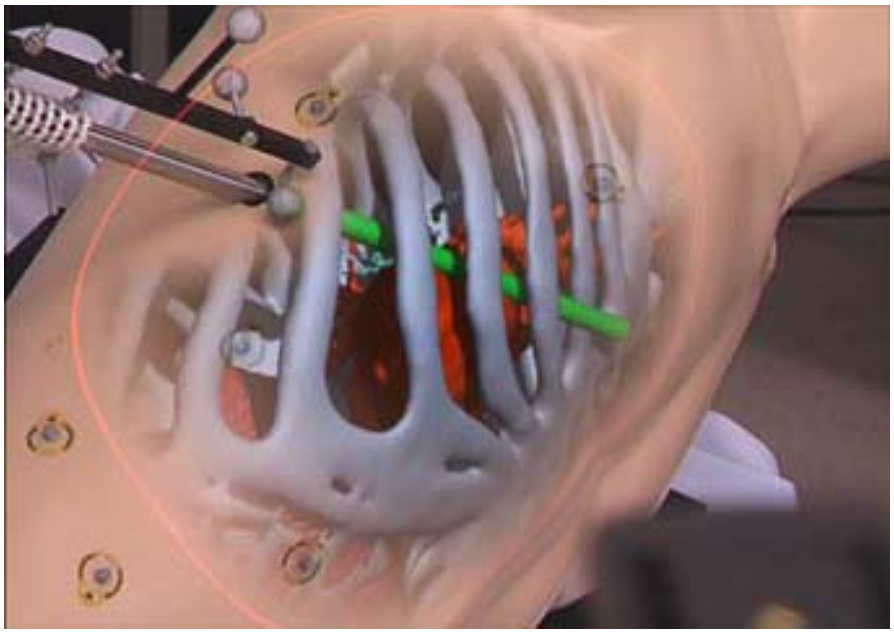
VU Augmented Reality on Mobile Devices

- Introduction – What is AR
- Interaction Techniques
- Navigation, Collaboration
- *Visualization Techniques*
- Visual Coherence
- Tracking
- …

Includes material from Denis Kalkofen, TU Graz,
Georg Klein Oxford, Jan Fischer

Augmented Graphics



- Information
  - Visualization
  - Understanding
- Virtual Objects
  - Graphics
  - Realism

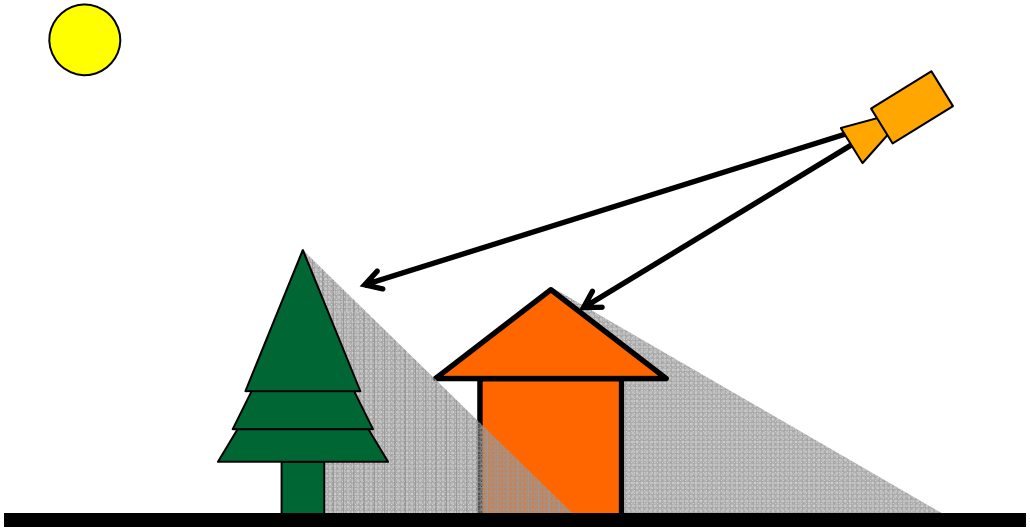
Topics

- Occlusion
- Lighting
- Camera effects
- Visual effects

3

Scene models

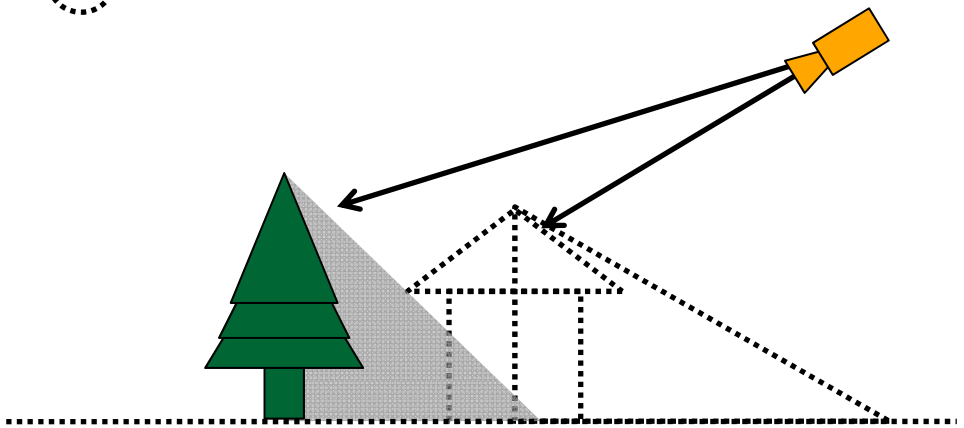
- Computer graphics completely models real scene
- Geometry accessible for calculations
  - Occlusions
  - Lighting
  - Shadows
  - Camera parameters



