

Towards Real-Time Object Removal and Inpainting Through a Diminished Reality Application for Smartphones

Henry Maximilian Cording

Supervised by Baran Usta and Dr. Michael Weinmann. Responsible Professor: Dr. Elmar Eisemann.

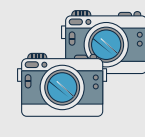
June 27th, 2022

1 INTRODUCTION

- In classic AR, existing objects in the scene cannot be altered, only augmented.
- Diminished Reality (DR) can remove, conceal, or replace real objects in a scene.



DR research has focused on desktop CPUs for programming and evaluation.



Typically, additional hardware or sensors are used to achieve image completion.

In this research, DR is integrated into a smartphone app to achieve inpainting, while using only the phone's RGB camera.

2 RESEARCH QUESTION

"Can the real environment within an AR scenario be modified, and holes in the output arising from these changes be inpainted in real-time, to create convincing manipulations of scenes using only a smartphone?"



Figure 1: The inpainting process. researchgate.net/figure/Inpainting-example-1-obtained-by-applying-the-proposed-method-to-larger-missing-areasa_fig11_269588939

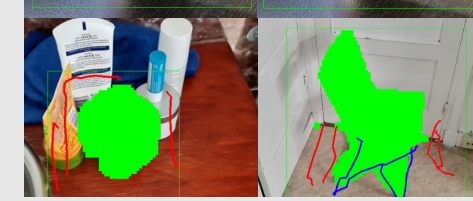
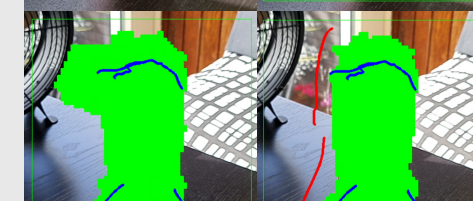
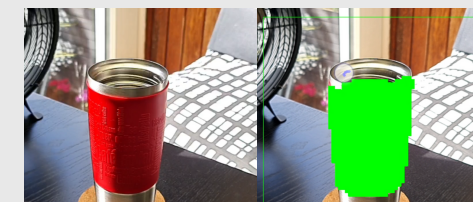
3 METHOD

Extracing Contours



1. Input frame
2. User draws rough selection
3. Downsampling and denoising
4. Canny edge detector
5. Dilation
6. Fill and visualization

Object Selection



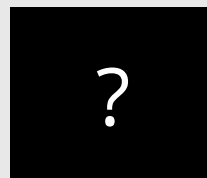
1. Input frame
2. Initial selection not accurate enough
3. User expands selection
4. User retracts selection
5. Selection amidst diverse collection
6. Selecting furniture

GrabCut

Fast Contour Tracking



Frame f : Select largest contour in ROI.



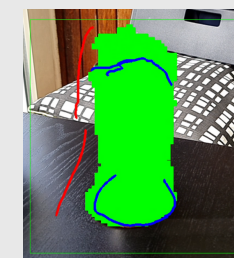
Frame $f + 1$: n contours were found, minimize sum of distance and visual dissimilarity.

$$c_{f+1} = \begin{cases} \max_{c \in C} A(c), & \text{if } c_f = \text{null.} \\ \min_{c \in C} d(o_c, o_{c_f}) + M_c \oplus M_{c_f}, & \text{otherwise.} \end{cases}$$

Object Tracking



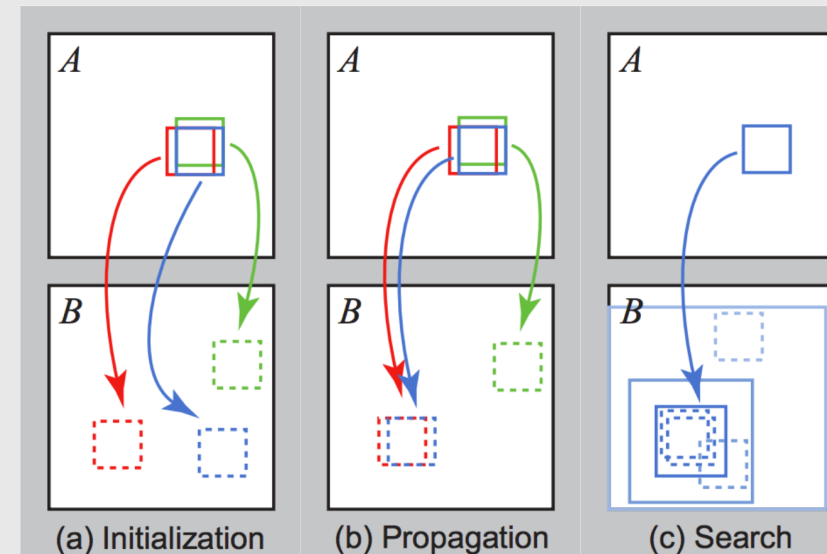
Frame f : Initialize tracker with f and bounding rectangle.



Frame $f + 1$: Provide frame, tracker returns estimated ROI. Based on size/position of ROI, the adjustments for GrabCut are scaled/translated.

