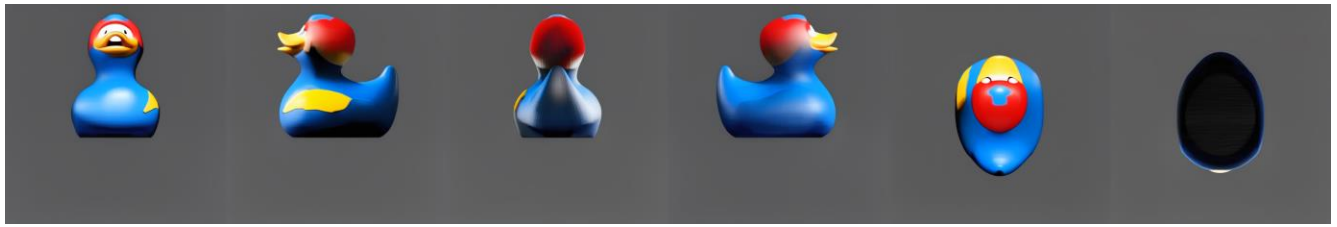
- 1 image as input forces the model

to infer unseen surfaces, leading to inconsistencies across views.

Phase 2: Benchmark

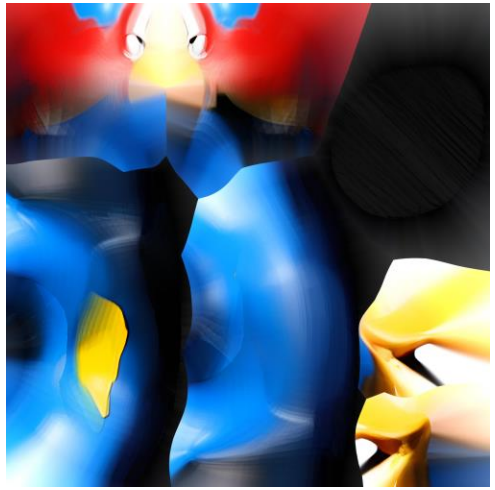


Above average flux generation

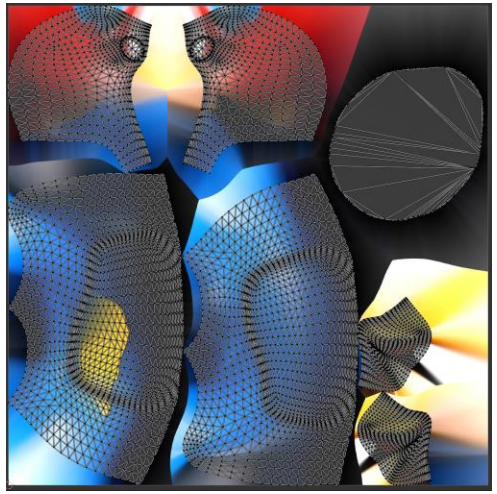


Multiview

3D Render



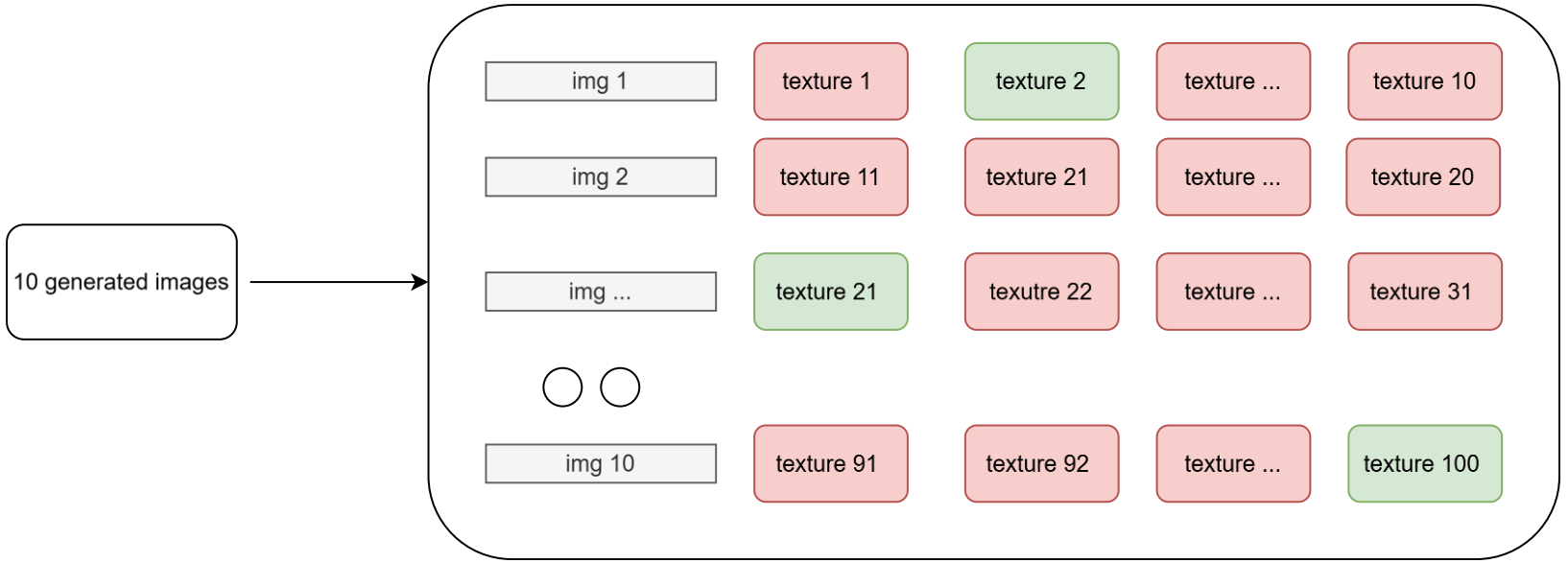
Texture



UV map over texture

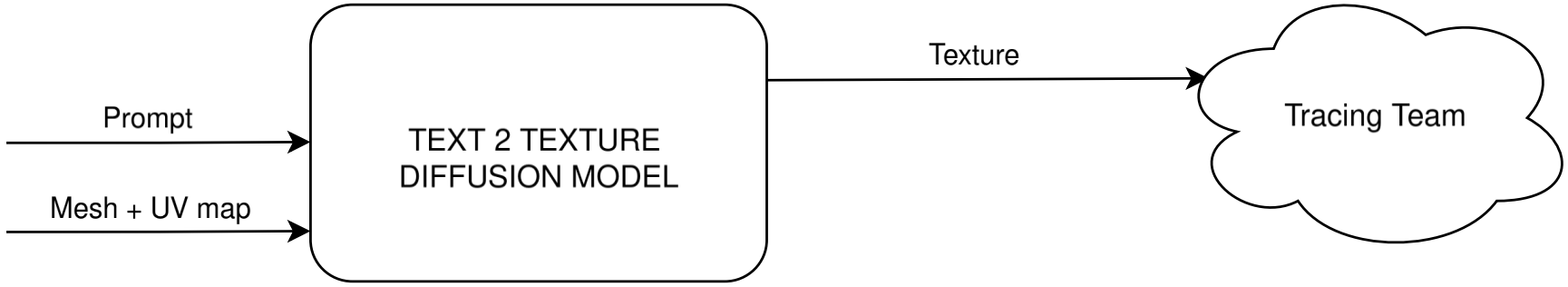
Criterion	Result	Evaluation Question
Texture alignment with mesh	Good / Medium / Poor	Do the graphical elements follow the curvature and regions of the mesh correctly?
UV seam artifacts	Low / Medium / High	Are visible seams or breaks appearing where UV islands meet?
Texture distortion	Low / Medium / High	Are shapes stretched or compressed due to UV mapping or projection?
Multiview consistency	Good / Medium / Poor	Does the design remain coherent when the model is viewed from different angles?
Background leakage	Yes / No	Are background colors or unwanted artifacts transferred onto the mesh surface?
Graphic element placement	Good / Medium / Poor	Are the design elements (e.g., stripes, bands) positioned correctly on the duck geometry?
Manual painting feasibility	Good / Medium / Poor	Would the generated design still be easy to reproduce with manual painting?
Suitable for tracing	Yes / No	Usable as input for team tracing ?

Phase 2: Conclusion



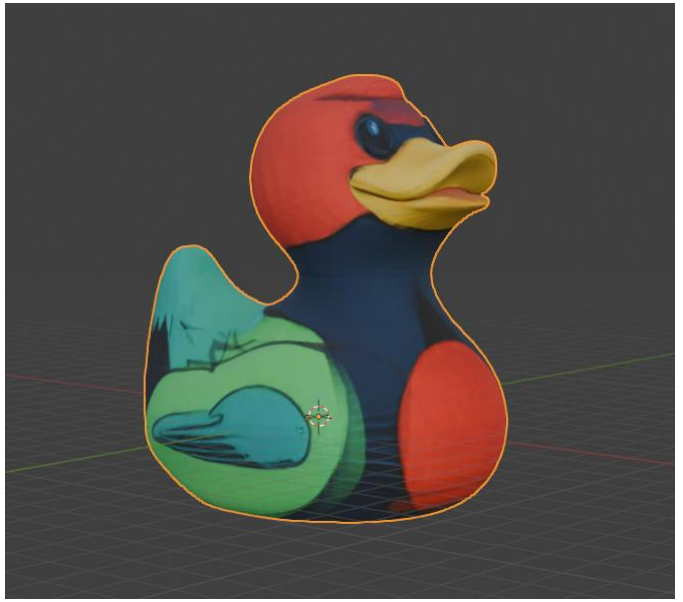
- Generated 100 texture from 10 Flux generated input images
- Success rate of 10%, this makes the method not suitable.