Augmented Reality

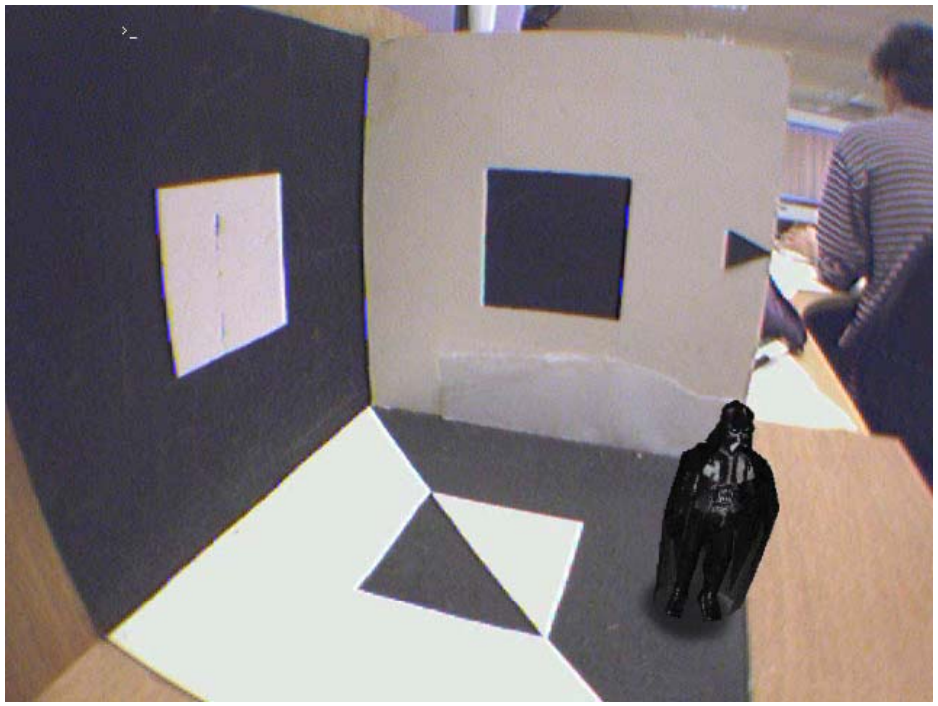
- Less information available
- Final image through compositing
  - Digital in video see-through
  - Physical in optical see-through, projection

- Approximate real world to simulate interaction
  - geometrical, appearance
  - post processing



5

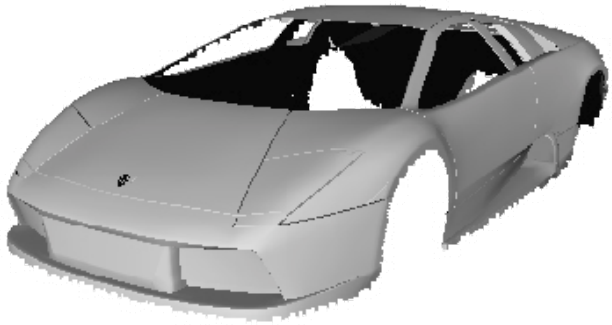
Occlusion Handling



- Virtual in front of real
  - Draw augmentation on top of video background
- Virtual behind real
  - Need strategy to distinguish visible from occluded augmentations

Phantom Rendering

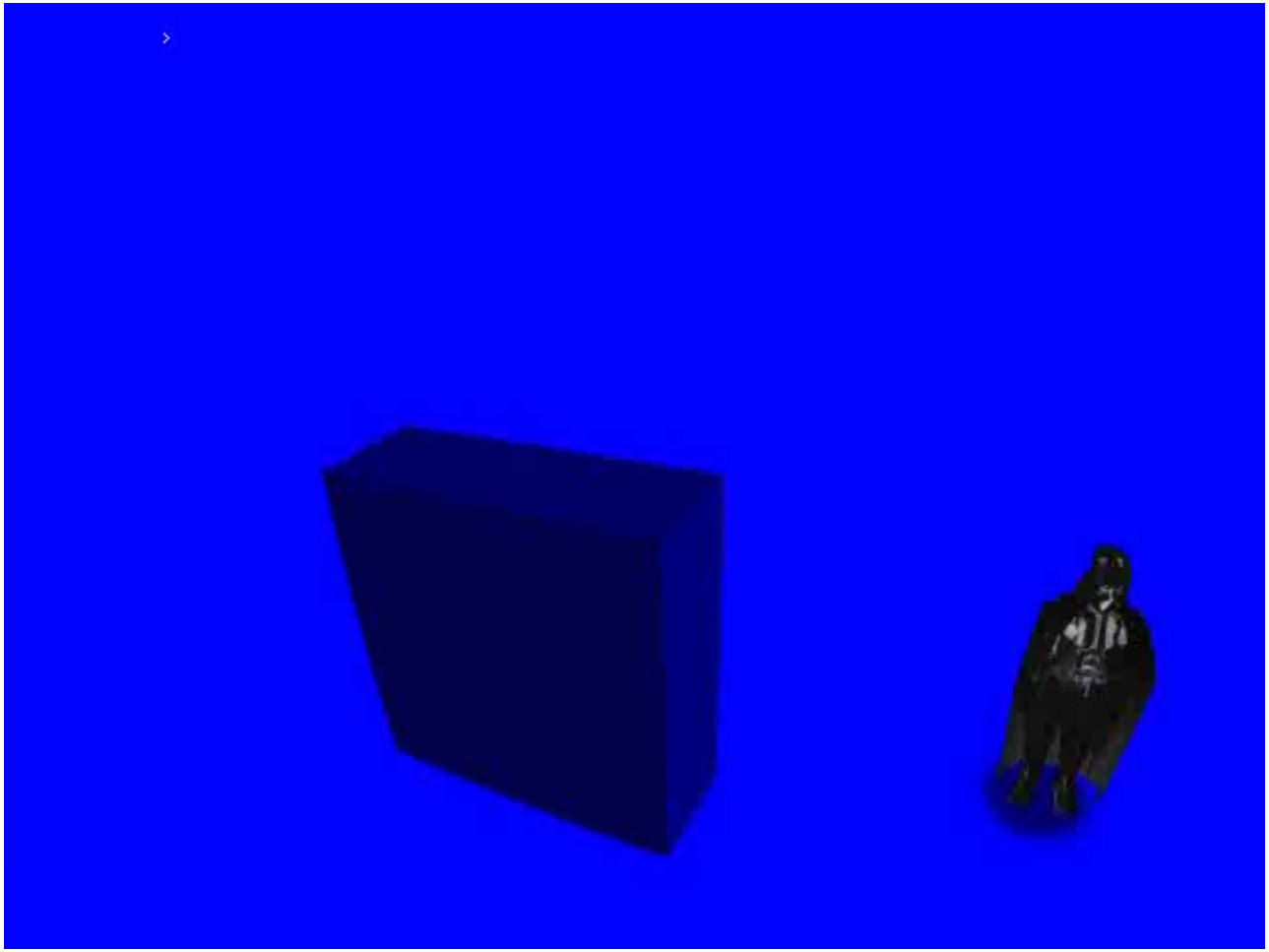
- Render registered virtual representations (Phantoms) of real objects
- Occlusions handled by graphics hardware



