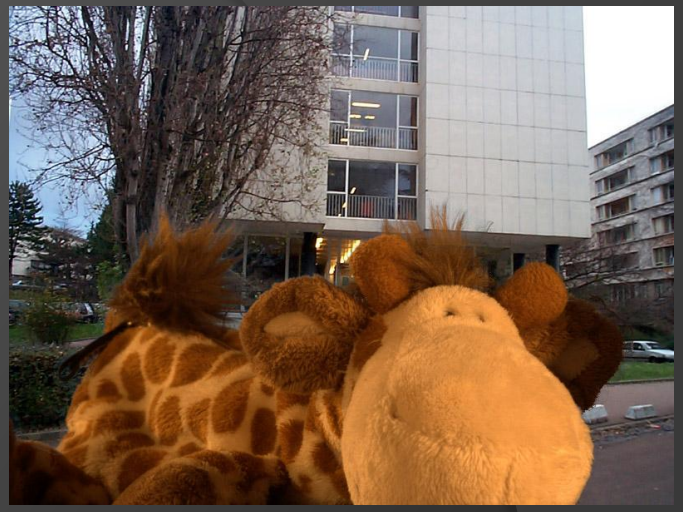


Henri Rebecq – 25th March 2014

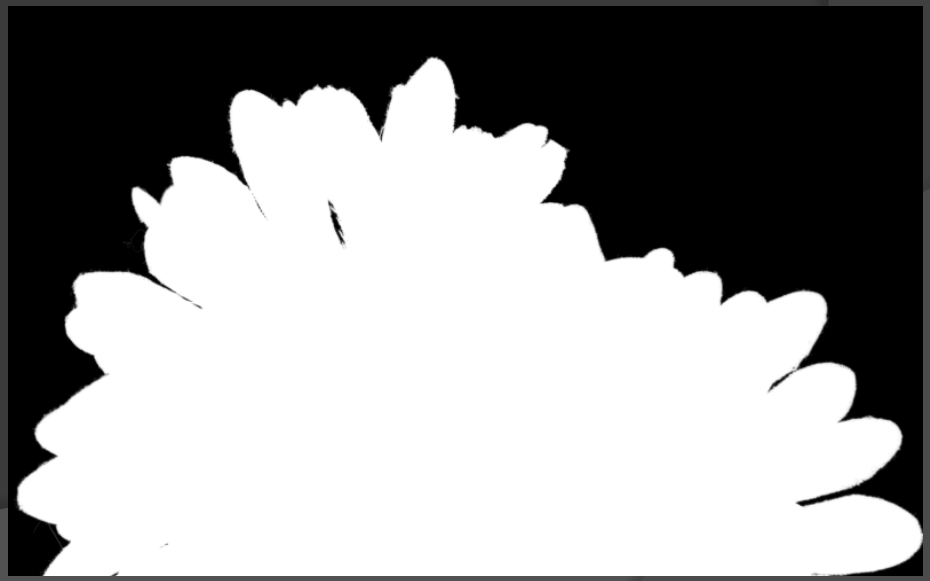
IMPROVING IMAGE MATTING USING COMPREHENSIVE SAMPLE SETS

Plan

- What is image matting?
- Description of the studied algorithm
- Implementation details & Results

What is image matting?

- Accurate foreground/background estimation in images
- Determining both full and partial pixel coverage (alpha-channel)



What is image matting?

- Observed image = convex combination of foreground and background

$$I_z = \alpha_z F_z + (1 - \alpha_z) B_z$$

$$\alpha \in [0; 1]$$

- 7 unknowns
- 3 equations

} Highly underconstrained problem

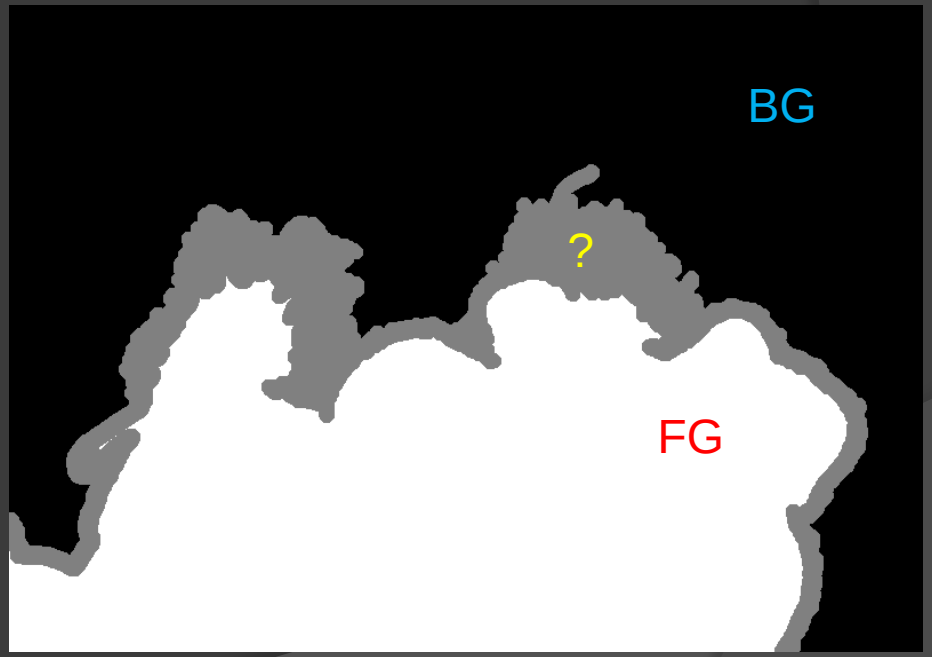
Main application

+

=

What is image matting?

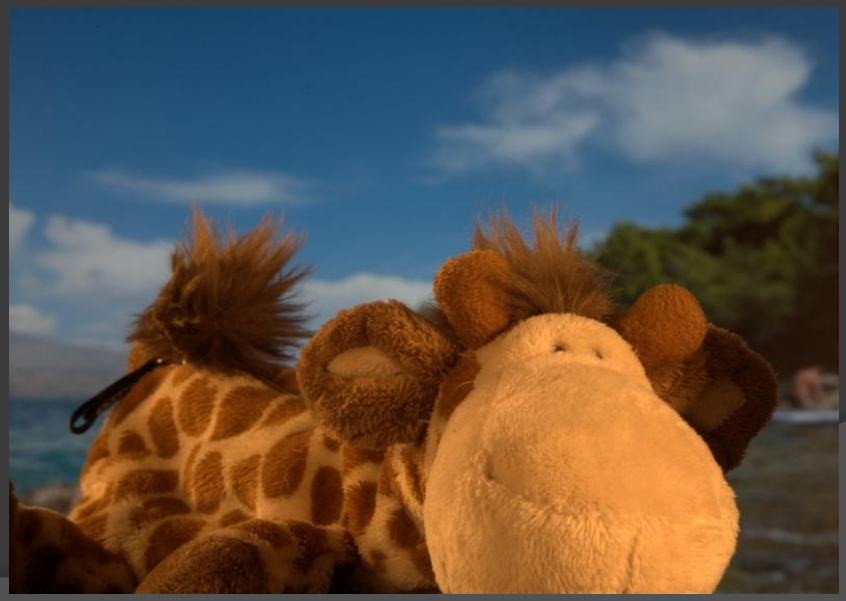
- How to reduce complexity?
  - User guidance



Trimap

What is image matting?