TEXT2Texture - pipeline

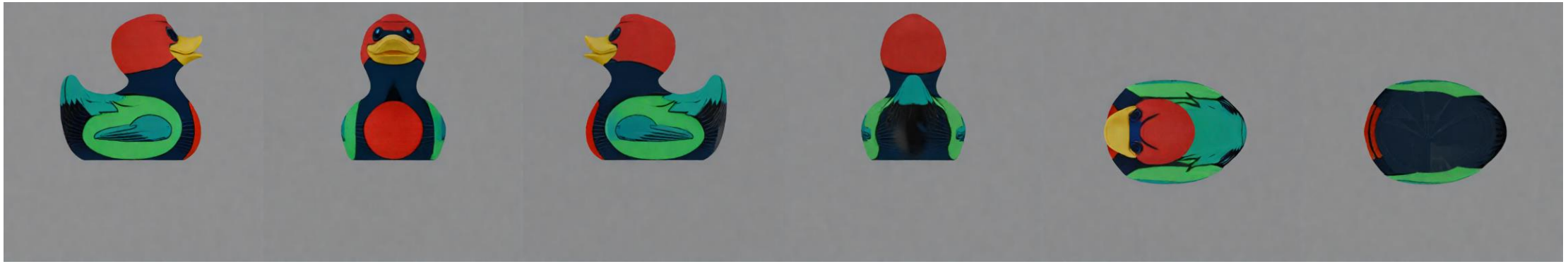


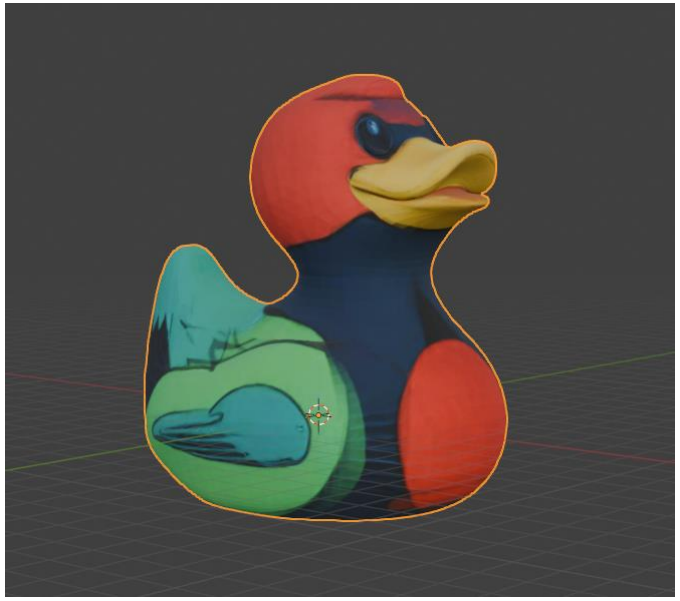
stability.ai

TEXT2Texture – last week issue



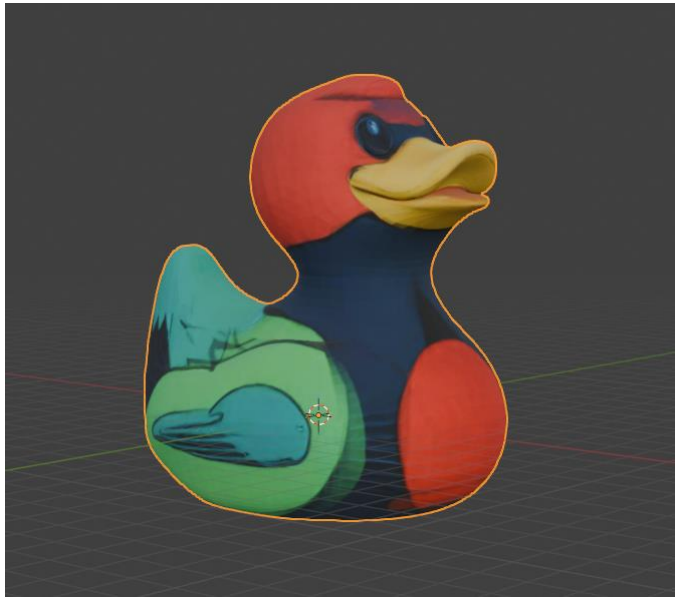
TEXT2Texture - résultats & benchmark





PROMPT="Cartoon superhero duck toy, red mask and blue wings , choose only one of these colors : red, blue, black, green."

NEG_PROMPT="fully painted background, gradients, High-gloss, realistic shading, 3D shadows, thin lines,small details tiny patterns, messy edges, photorealism, body horror"



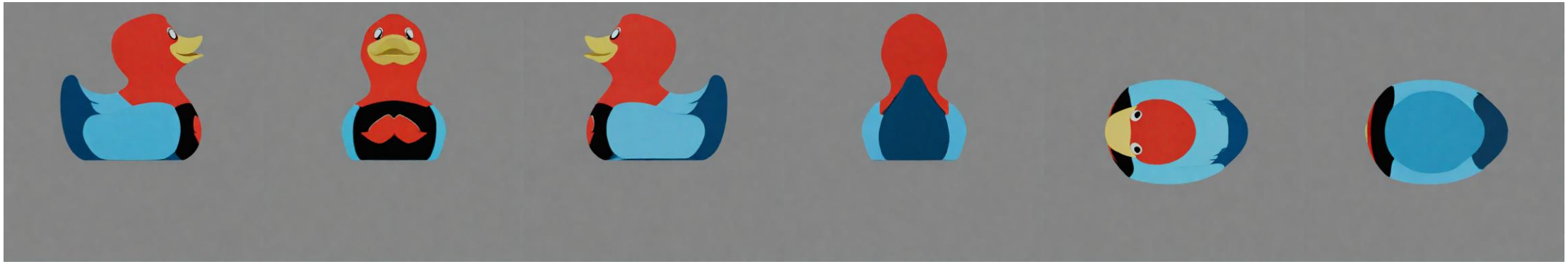
PROMPT="Cartoon superhero duck toy, red mask and blue wings , choose only one of these colors : red, blue, black, green."

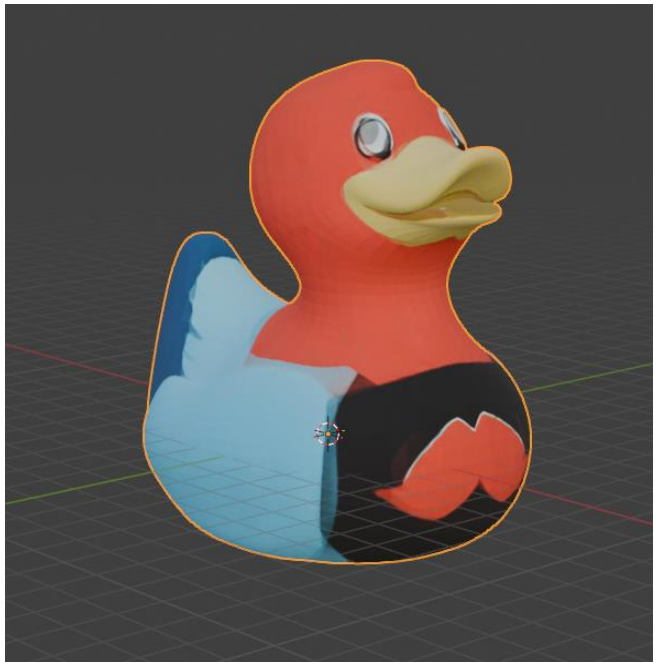
NEG_PROMPT="fully painted background, gradients, High-gloss, realistic shading, 3D shadows, thin lines,small details tiny patterns, messy edges, photorealism, body horror"

- Red mask ☑
- Blue wings ☑
- Red | blue | black | green ☑

Criterion	Result	Evaluation Question
Texture alignment with mesh	Good / Medium / Poor	Do the graphical elements follow the curvature and regions of the mesh correctly?
UV seam artifacts	Low / Medium / High	Are visible seams or breaks appearing where UV islands meet?
Texture distortion	Low / Medium / High	Are shapes stretched or compressed due to UV mapping or projection?
Multiview consistency	Good / Medium / Poor	Does the design remain coherent when the model is viewed from different angles?
Background leakage	Yes / No	Are background colors or unwanted artifacts transferred onto the mesh surface?
Graphic element placement	Good / Medium / Poor	Are the design elements (e.g., stripes, bands) positioned correctly on the duck geometry?
Manual painting feasibility	Good / Medium / Poor	Would the generated design still be easy to reproduce with manual painting?
Suitable for tracing	Yes / No	Usable as input for team tracing ?

TEXT2Texture - résultat





"prompt": "Cartoon superhero duck toy, solid red mask, solid blue wings. Minimalist vector art, flat solid colors, unshaded, clean edges. Use strictly only these colors: red, blue, black, green.",

negative_prompt: "realistic, gradients, 3D shading, glossy, background, complex patterns, thin lines, messy, yellow, purple, orange, pink, gray, brown, ugly, deformed, blurry, bad anatomy",

"prompt": "Cartoon superhero duck toy, solid red mask, solid blue wings. Minimalist vector art, flat solid colors, unshaded, clean edges. Use strictly only these colors: red, blue, black, green.",

negative_prompt: "realistic, gradients, 3D shading, glossy, background, complex patterns, thin lines, messy, yellow, purple, orange, pink, gray, brown, ugly, deformed, blurry, bad anatomy",

- Red mask ☑
- Blue wings ☑
- Red | blue | black | green ☑

Criterion	Result	Evaluation Question
Texture alignment with mesh	Good / Medium / Poor	Do the graphical elements follow the curvature and regions of the mesh correctly?
UV seam artifacts	Low / Medium / High	Are visible seams or breaks appearing where UV islands meet?
Texture distortion	Low / Medium / High	Are shapes stretched or compressed due to UV mapping or projection?
Multiview consistency	Good / Medium / Poor	Does the design remain coherent when the model is viewed from different angles?
Background leakage	Yes / No	Are background colors or unwanted artifacts transferred onto the mesh surface?
Graphic element placement	Good / Medium / Poor	Are the design elements (e.g., stripes, bands) positioned correctly on the duck geometry?
Manual painting feasibility	Good / Medium / Poor	Would the generated design still be easy to reproduce with manual painting?
Suitable for tracing	Yes / No	Usable as input for team tracing ?