1. Draw Video
2. Disable writing to color buffer (glColorMask or glBlendFunc(0,1))
3. Render phantoms of real scene ➔ sets depth buffer
4. Enable writing to color buffer
5. Draw virtual objects

7

Phantom Rendering



Phantom Rendering

- Polygonal data [Breen96,Fiala06]

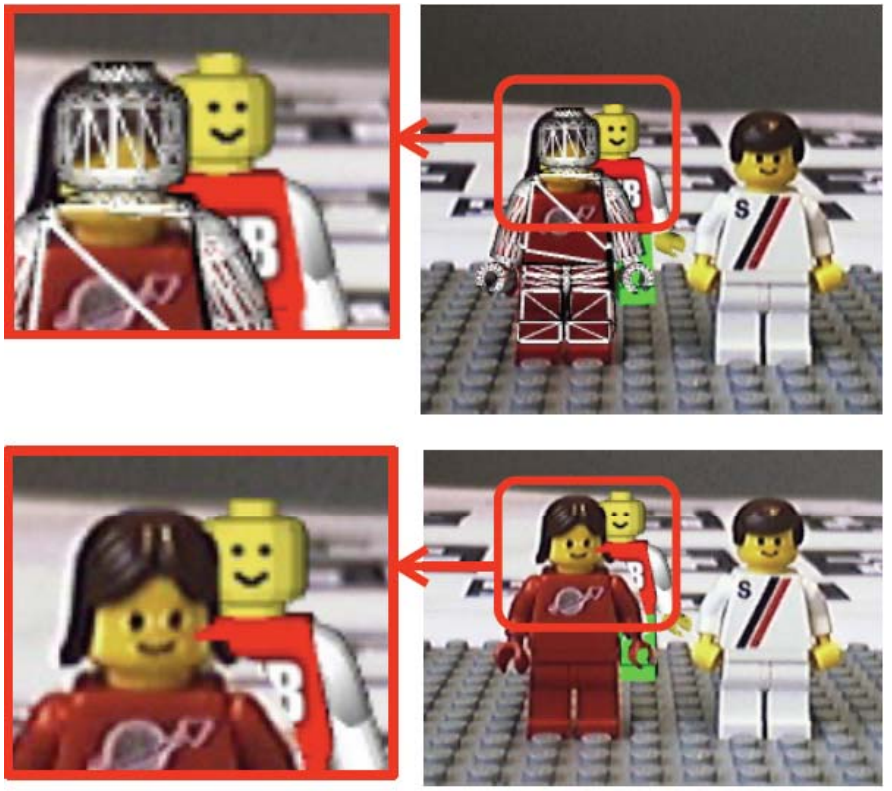
Phantom Rendering

- Volume data
- Phantom = Isosurface using ’FirstHit Raycasting’

[Fischer04]