- How to reduce complexity?
  - Assumptions on image statistics
    - Smooth zones
    - Correlation between nearby pixels…



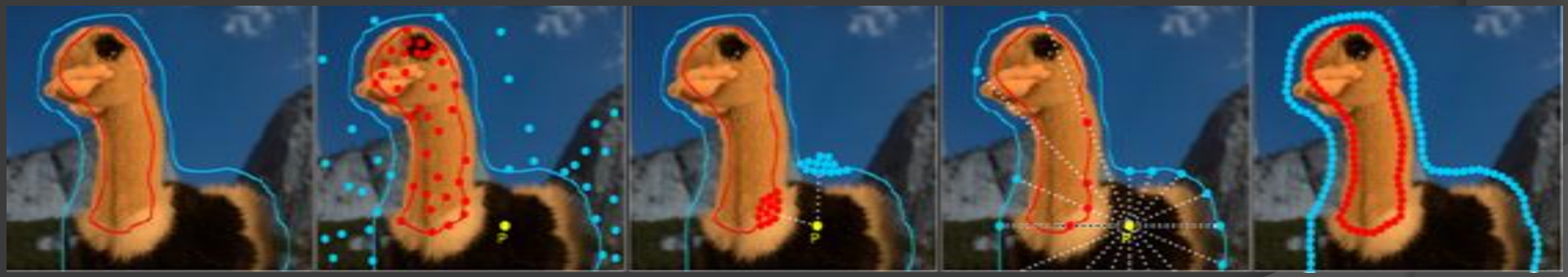
Color sampling method

- for each unknown pixel z
  - sample nearby known F_z and B_z colors
  - estimate matte using compositing equation:

$$\alpha_z = \frac{(I_z - B_z).(F_z - B_z)}{||F_z - B_z||^2}$$

Color sampling method

- How to choose correctly (F_z,B_z)?
  - Sample multiple candidate points (sample set)
  - Choose best pair among all possible pairs

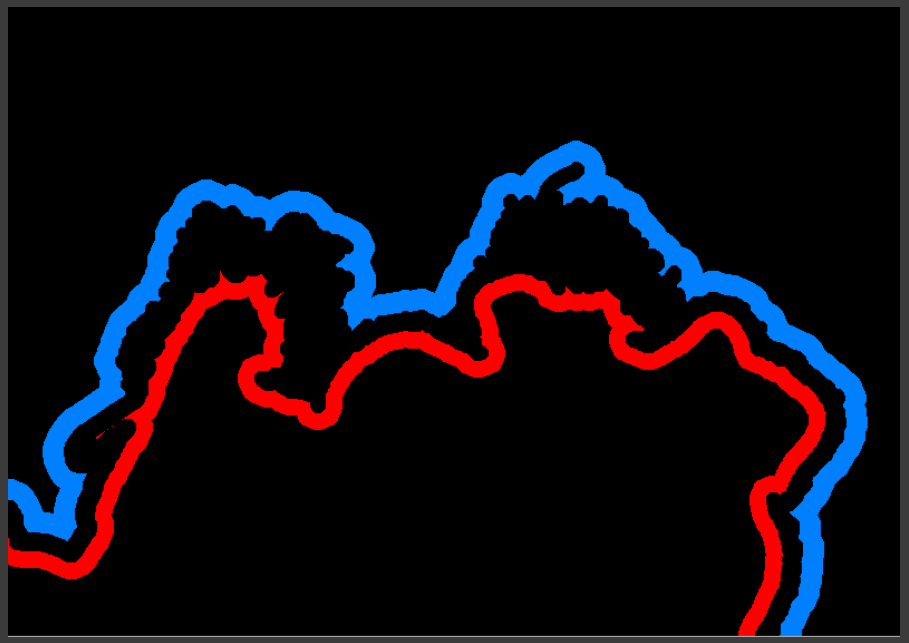
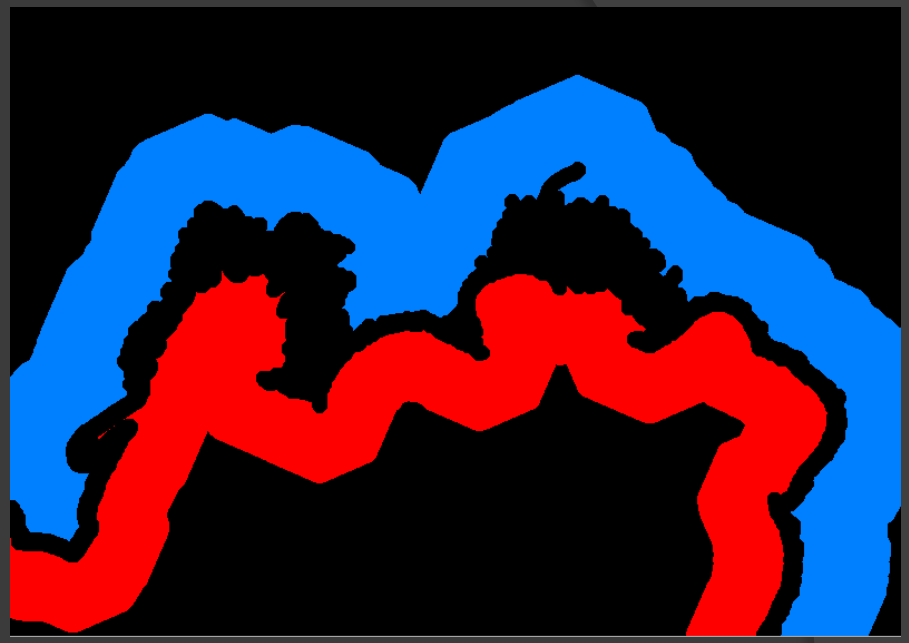
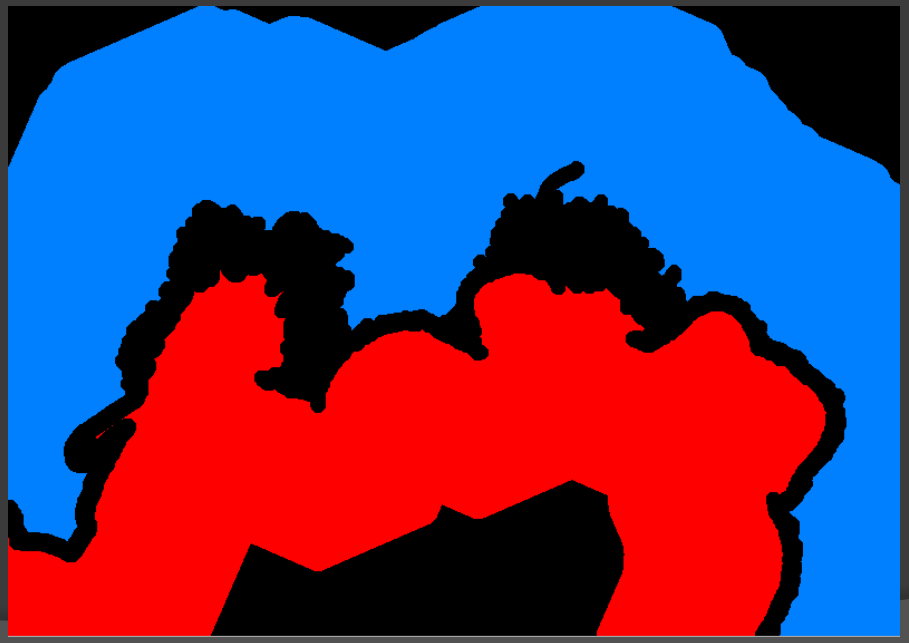
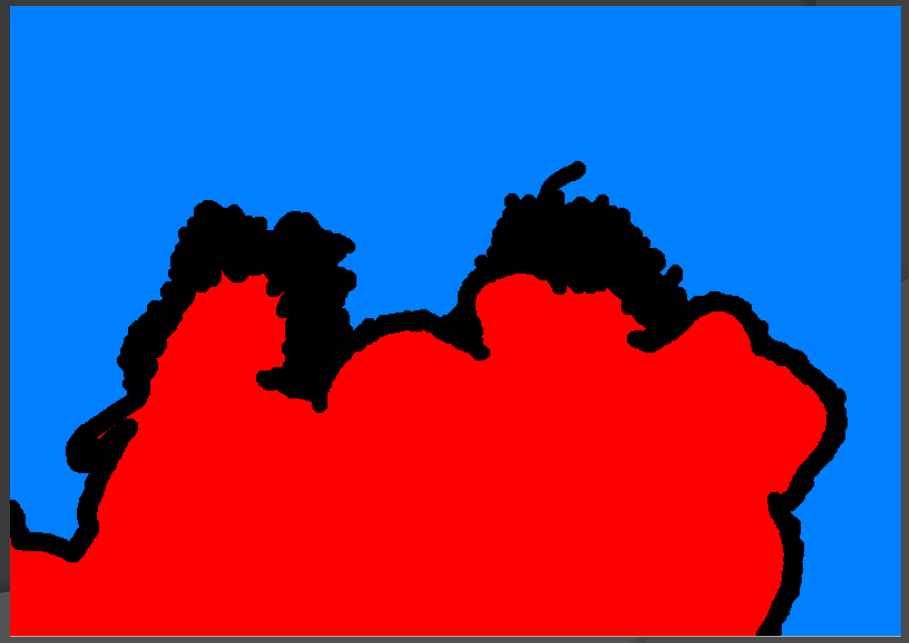