TEXT2Texture – variances des résultats

"prompt": "Cartoon superhero duck toy, solid red mask, solid blue wings. Minimalist vector art, flat solid colors, unshaded, clean edges. Use strictly only these colors: red, blue, black, green.",

"negative_prompt": "realistic, gradients, 3D shading, glossy, background, complex patterns, thin lines, messy, yellow, purple, orange, pink, gray, brown, ugly, deformed, blurry, bad anatomy",

Générés en 5m15s

GenAI Choice - resume

- I2T
  - Background leakage
  - FLUX Image generation inconsistency
  - More deterministic results
  - 15 minutes for good results
- T2T
  - Much less background leakage
  - More variance in results
  - 1m30 for good results
  - 45 seconds by generation

Tracing

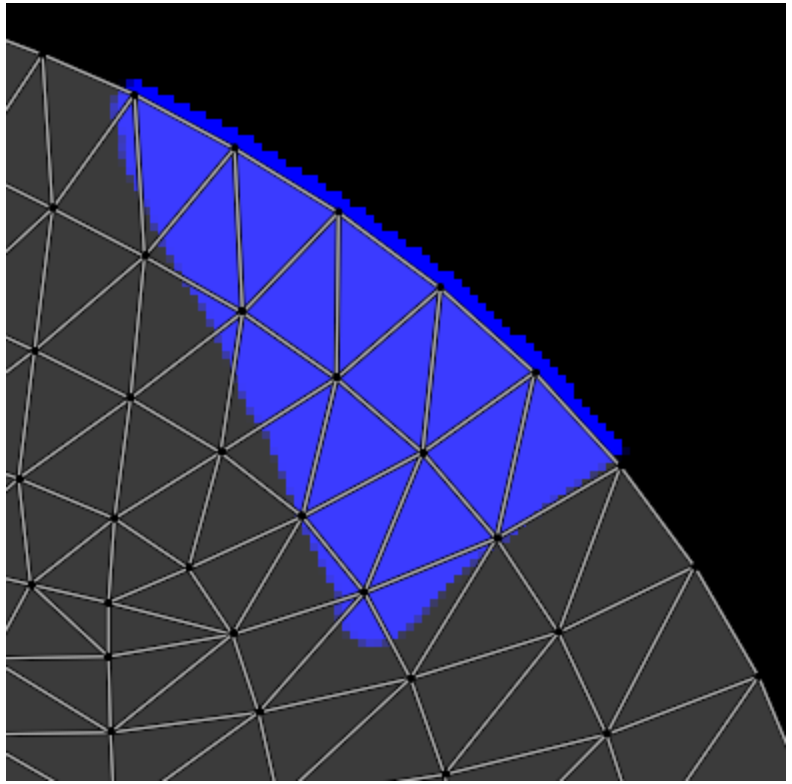
Identified issues in the pipeline

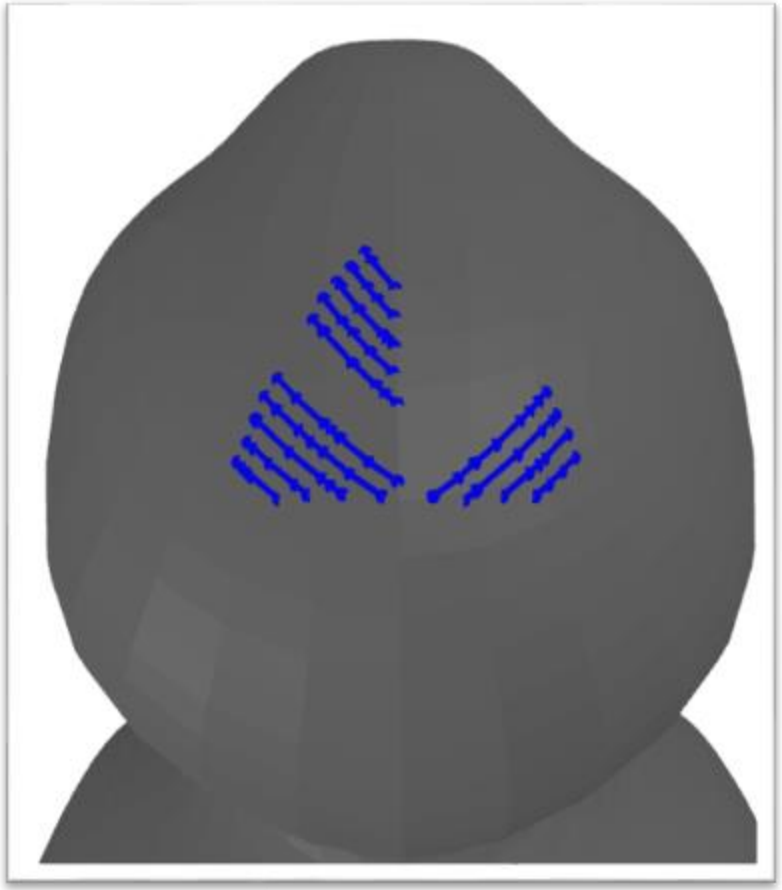
- "Out-of-bounds"
- Palettization
- Small artifacts
- UV seams
- Unreachable area
- Fill slicing failures

Tracing – Out-of-bounds

Identified issues in the pipeline

Due to the way the textures are generated, islands may extend out of the UV mapped area, rendering simple 3D projection impossible





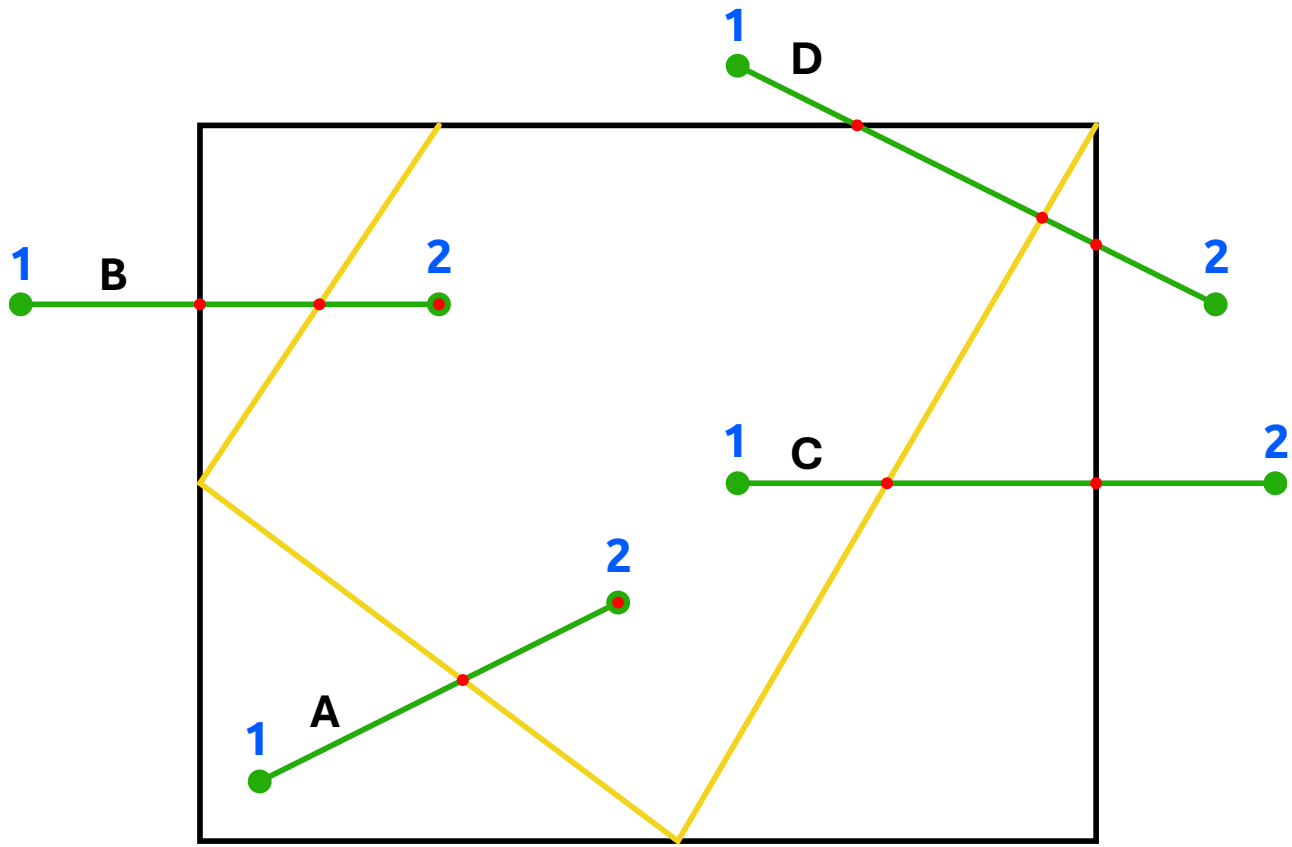
Example of "out-of-bounds" issue

Tracing – Out-of-bounds

Identified issues in the pipeline

Projecting a segment from UV (2D) to mesh (3D)

- **A: Inside → Inside**
  - Add edge intersections
  - Add projected point
- **B: Outside → Inside**
  - Add UV boundary
  - Add edge intersections
  - Add projected point
- **C: Inside → Outside**
  - Add edge intersections
  - Add UV boundary
  - Finish trace
- **D: Outside → Outside** (not implemented because rare case)
  - Easier with more generalized method, similar to fill slicing