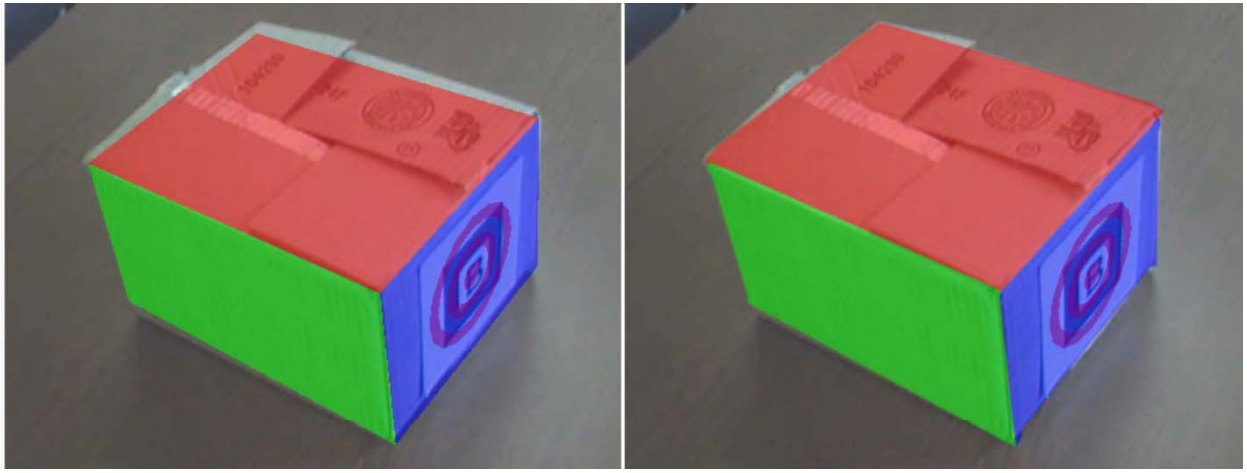
Problems of Phantom Rendering

- Requires accurate
  - Model
  - Tracking data
  - Registration



Occlusion Refinement

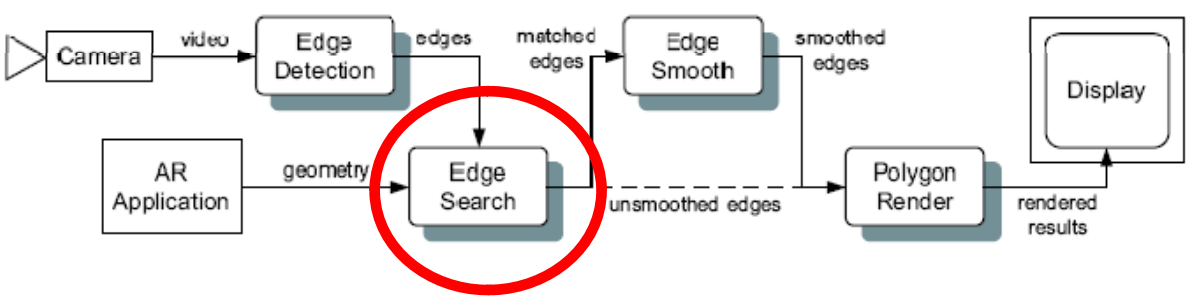
- To overcome poor
  - Model
  - Tracking
  - Registration



[DiVerdi06]

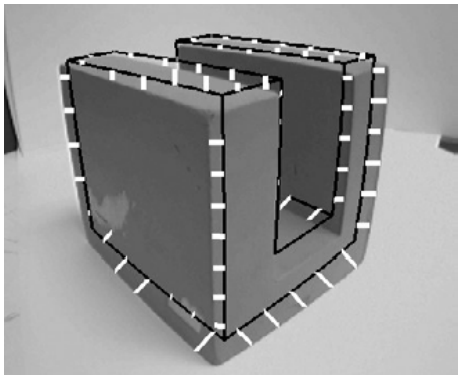
Occlusion Refinement

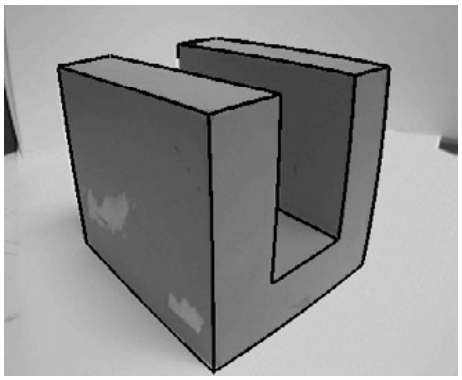
- Image based post-processing



[DiVerdi06]

[Klein05]

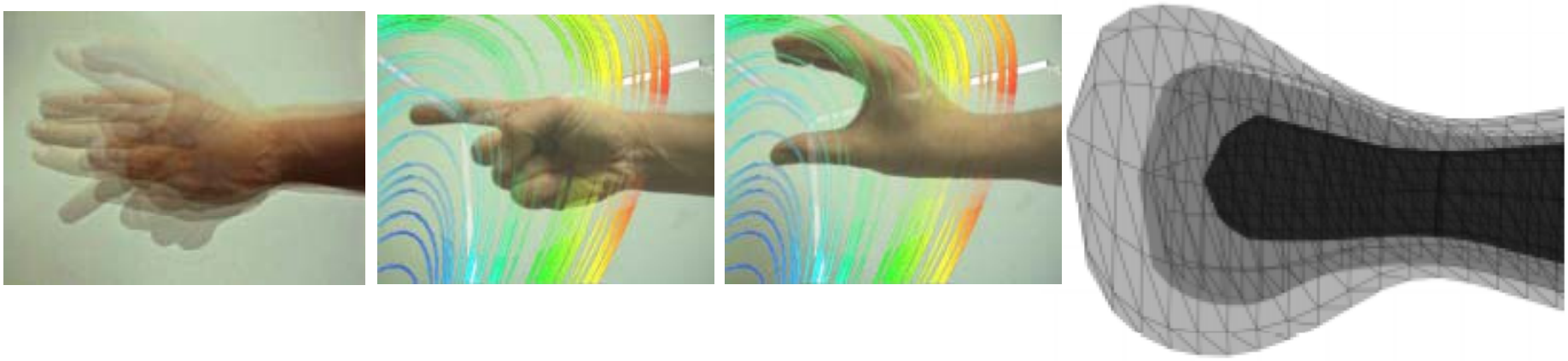




Occlusion Refinement

Probabilistic Occlusion Handling

- Soft transition between occluding and hidden objects
- Compensate for tracking and registration error, by reducing the occluder's transparency depending on the probability of occlusion