Generating sampling set

Split FG and BG regions into subregions

Input image

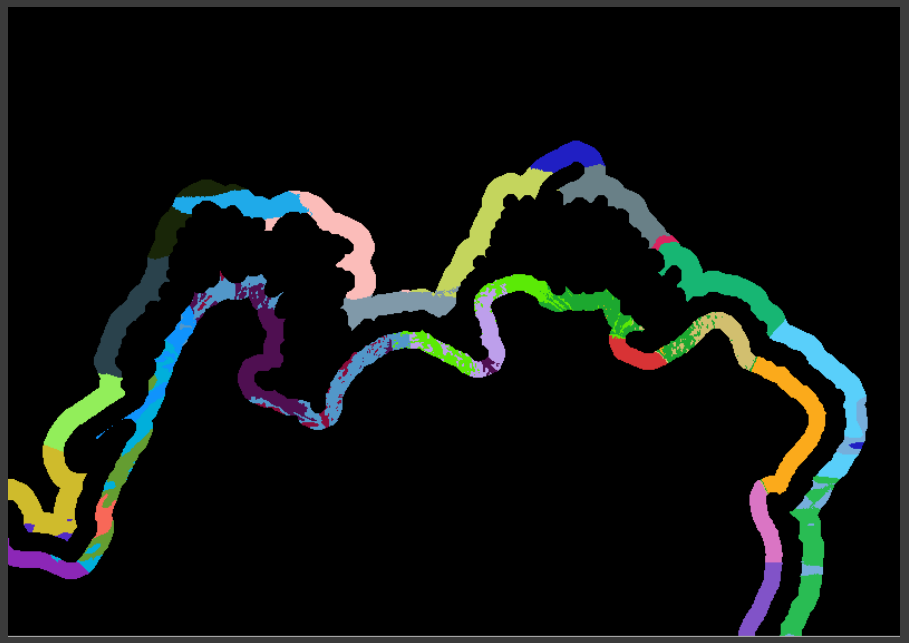
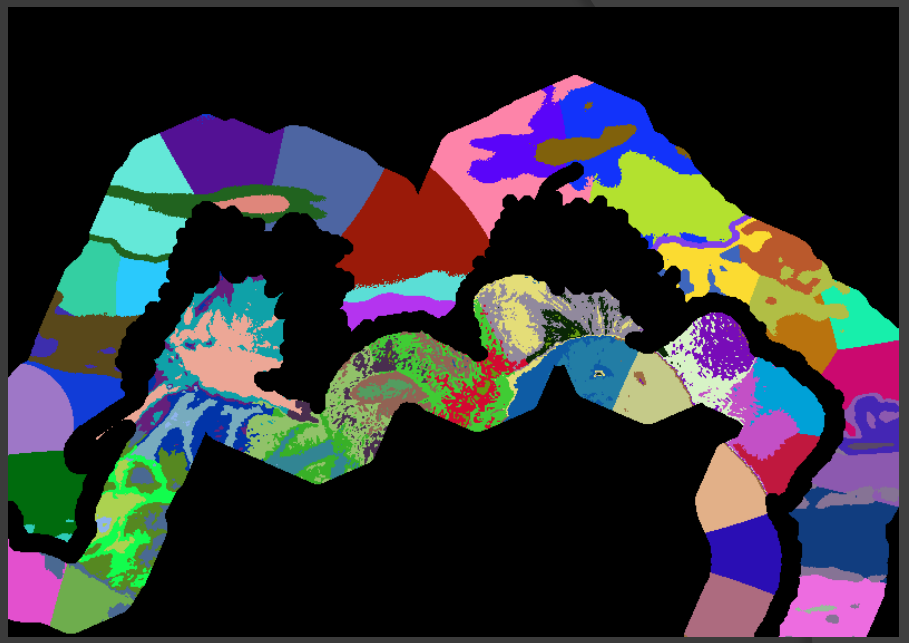
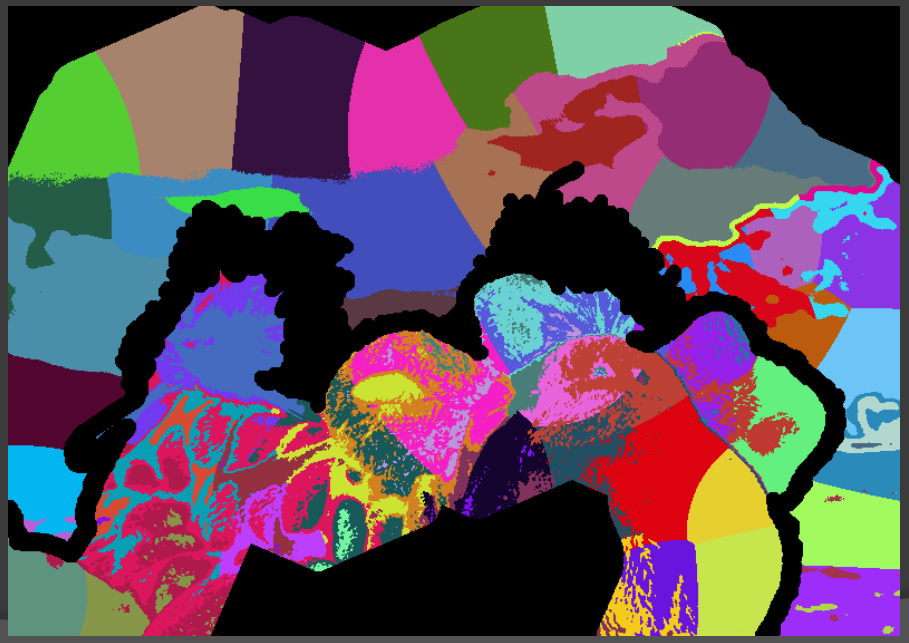
User defined trimap