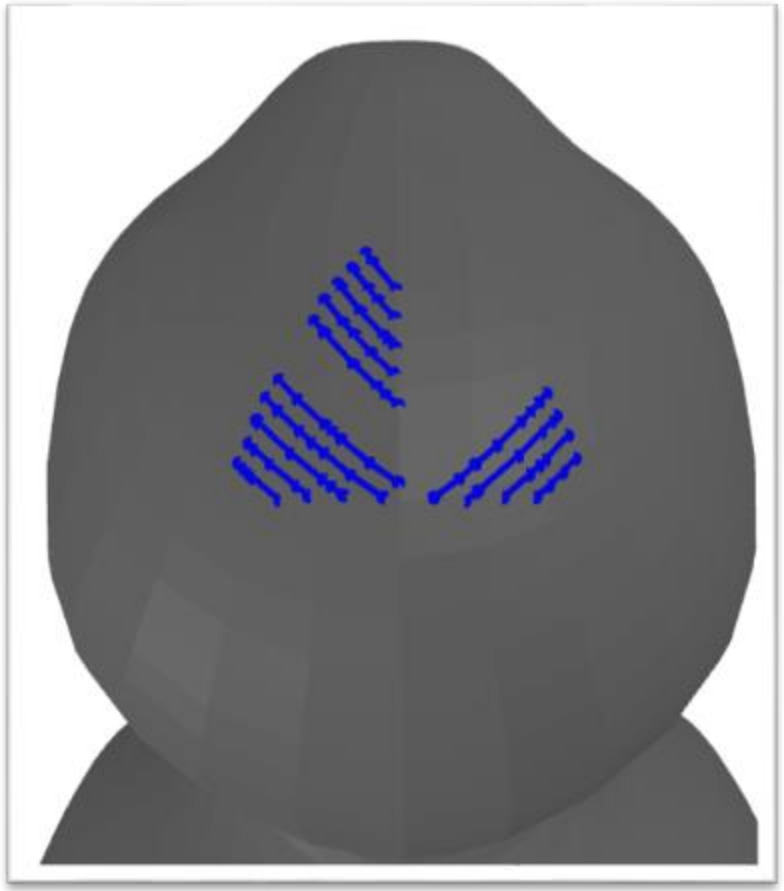


UV boundary crossing cases

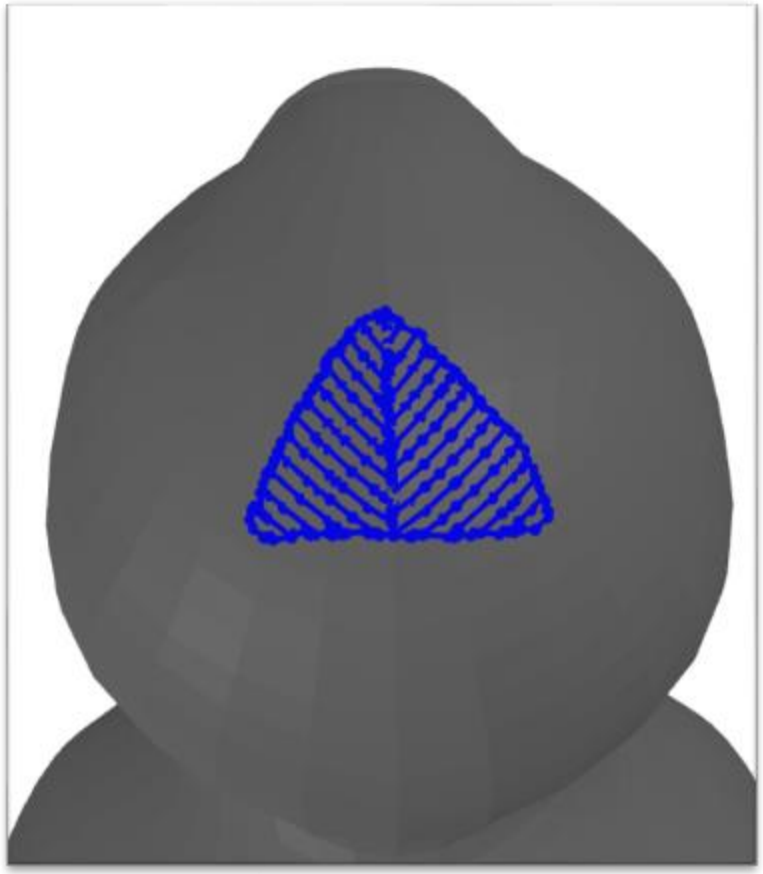
Tracing – Out-of-bounds

Fix implemented

"Out-of-bounds"



Before "out-of-bounds" fix

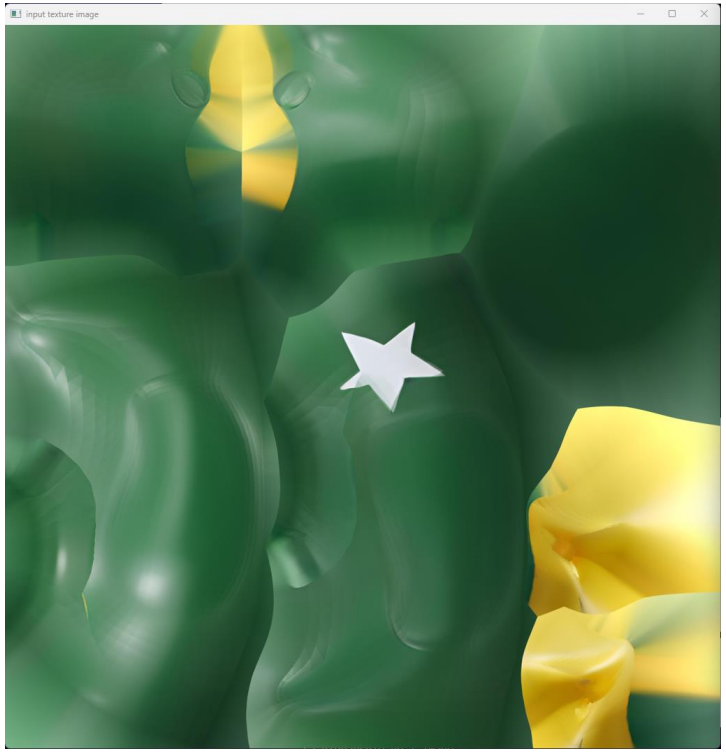


After "out-of-bounds" fix

Tracing - palettization

Identified issues in the pipeline

The default color quantization method (MEDIANCUT) is heavily biased by the primary color, which "floods" the other colors



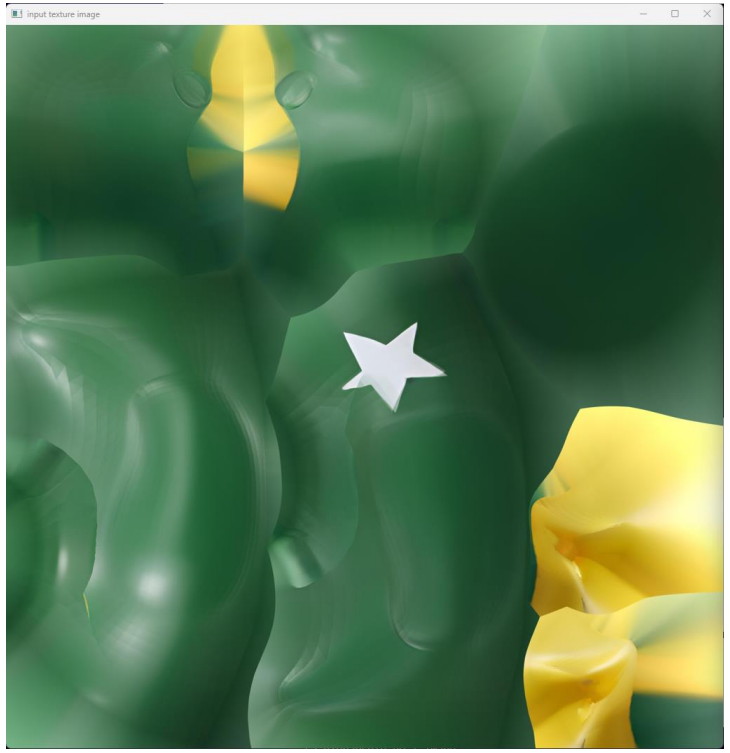
Input texture

Palettized texture

Tracing - palettization

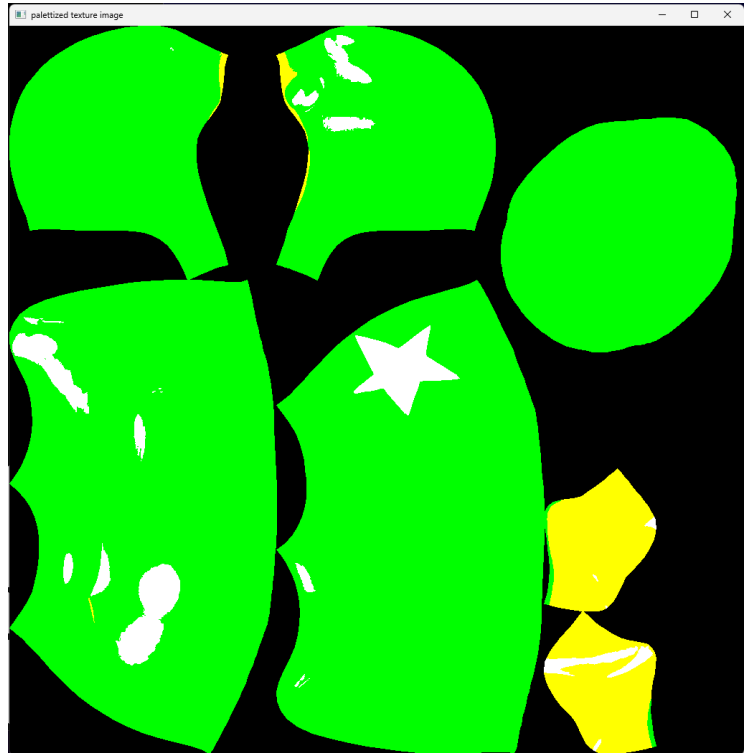
Fixes implemented : **how**

- replace quantize.MEDIANCUT with openCV KNN solution
- add a function to process only UV map related pixels
- add a Tracer class parameter to exclude a specific color from the palettization



Input texture

Palettized texture - "before"



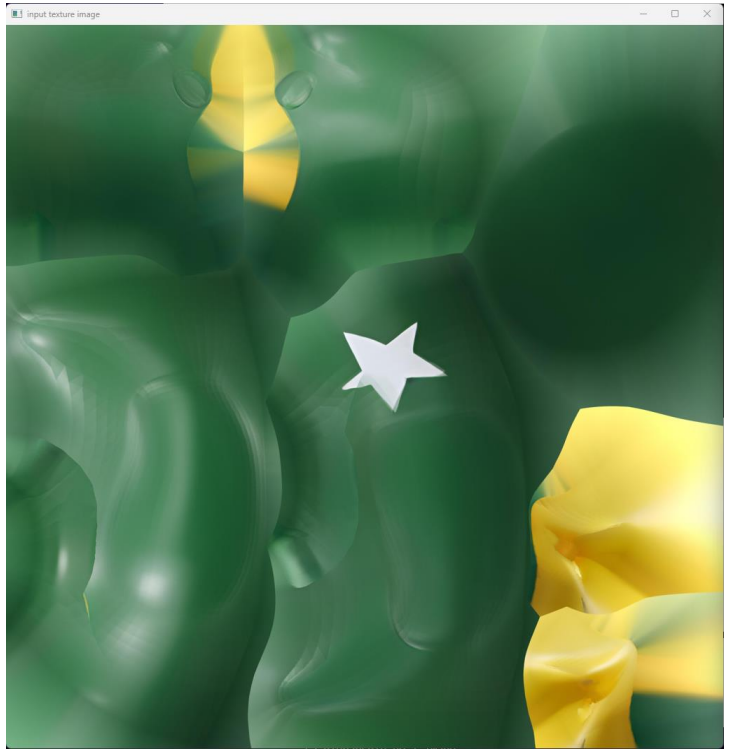
Palettized texture - "after"