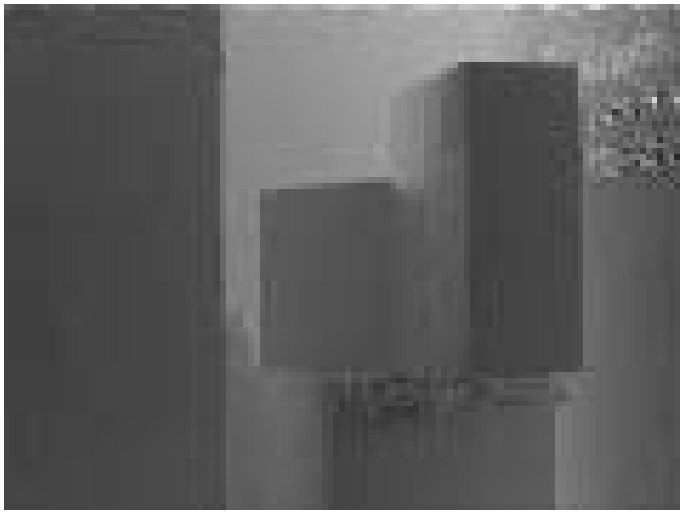


[Fuhrmann99]

Model Free Occlusion Handling

- Instead of tracked & registered phantom model
- Construct depth map from video
  - from Computer Vision
  - Stereo, Shading, Structured Light etc.
- Consider Performance

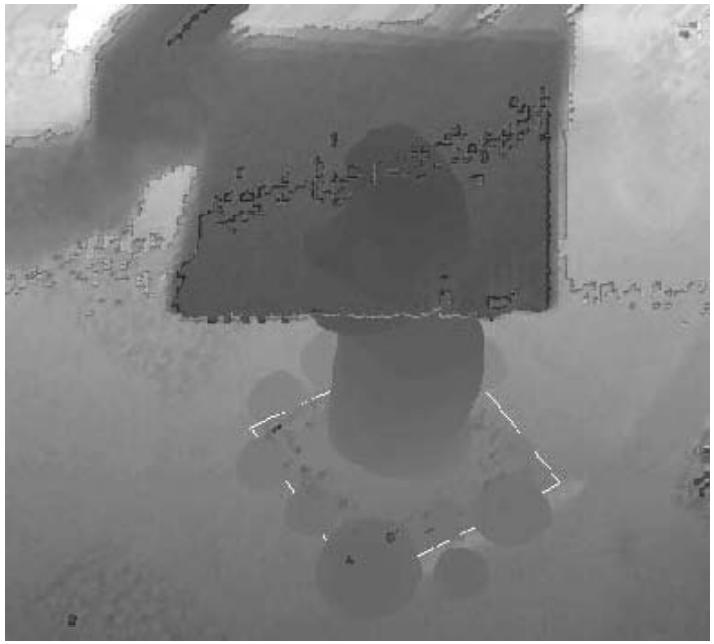
[PointGrey Bumblebee]

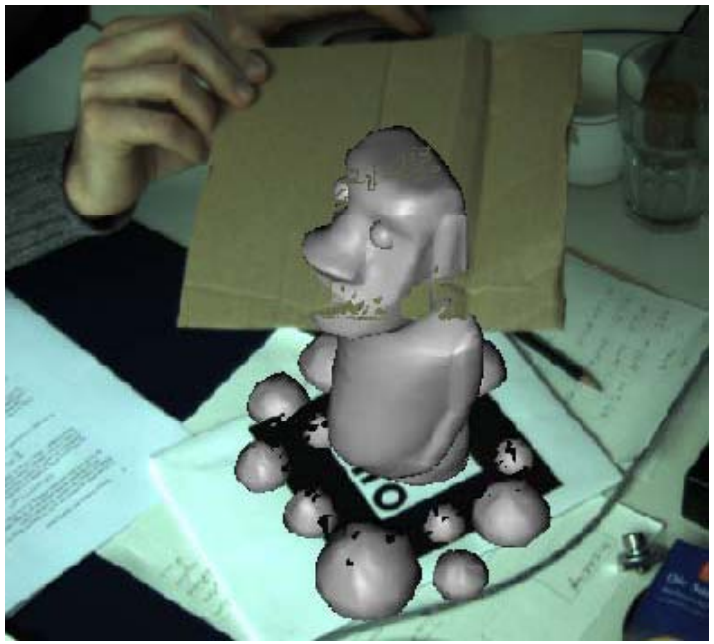


[Fischer07]