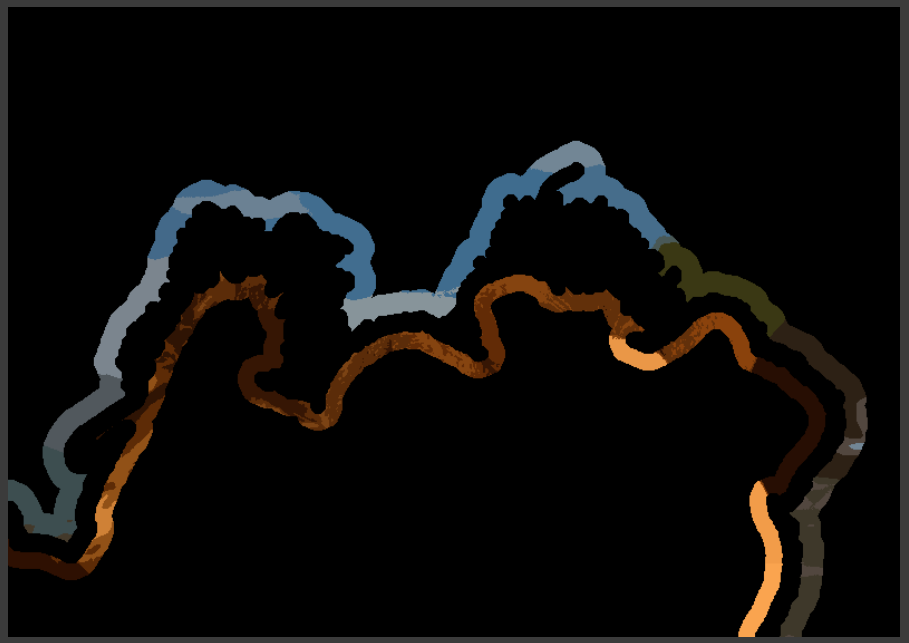
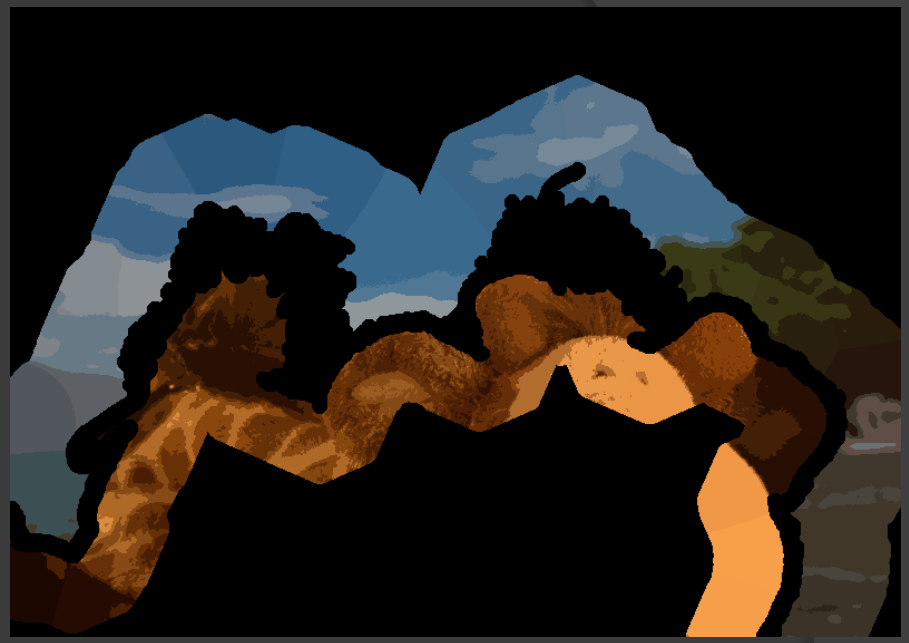
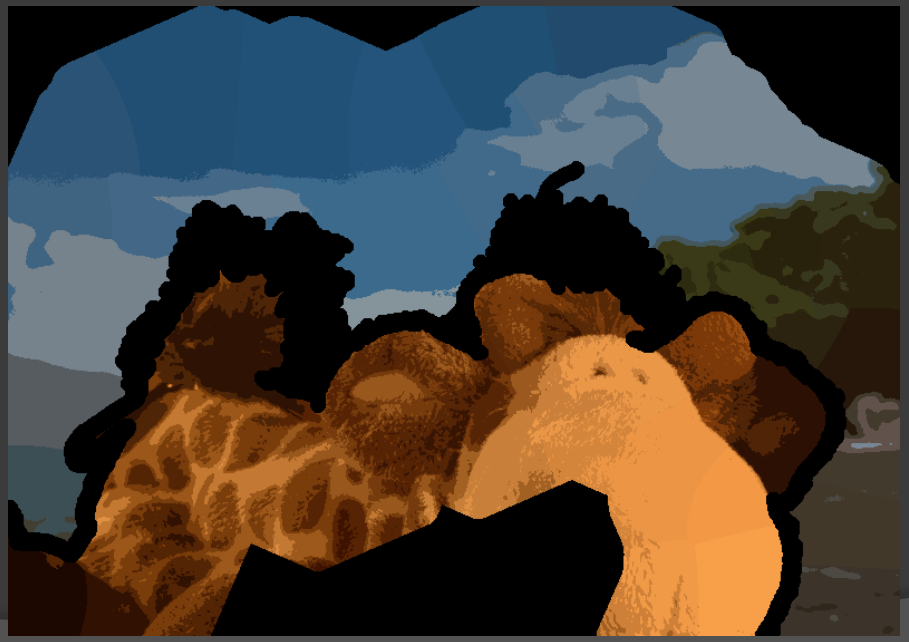
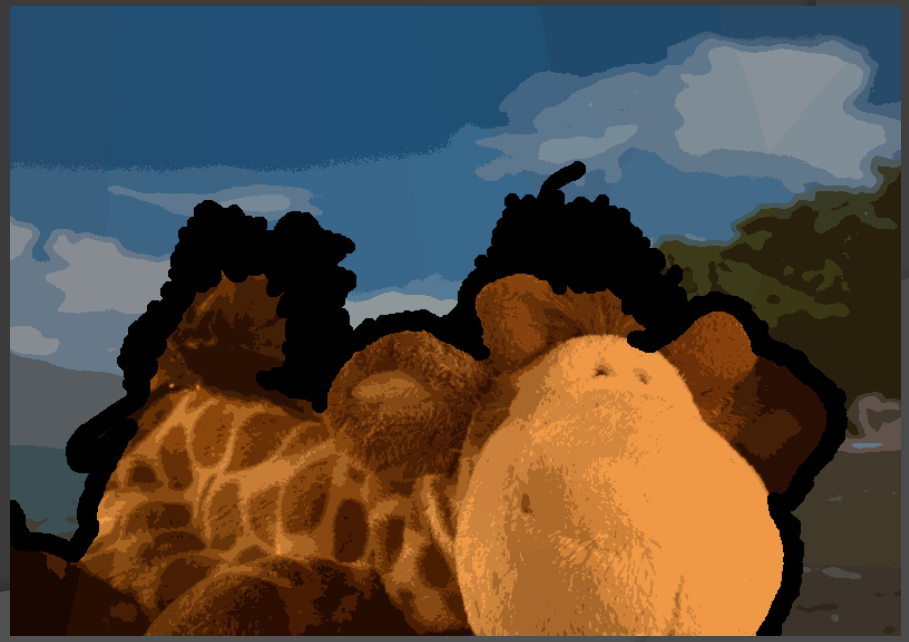





Generating sampling set

Split FG and BG regions into subregions

Perform two-level color and spatial clustering (GMM)