palette : yellow, green, white / excl. color : black

Tracing - palettization

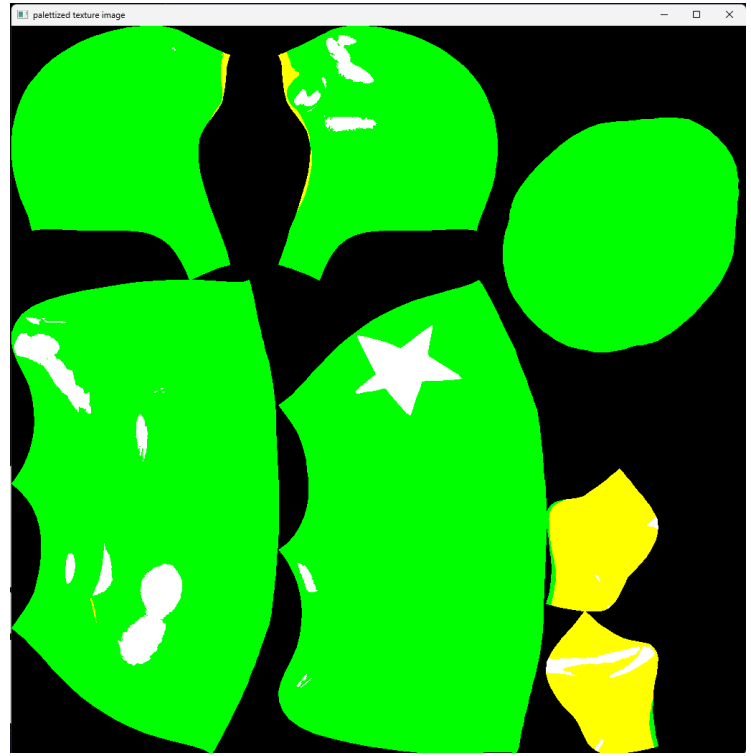
Fixes implemented : **Why**

- the UV Map masked reduce the computation effort of the palettization
- KNN approach is not biased by any color weight
- Excluding (a) color(s) ensure that unrelated colors (e.g. background after masking step) will not interfere



Input texture

Palettized texture - "before"

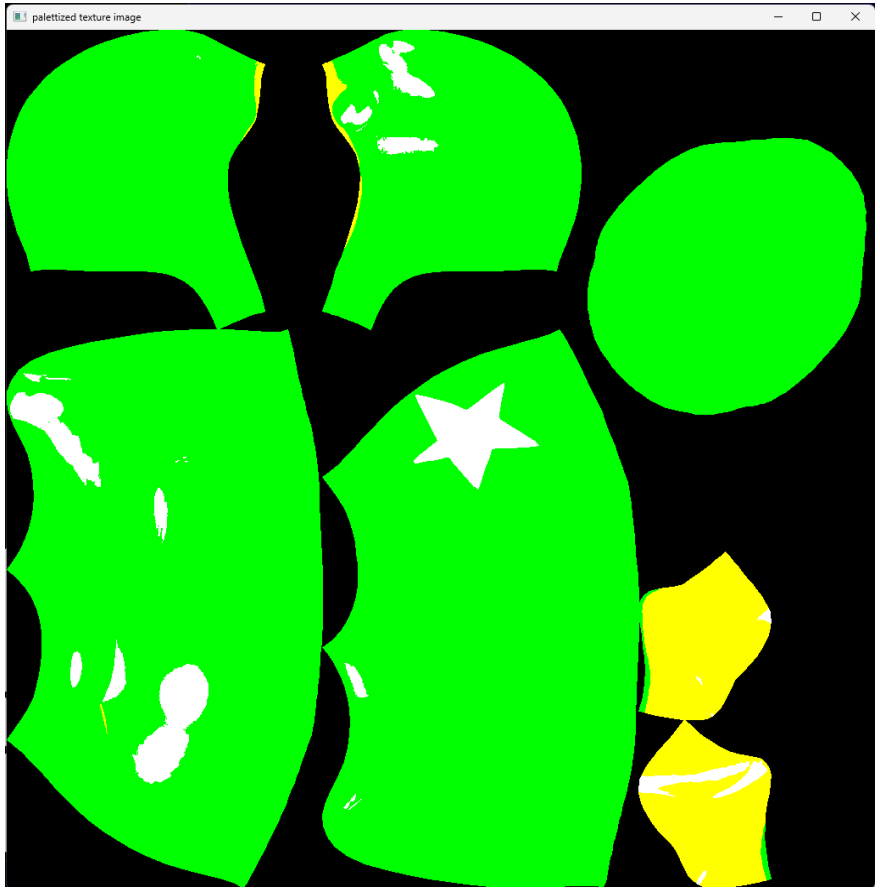


Palettized texture - "after"
palette : yellow, green, white / excl. color : black

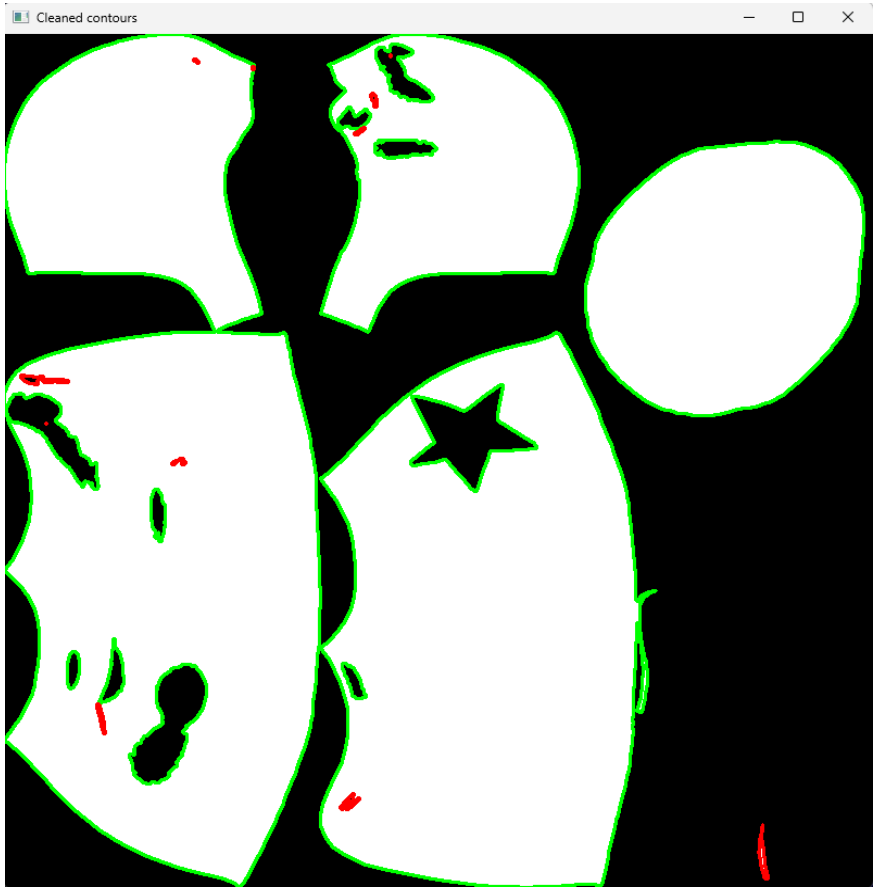
Tracing – small artifacts

Identified issues in the pipeline

When dealing with AI generated textures, some color artifacts may appear, producing very small islands that should otherwise be of the surrounding color



Palettized texture



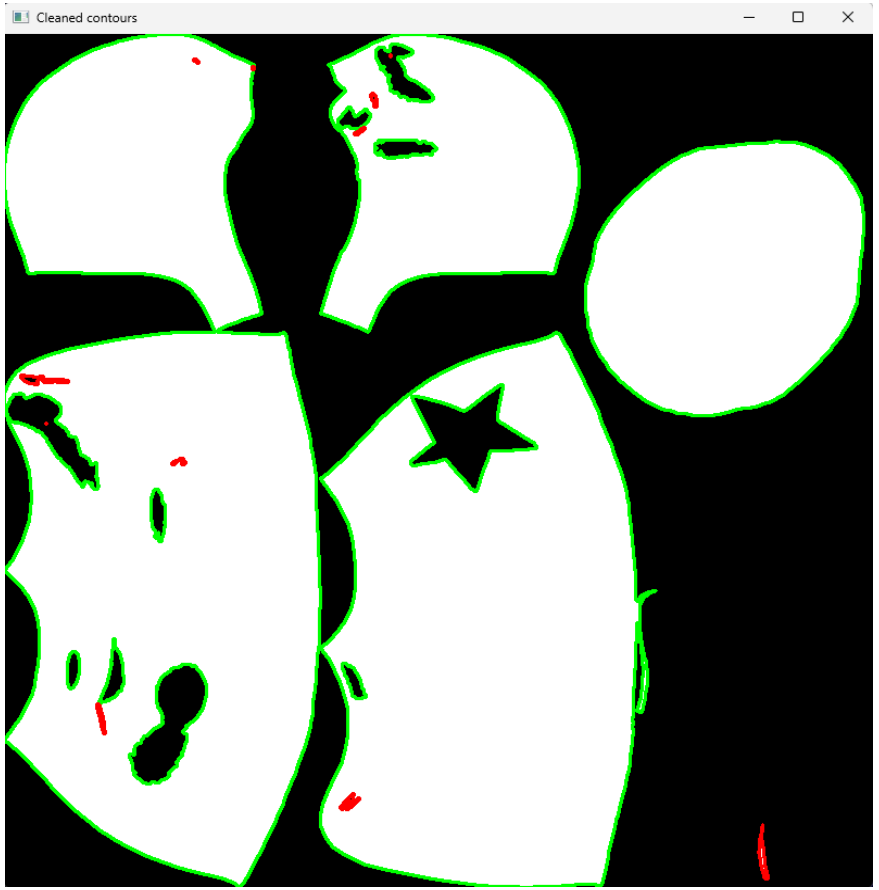
Smalls artefacts

Tracing – small artifacts

how to solve

When dealing with AI generated textures, some color artifacts may appear, producing very small islands that should otherwise be of the surrounding color

- Add a treshold for too small artefacts
- Use the main surrounding color to replace artefact's one.