Modell Free Occlusion Handling

- Initialize rendering of augmentations with depth map from video

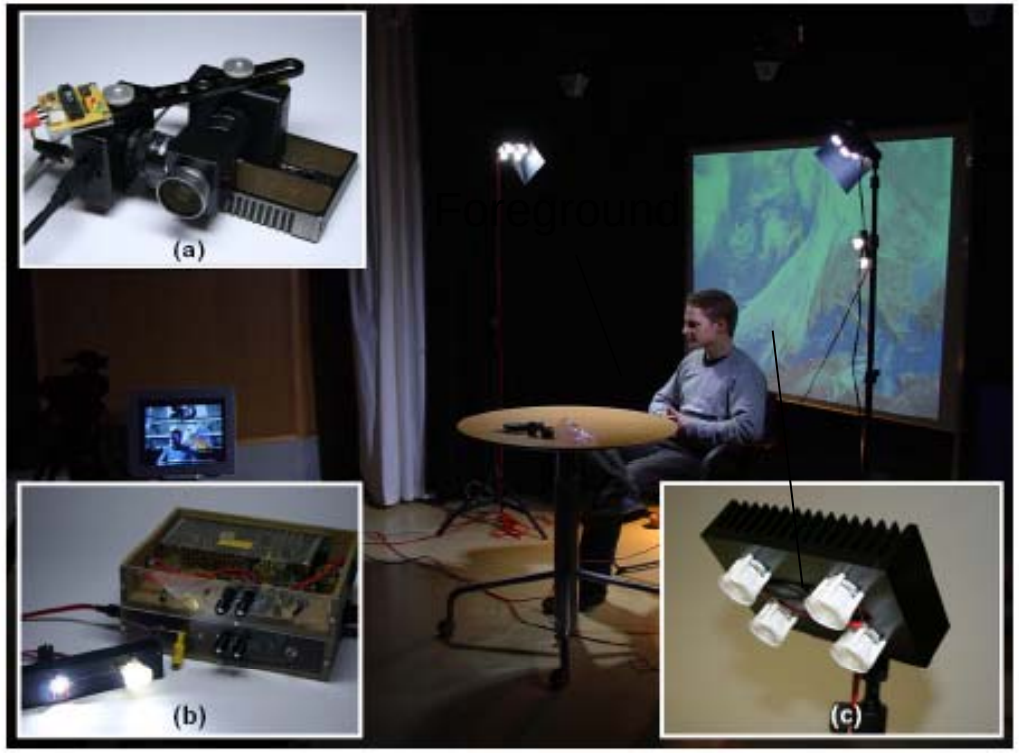




[Fischer07]

Model Free Occlusion Handling

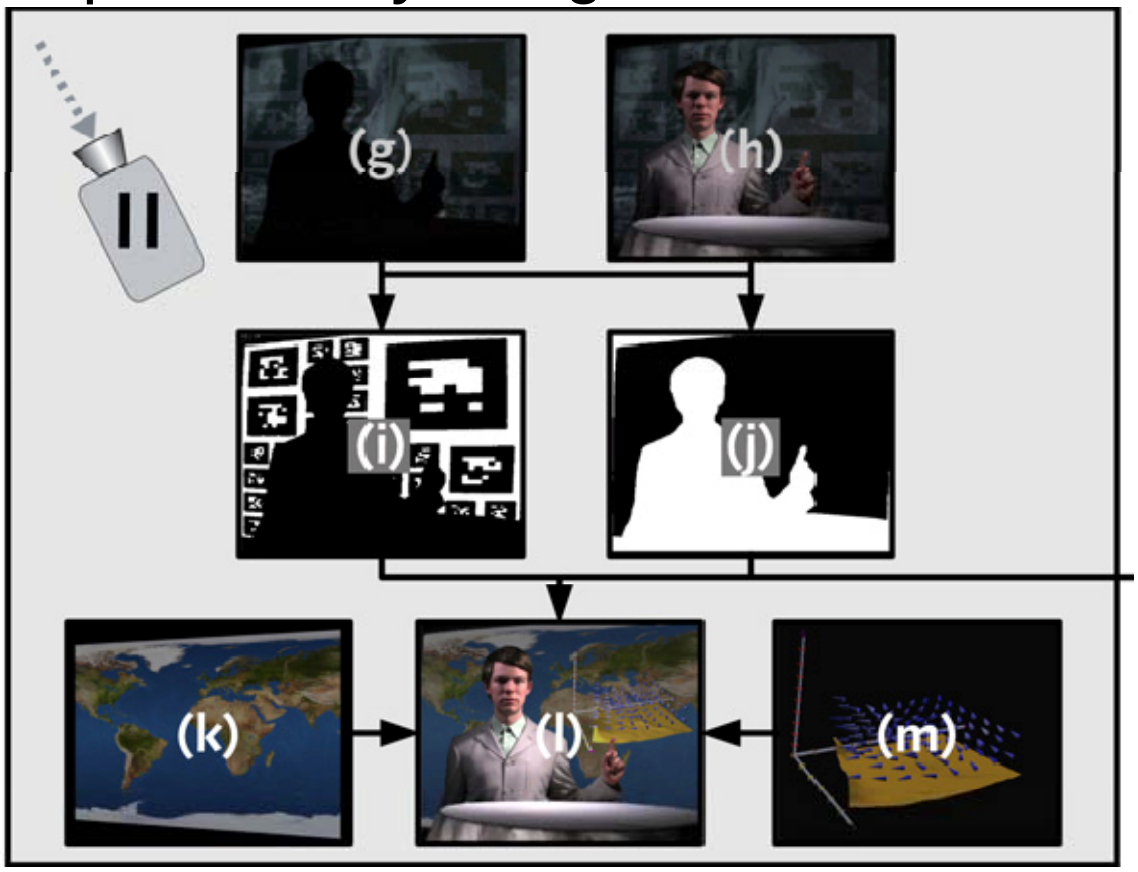
- Using a priori knowledge about scene layout
  - Foreground and background objects are known
- Online separation by foreground illumination



[Grundhoefer07]