KCF & CSRT Trackers

PatchMatch Overview

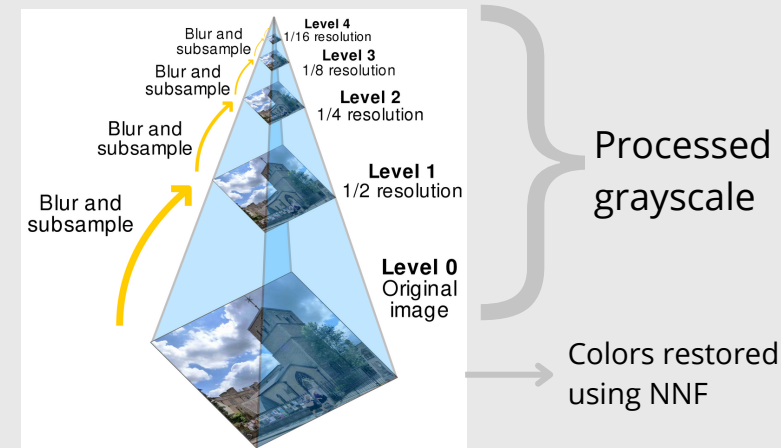


connellybarnes.com/work/class/2013/cs6501/proj1/

Nearest neighbor field (NNF) maps masked patches to outside patches, and improves iteratively.

Inpainting

Details & Optimizations



commons.wikimedia.org/w/index.php?curid=42549151

$$d(A, B) = \sqrt{(r_A/h - r_B/h)^2 + (c_A/w - c_B/w)^2} * \ln(\ln(L^2(A, B) + 1) + 1)$$

Distance measure determines mapping behavior.

4 RESULTS



Figure 2: Inpainting and augmentation using our application on Android.

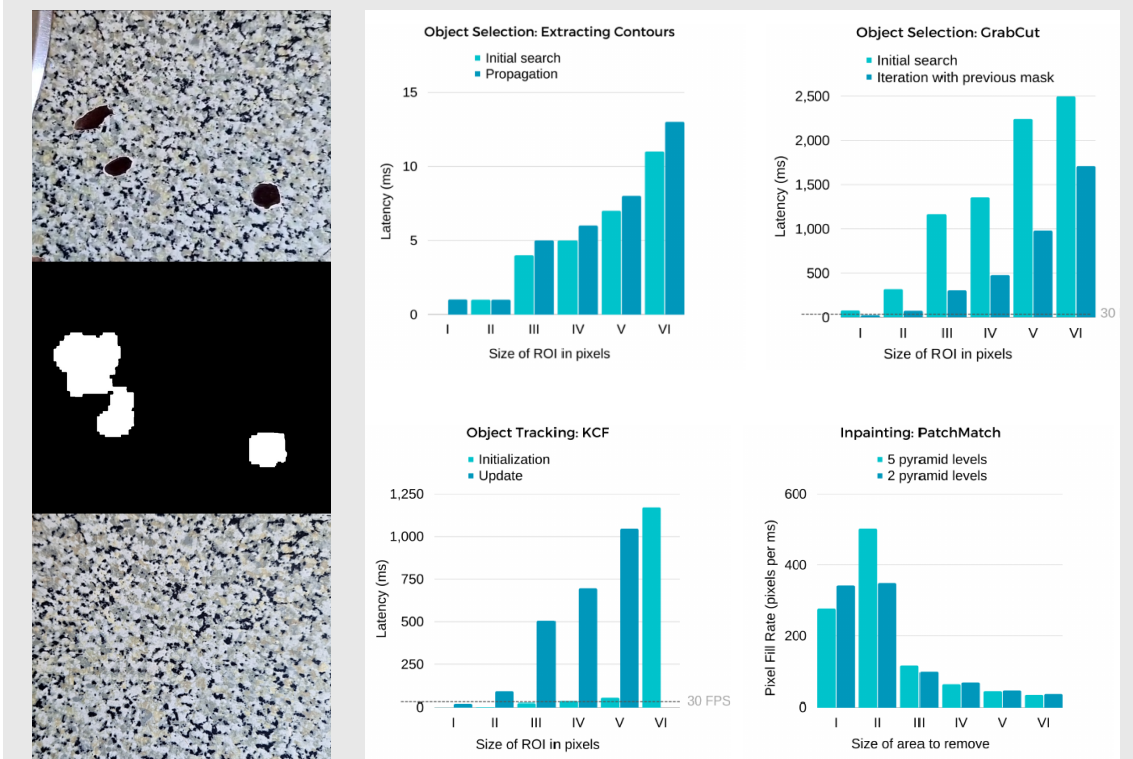


Figure 3: Inpainting is convincing over unstructured textures with planar background.

Figure 4: Measurement of latency and pixel fill rate for DR algorithms.

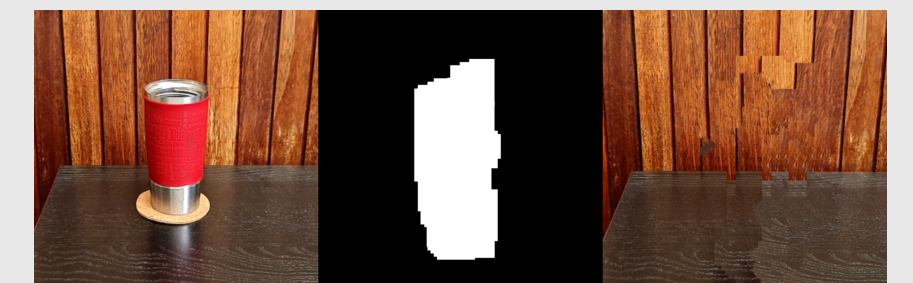


Figure 5: Reduced coherence for complex scenes.

5 DISCUSSION & CONCLUSION

- Extracting contours is fast, but unreliable.
- With GrabCut and KCF, this is the opposite.
- Inpainting worked well for planar surfaces and unstructured textures. Coherence is lost at plane intersections (Fig. 5).

Answer to research question: Real environments in an AR scenario can be modified, and holes in the output inpainted using only a smartphone. However, for real-time performance, many more optimizations are required. The mobile platform is a restricting factor.