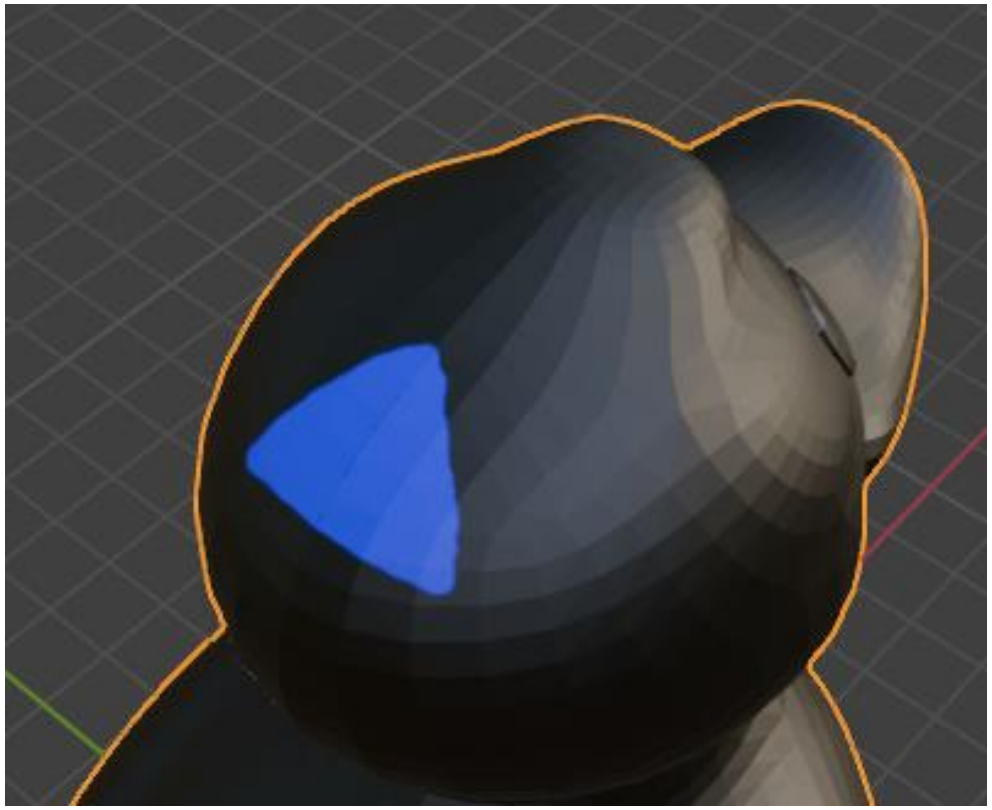


Smalls artefacts

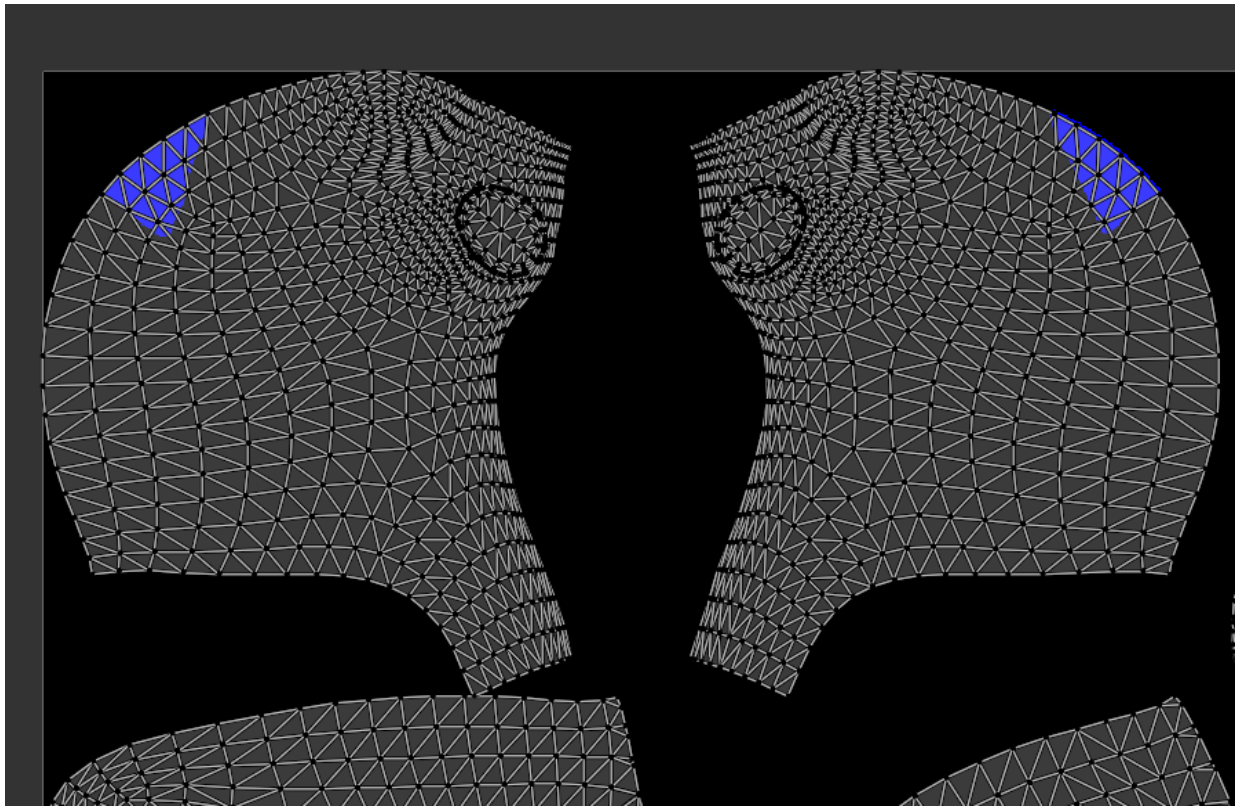
Tracing – Seams

Identified issues in the pipeline

When a particular pattern covers multiple UV islands (i.e. crosses a UV seam), multiple traces will be produced, splitting the shape at the seam



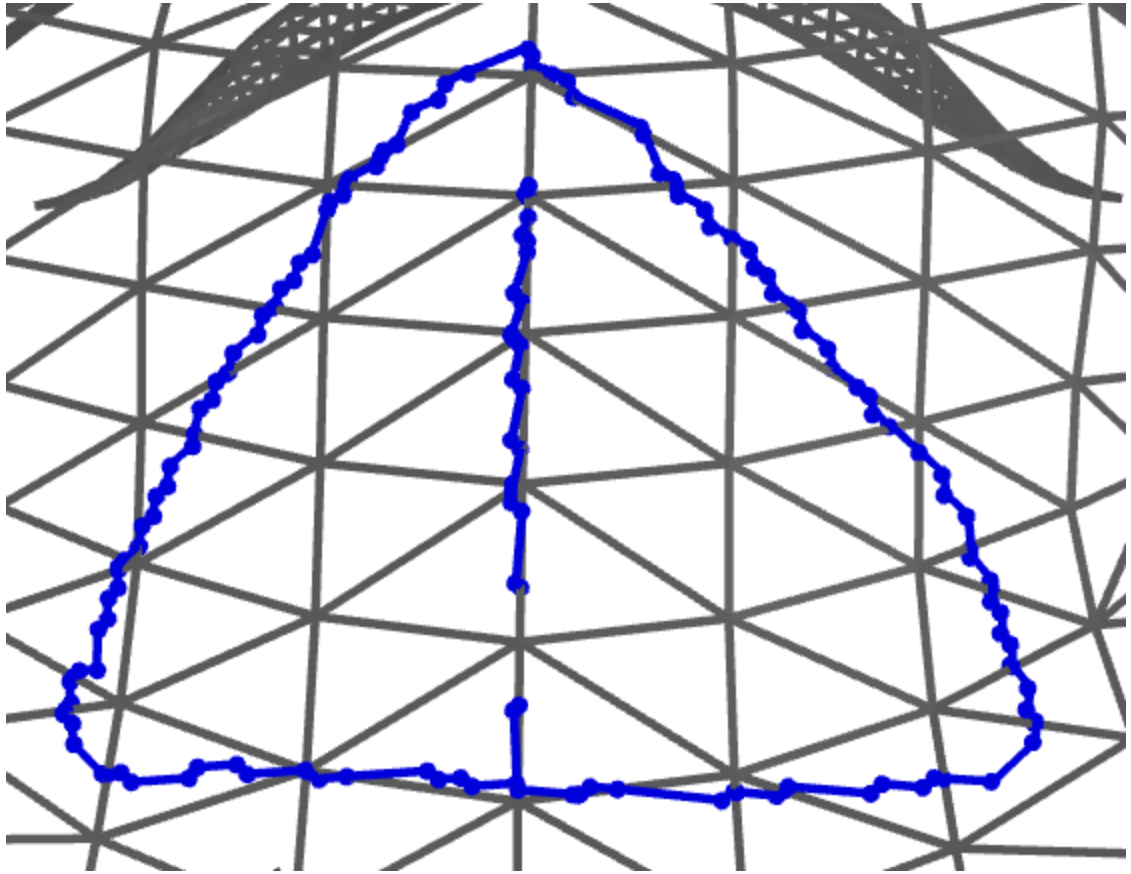
Single monolithic triangle texture



Triangle broken across UV seam

Tracing – Seams

Current result



Current result with trace on seam

Tracing

Identified issues in the pipeline

The path produced by the tracer does not take into account the reachability of the zones, and will produce undrawable traces (e.g. edges of the eyes)