Model Free Occlusion Handling

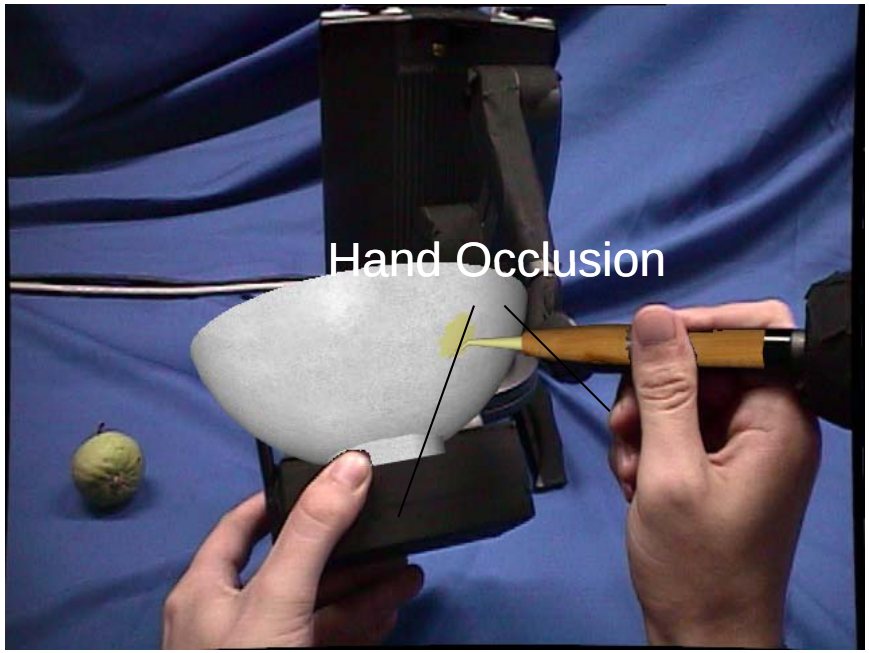
- Online separation by foreground illumination



[Grundhoefer07]

Model Free Occlusion Handling

- Using a priori knowledge about scene layout
  - Hands are always in front and visible
- Hand segmentation by using hand color



[Sandor07]

Semi-Automatic Model Free Occlusion Handling

- Manual a priori foreground segmentation
- Boundary tracking

Boundary

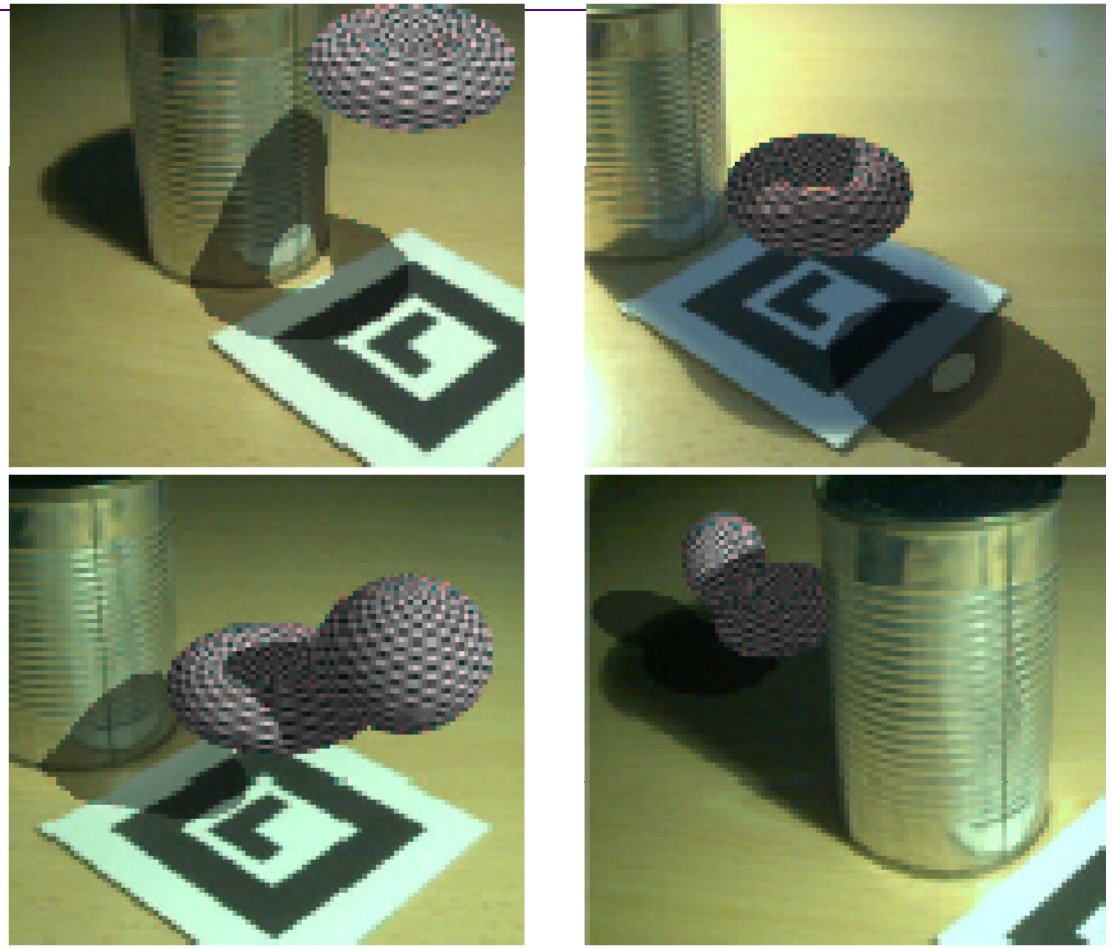
[Lepetit00]