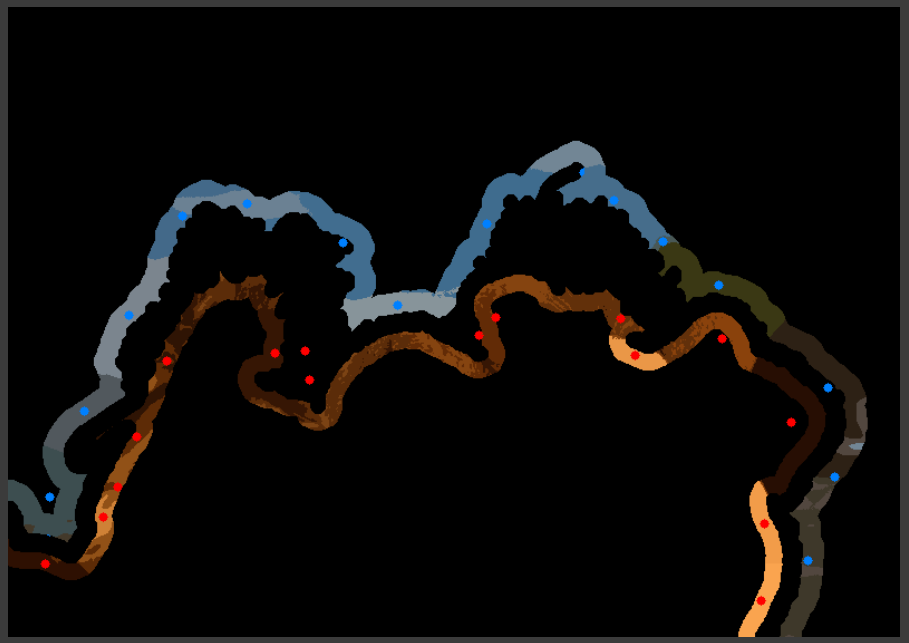
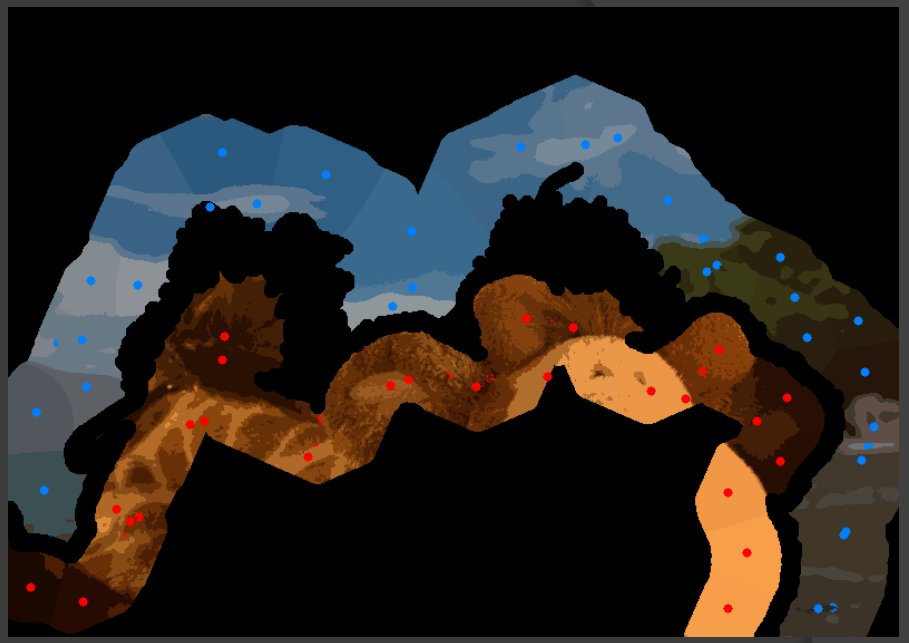
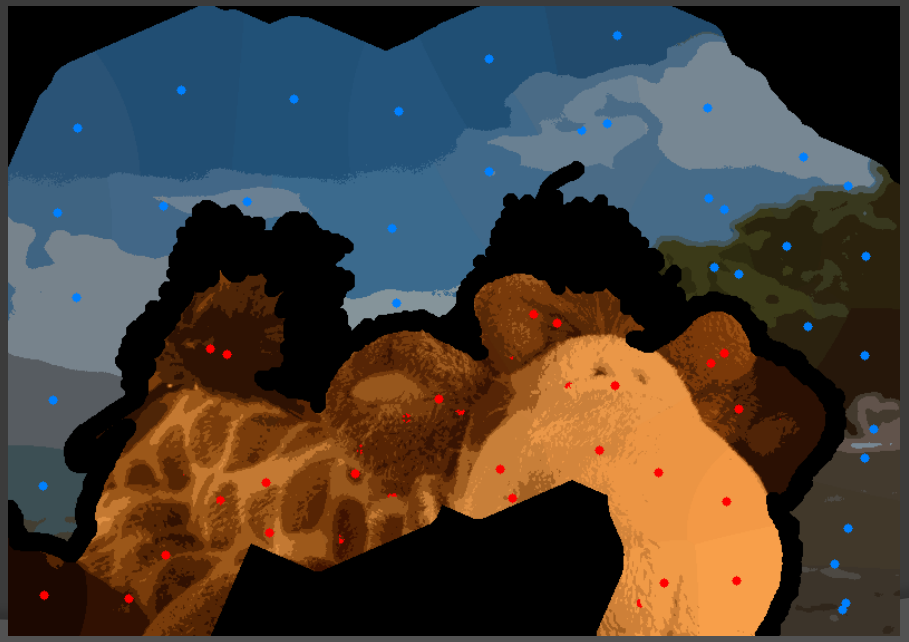
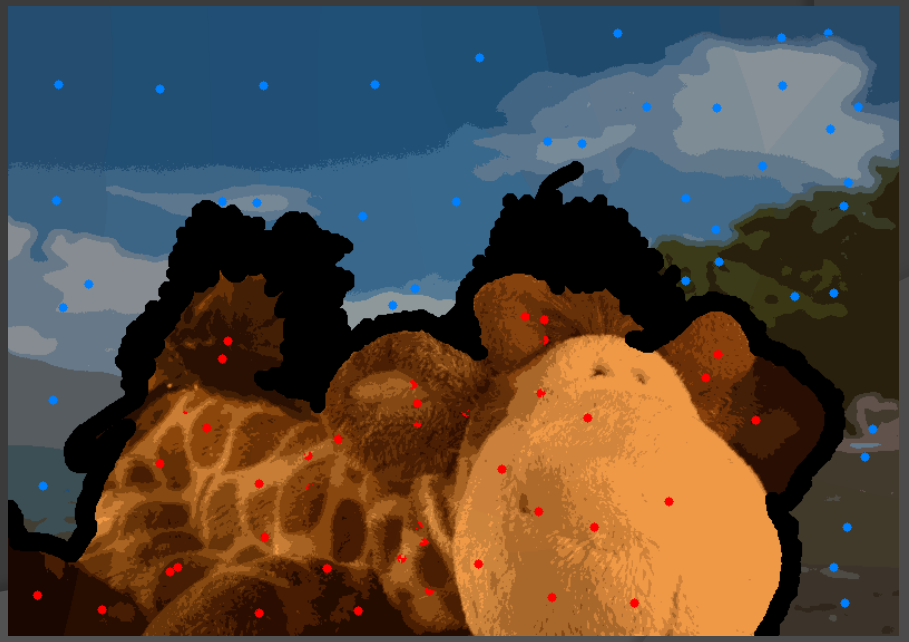






Generating sampling set

Split FG and BG regions into subregions

Perform two-level color and spatial clustering (GMM)

Choose centroids as candidate samples

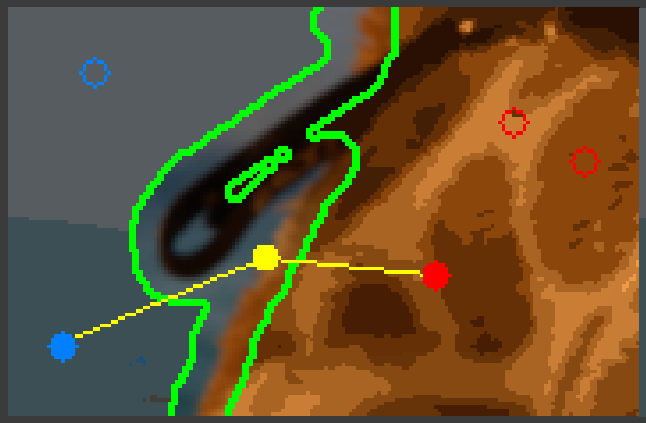





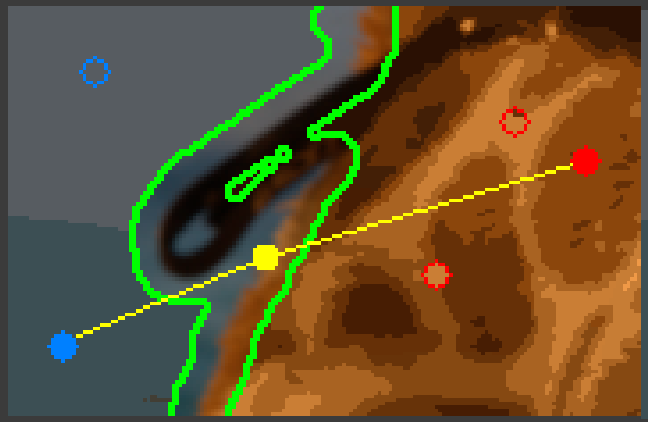
Why is this idea good?

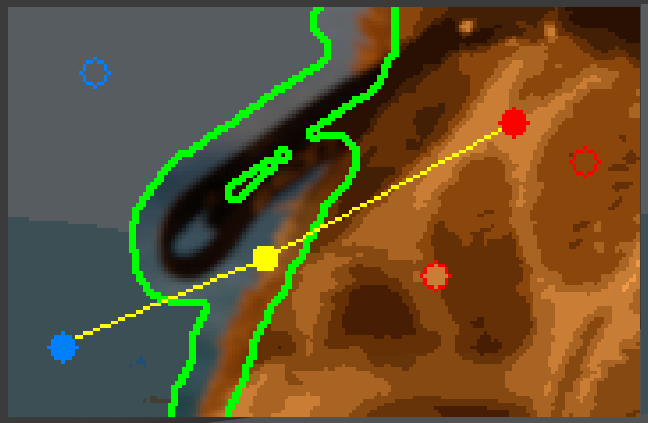
- Comprehensive sample set: each color distribution appears in the sample set
- True samples are never missed

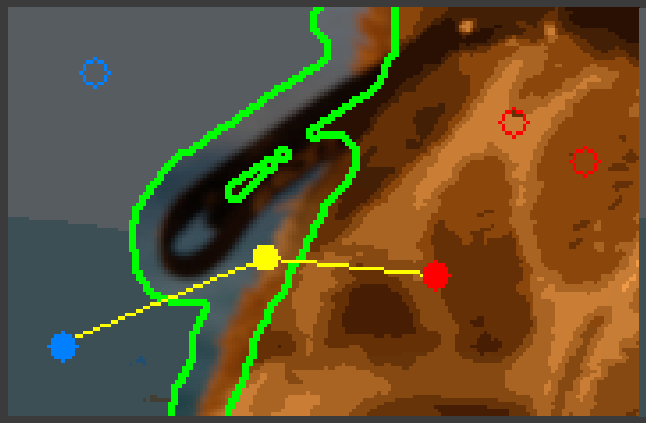
Computing alpha matte

- for each unknown pixel z
  - select corresponding subregion
    (the closer to the boundary the smaller region)
  - try all (F,B) pairs and choose the one which best "explains" pixel z
    (which maximizes a certain objective function among all possible pairs)
  - Compute alpha using compositing equation