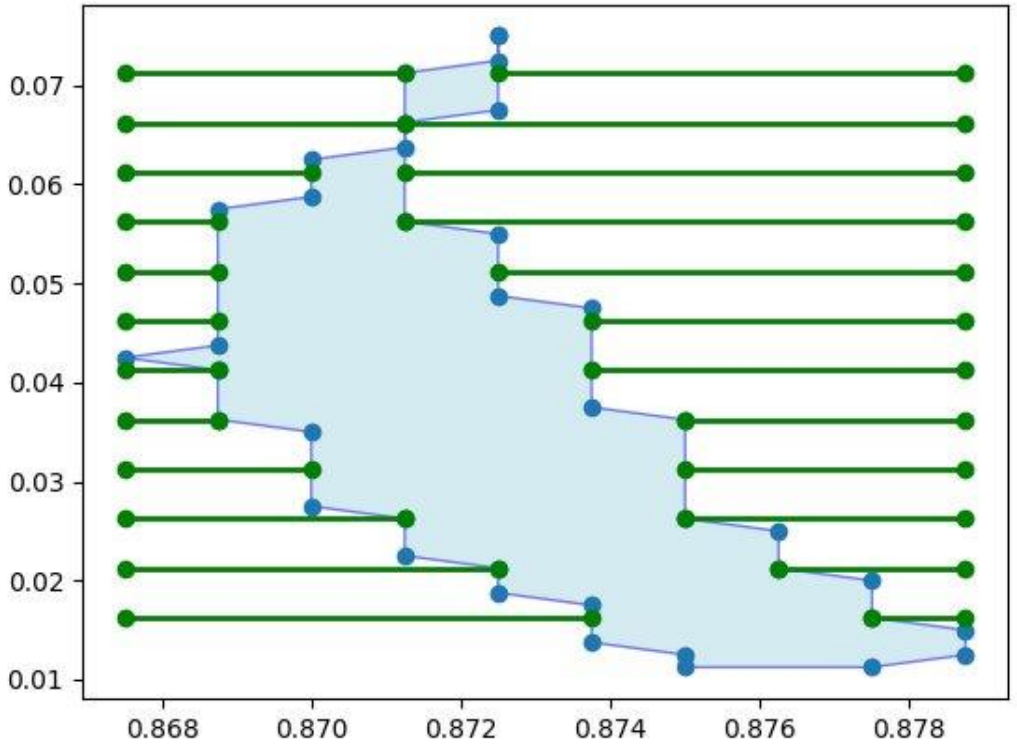
Steep angles / sharp edges on the eyes

Tracing – fill slicing

Identified issues in the pipeline

Fill slicing sometimes fail, inverting inside and outside areas of color islands

- Reasons of failures need to be defined
- Observed cases of failures : island where at some point the surface inside closed contours is null



"outer-filling" example

Tracing

Identified issues in the pipeline

Solved this week

- "Out-of-bounds"
- Palettization

For next week

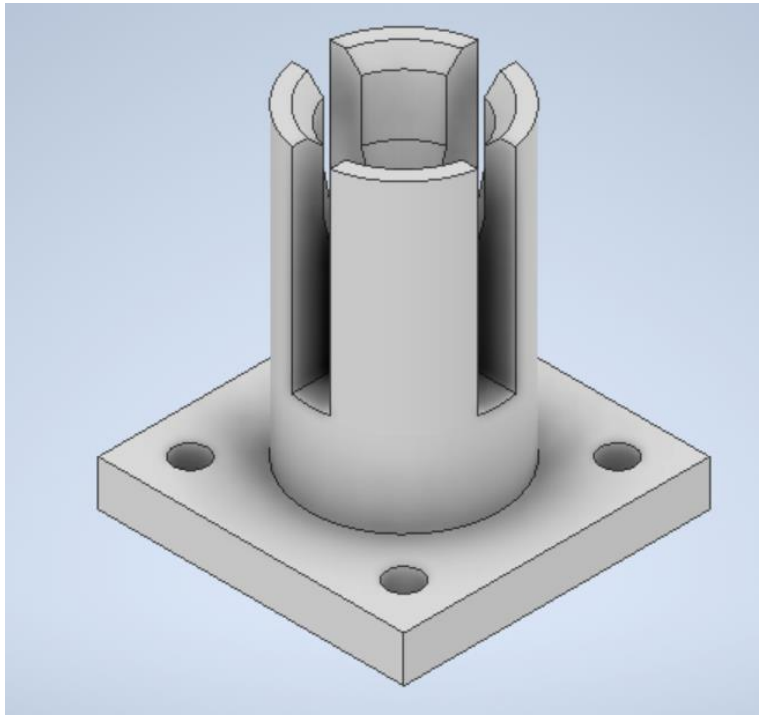
- Small artifacts
- UV seams
- Unreachable area
- Fill slicing failures

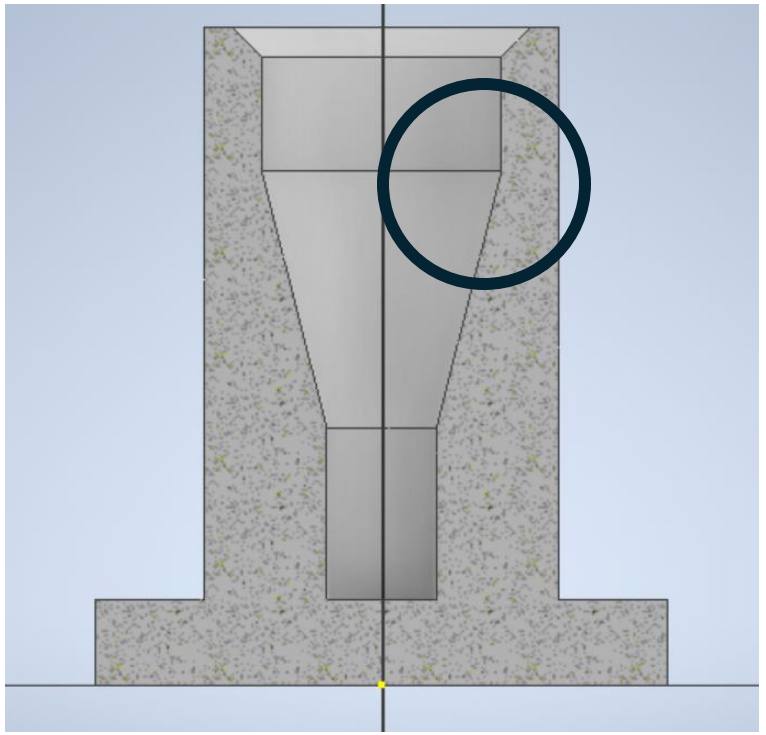
Impression 3D

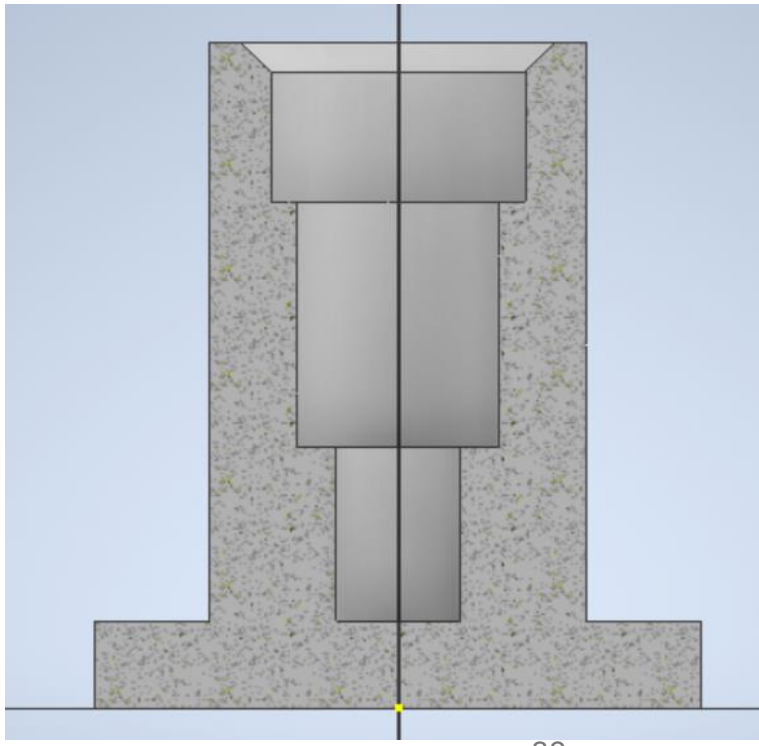
- Correct the stability of support (duck support)
  - Temporary solution (not a must have for now)

Impression 3D

- Objective: Remove the friction effect between pen and support
  - How: Try to remove contact surface
  - Result: Wrong idea, pens still get stuck







Robot

- Investigate a safety solution
- Can catch and manipulate tools (pens)

- Implement a safety solution
- Draw on a 3D surface (duck)
- Draw with 2 pens (change pens during drawing)

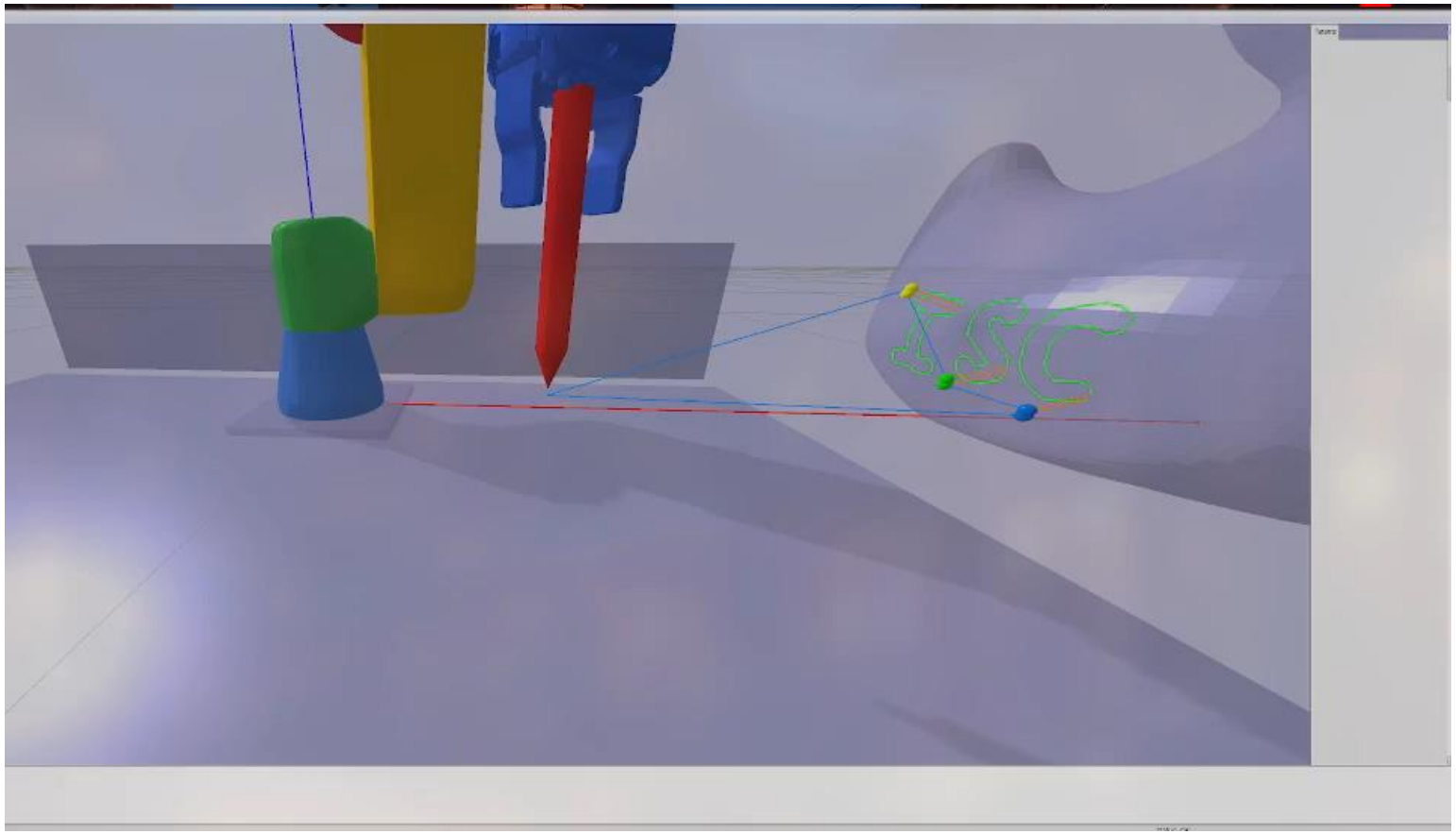
Robot

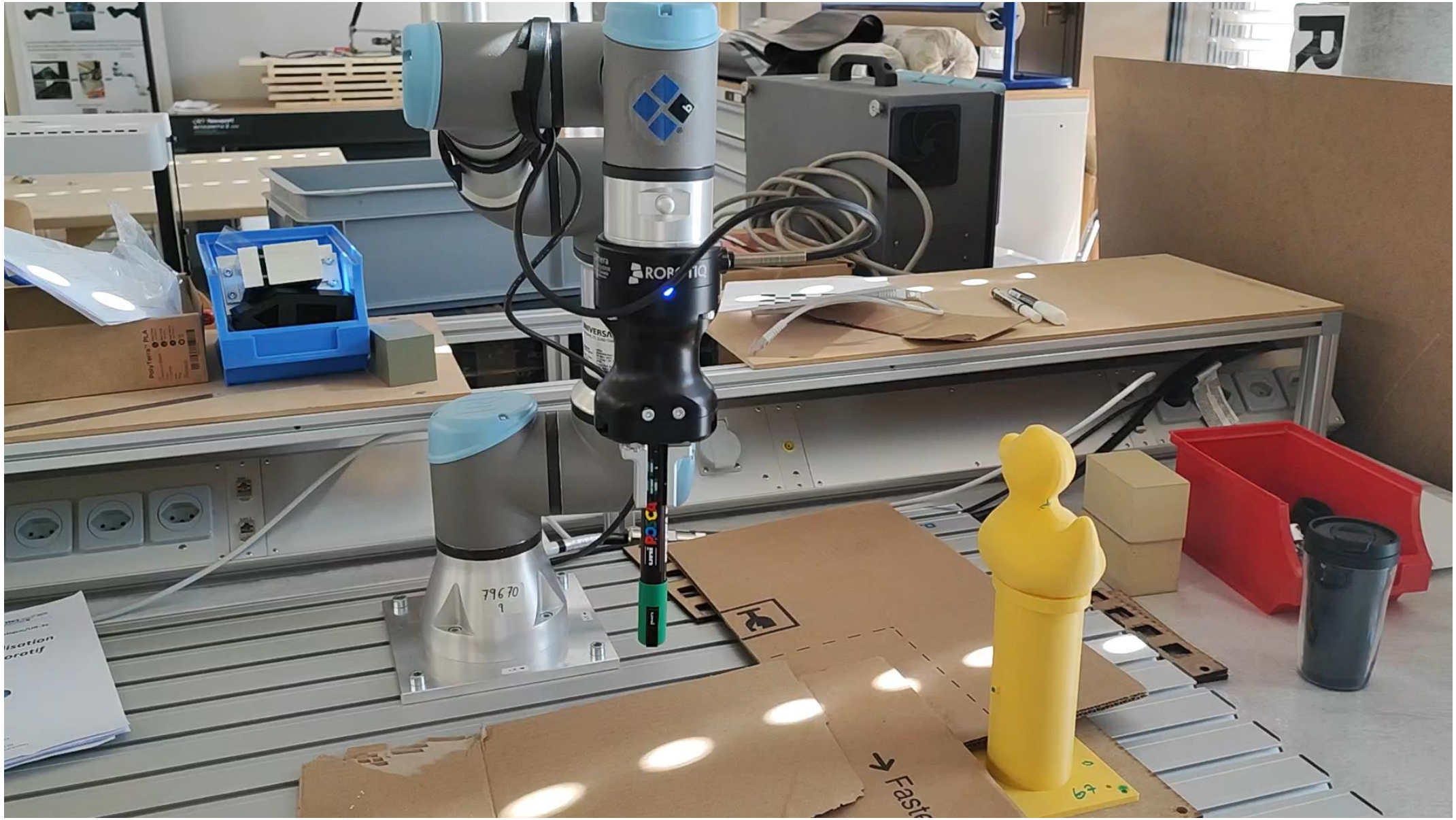
- Pipeline of pen transition complete
  - Need adjustment : Calibration

Robot

PyBullet

- URDF and STL import
- Collision detection (self, other)
- Basic pathing simulation

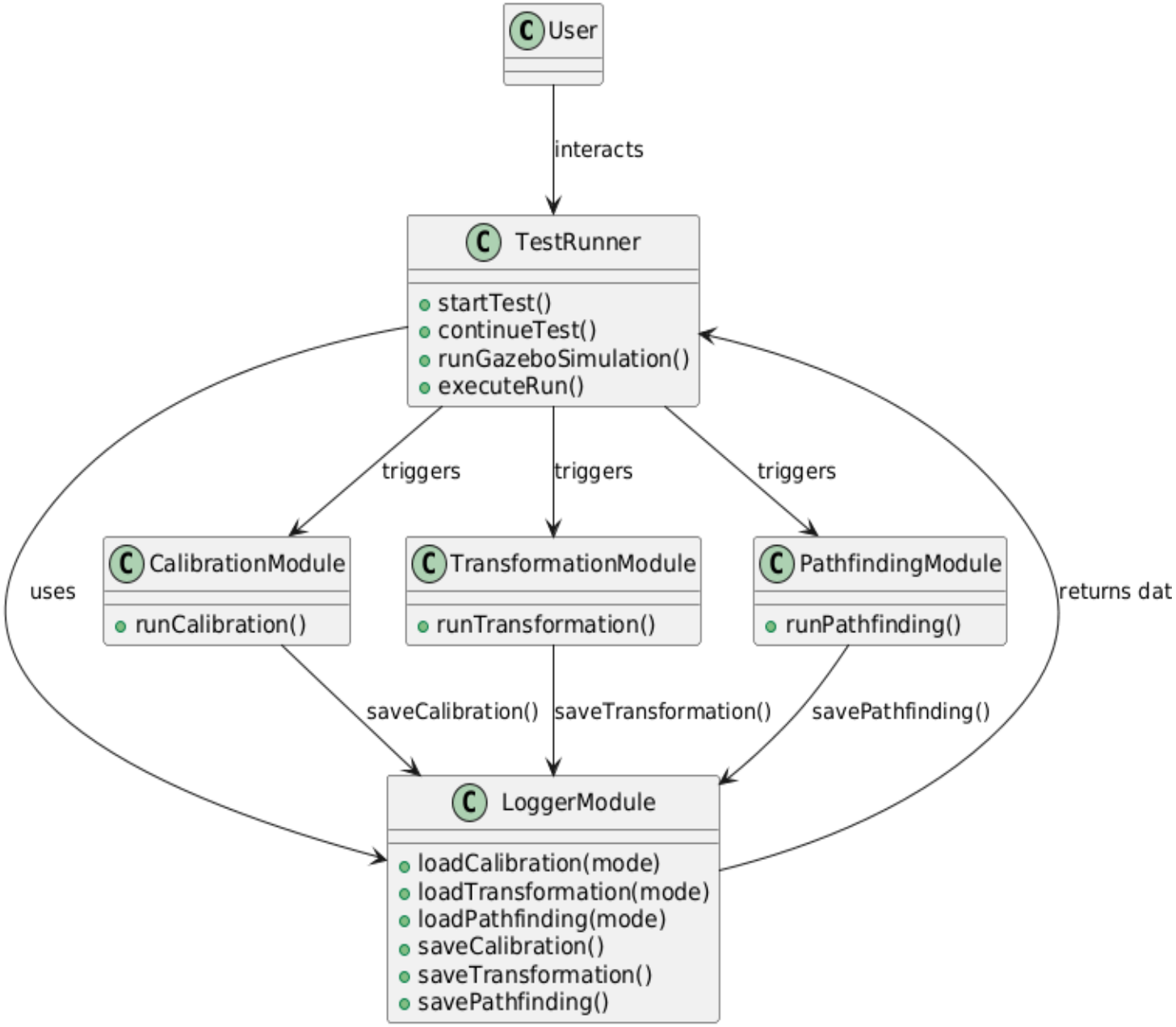



79670

Robot

Test architecture

- Complete redesign with modules for more flexibility



Robot: work

- Pen transition
- PyBullet
  - Collision detection
- Path generation
  - Follow data from tracing team
  - Joints configuration
- Logger
  - Save/load data
  - Log forces

Robot: issues

- Assembling pipeline
  - Intern communication/organisation
  - Code separations
- Transformation object to robot coordinate
  - Object calibration ≠ points
  - Objects model in simulation
- Simulation vs Reality
  - Object position
  - Time and frequency
  - In-run errors

Robot

- **Milestones planned for week 5:**
  - Clean code and pipeline
  - Correction safety PyBullet
  - Draw on a 3D surface (duck)
  - Draw with 2 pens (change pens during drawing)

Milestone

About the week 4

Using the **complete pipeline**, draw **AI generated contours** with a **selected GenAI solution**, on **a duck**, installed on a **stable support**, using **multiple colors**, while changing **colors automatically**

- Complete pipeline :
  - GenAI : priorization of model selection
  - Robot : not all parts are together (e.g. switching color)
  - Tracing : AI generated contours
  - Complete pipeline not integrated
- AI generated contours :
  - GenAi : chosen solution not integrated
- Draw on a duck :
  - Robot : axis alignement mismatch and safety issues
- Multiple colors :
  - Robot : solution not integrated

Milestone

For the week 5

Using the **complete pipeline**, starting from an **orchestration interface**, draw **AI-generated contours**, on a duck using **multiple colors with automatical changes.**