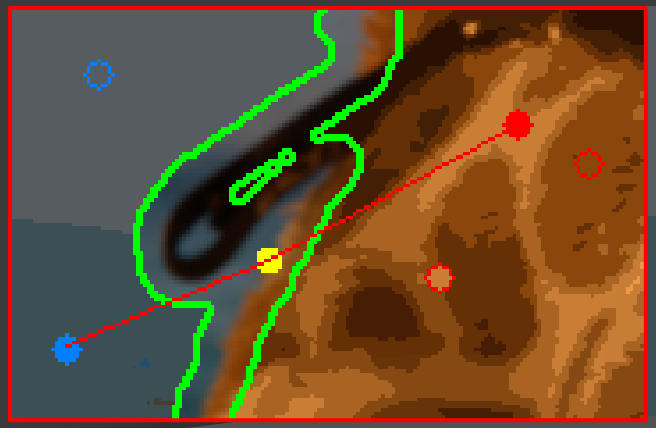







Objective function

(F_i,B_i) are not too far from I_z

$$O_z(F_i, B_i) = K_z(F_i, B_i) \times S_z(F_i, B_i) \times C_z(F_i, B_i)$$

(F_i,B_i) can explain color I_z well

favors well-separated color distributions

Color-based term

Observed pixel = good convex combination of (F_i,B_i)

$$K_z(F_i, B_i) = exp(-||I_z - (\alpha F_i + (1-\alpha)B_i||)$$

Spatial term

Selected pair should be close to observed pixel

$$S_z(F_i, B_i) \propto exp(-||z - F_i^s||) \times exp(-||z - B_i^s||)$$

Overlapping term

Cohen's "d value"

Selected candidates should come from well-separated color distributions

$$C_z(F_i, B_i) \propto d(F_i, B_i) = \frac{\mu_{F_i} - \mu_{B_i}}{\sqrt{\frac{(NB_i - 1)\sigma^2_{B_i} + (NF_i - 1)\sigma^2_{F_i}}{NB_i + NF_i - 2}}}$$

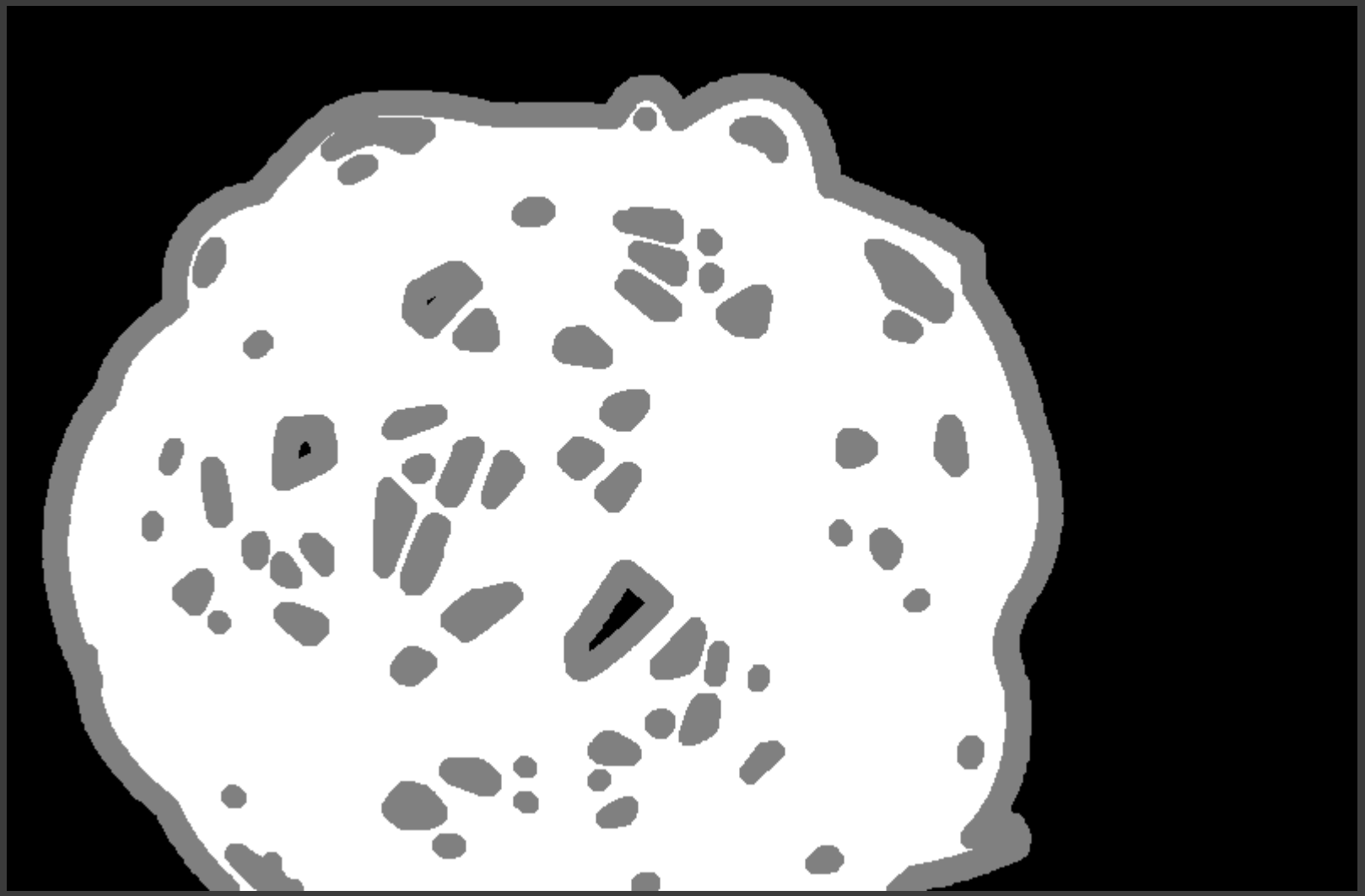
inversely proportional to overlap between distributions

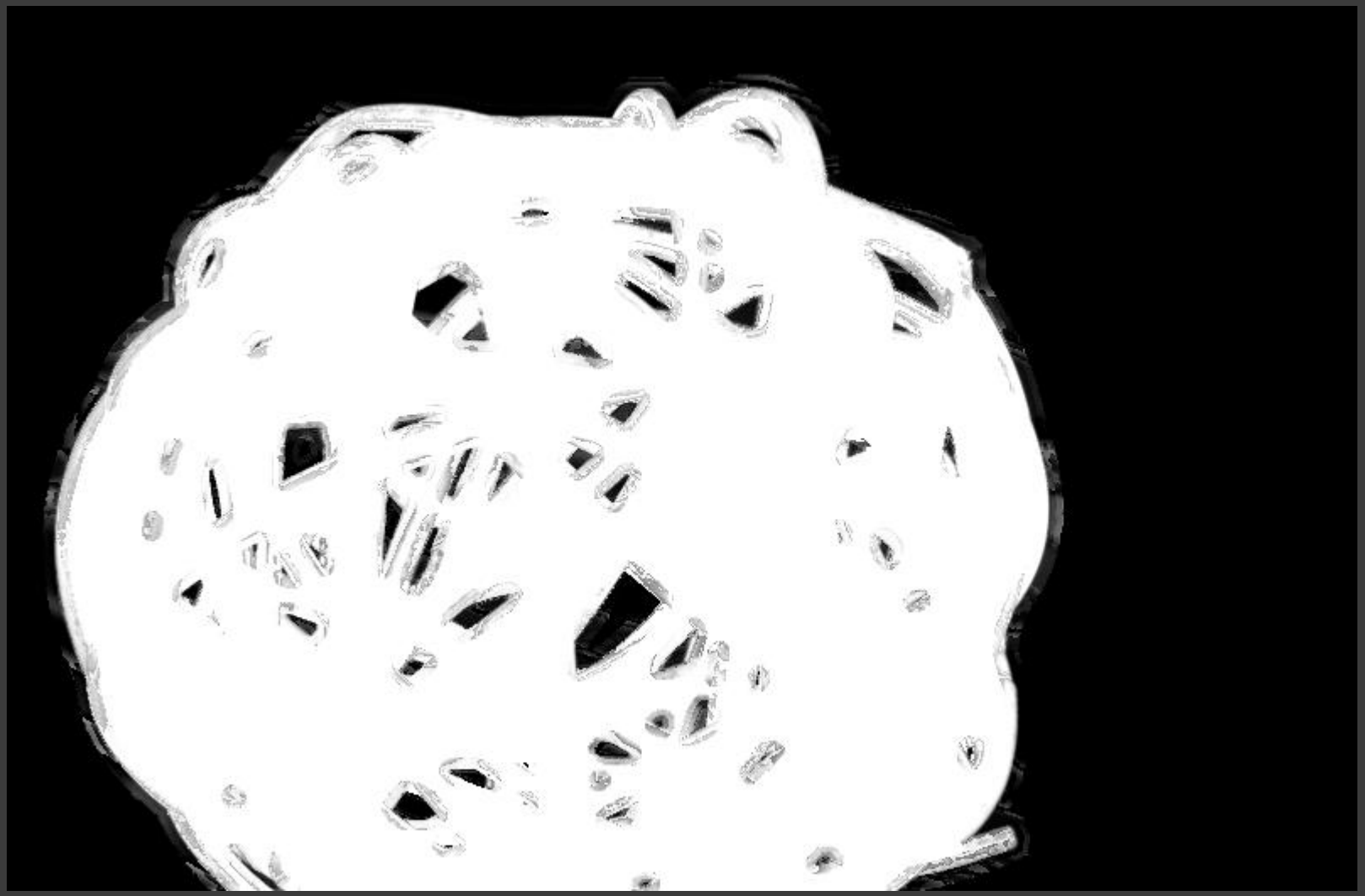
Pre/post processing

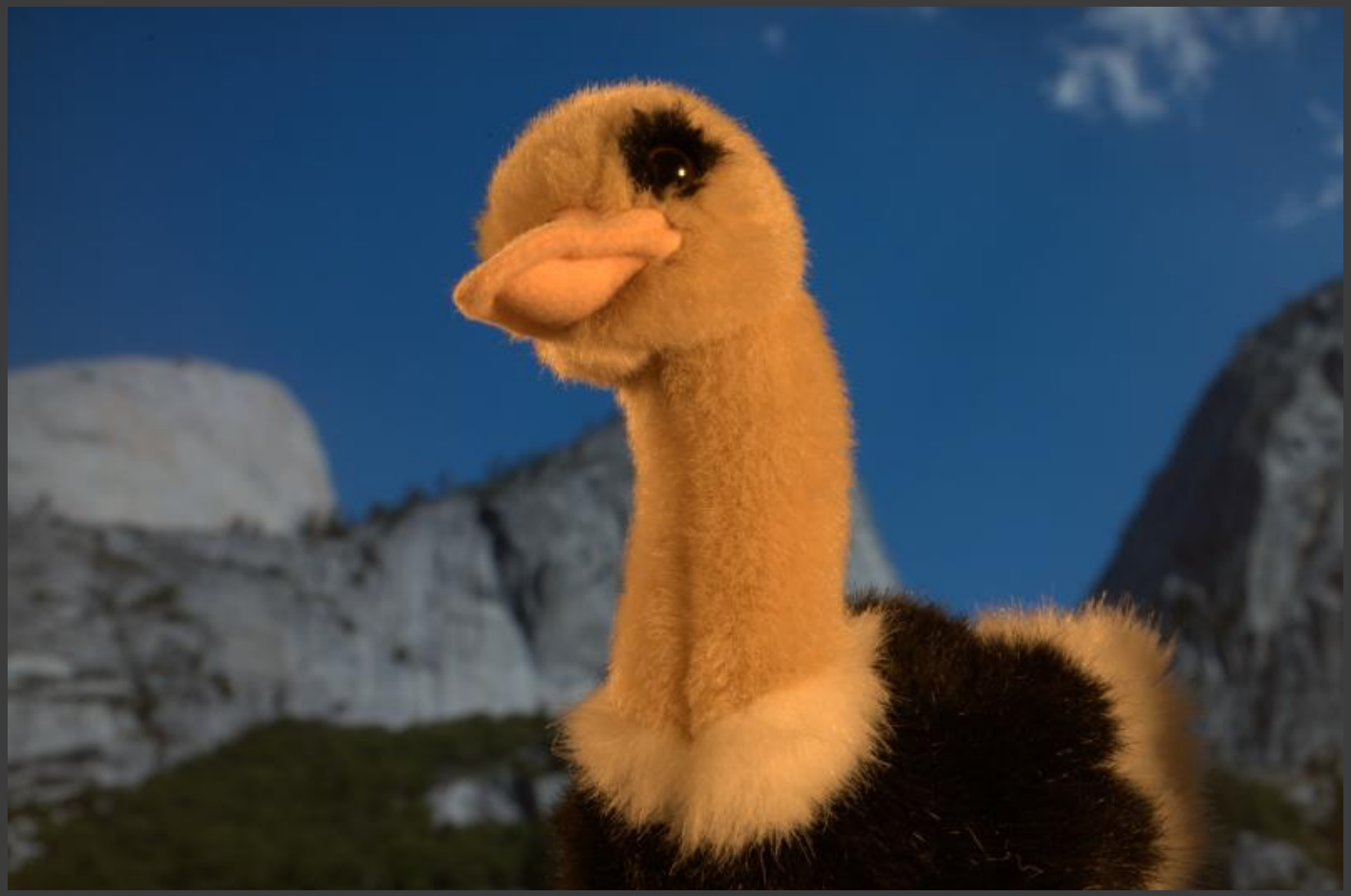
- Used to refine the alpha matte
- Pre-processing: known regions are expanded for pixels close to the boundary (wrt color and spatial distance)
- Post-processing: laplacian method to smooth alpha matte (regions with high confidence have higher smoothing weights)

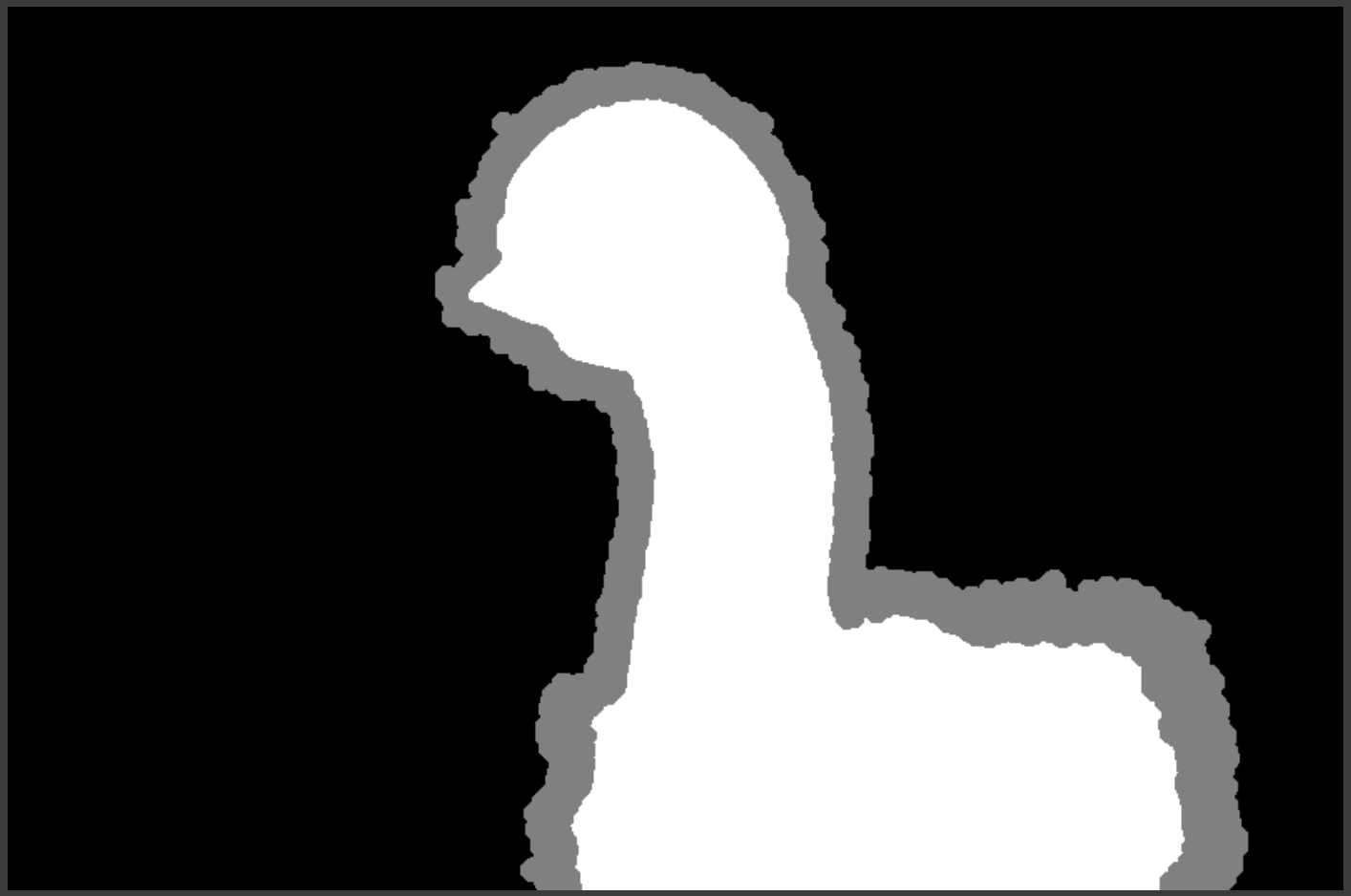
Results

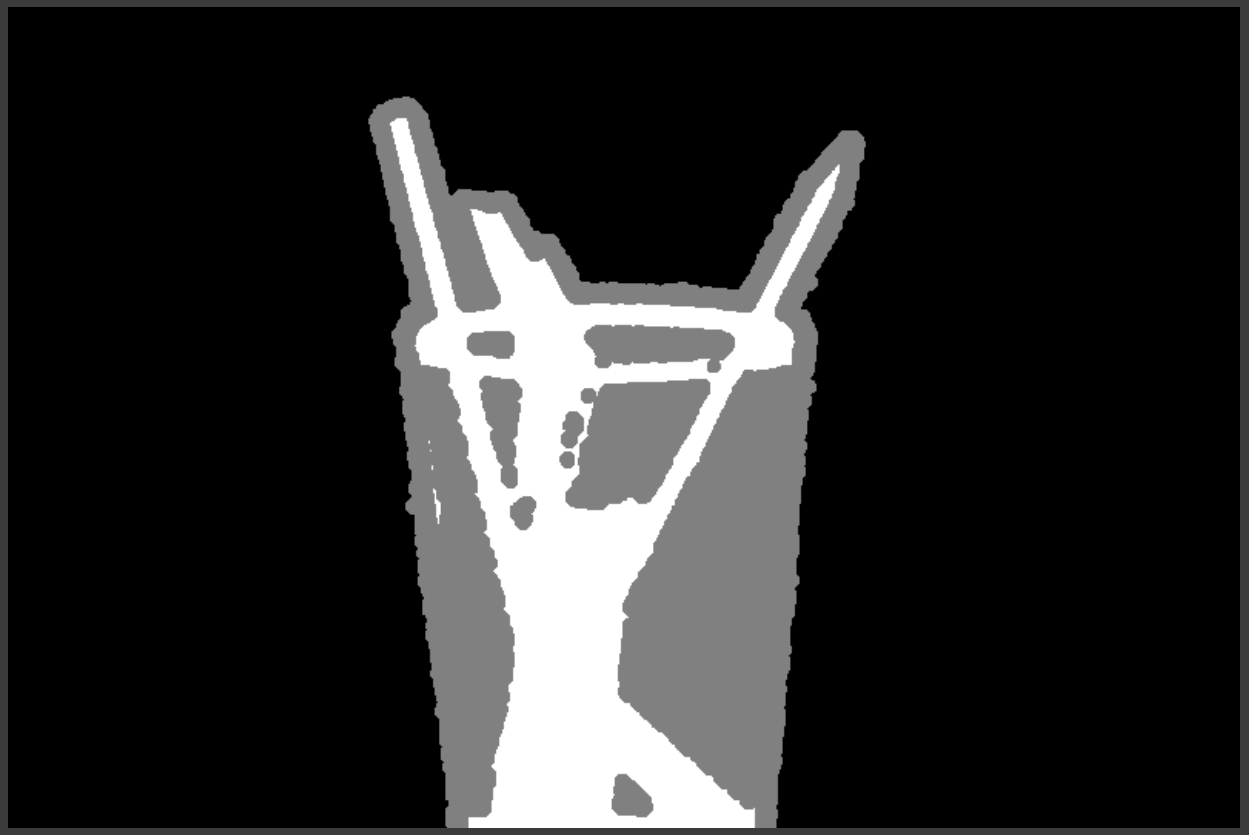
- Everything implemented in C++ using OpenCV
  - Region partitioning
  - Two level hiarchical clustering
  - Brute-force objective maximization
  - Alpha-matte computing
  - ~~Pre-post processing~~

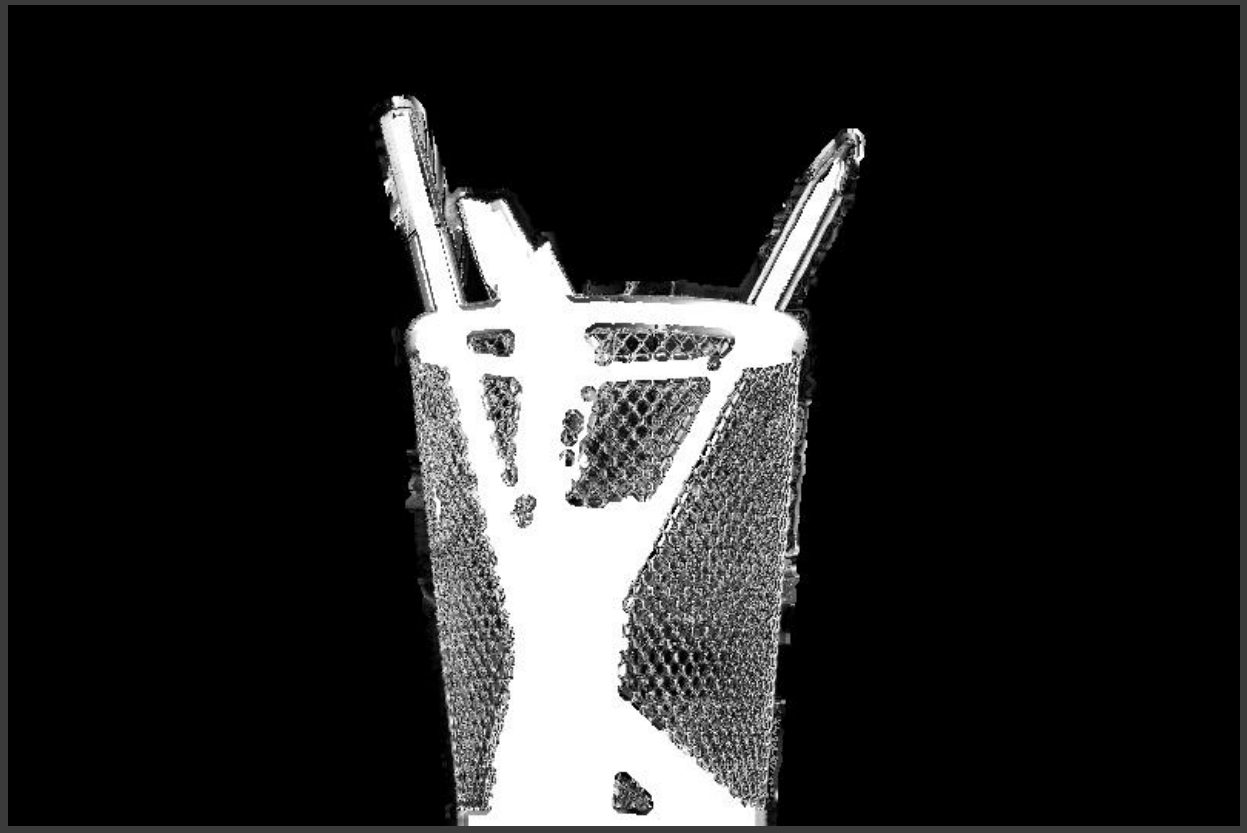












Choice of parameters

- Paper not comprehensive enough: only a shallow description of the algorithm
  - Number of subregions? Size of each subregion?
    Fixed number of subregions (4), width grows quadratically
  - Number of color/spatial clusters? (“number of peaks in the histogram”…)
    For each subregion, fixed number of color/spatial clusters
  - Multi-dimensional overlapping term?
    L^2-norm of per-component Cohen’s d value

Thank you for your attention

Demonstration if time permits