Lighting

- Shadows
- Global illumination
- Interaction between real and virtual objects

Shadows

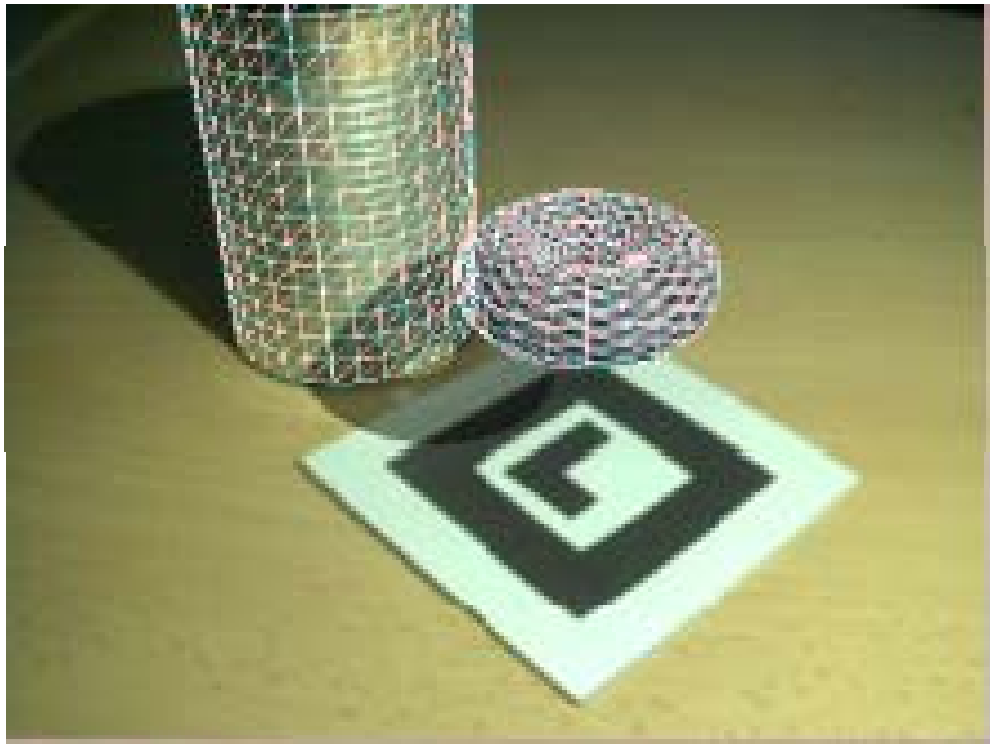
- Important for
  - Depth perception
  - Scene interpretation
- Relations
  - real ➔ real
  - real ➔ virtual
  - virtual ➔ real
  - virtual ➔ virtual



[Haller03]

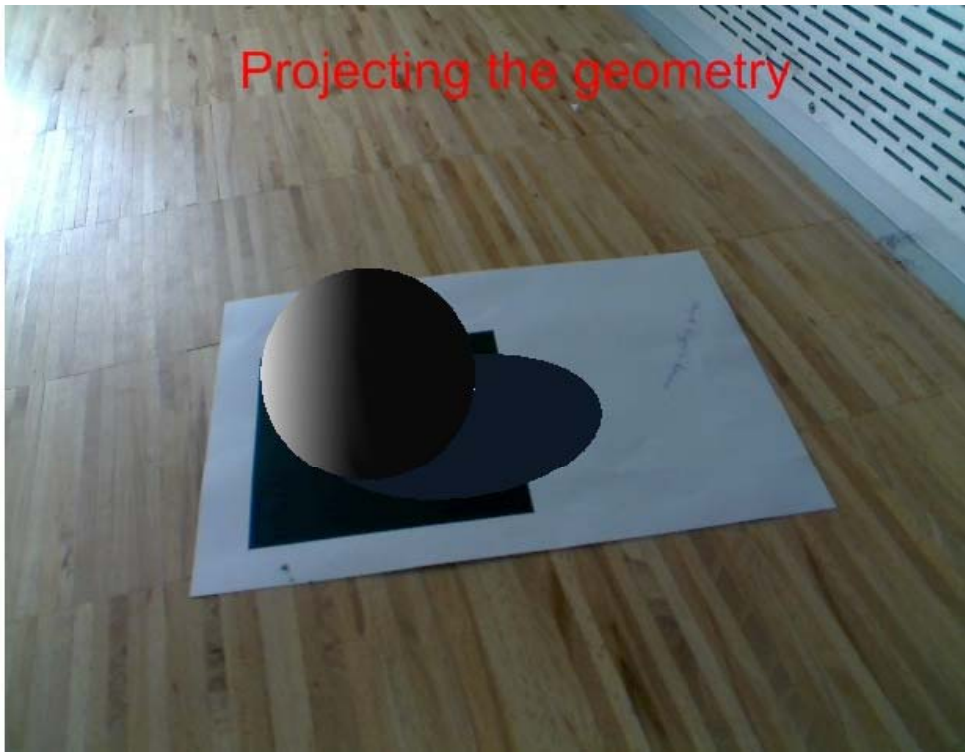
Scene model

- Virtual objects
- Impostors for real objects
- Light sources
  - real must be reflected in shading
  - real must be used to shadow casting



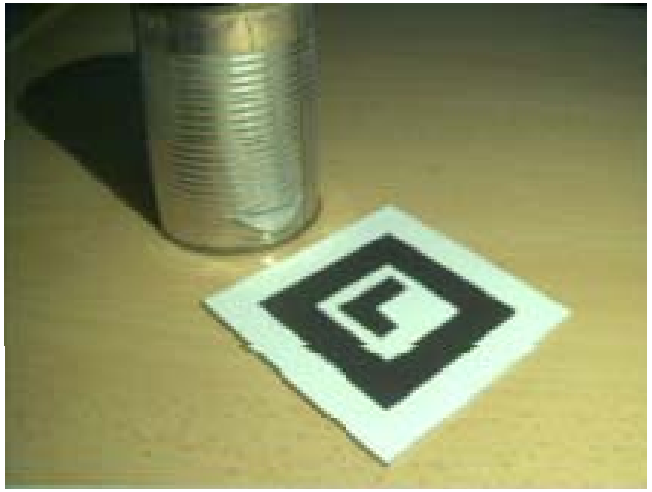
Drop shadow on plane

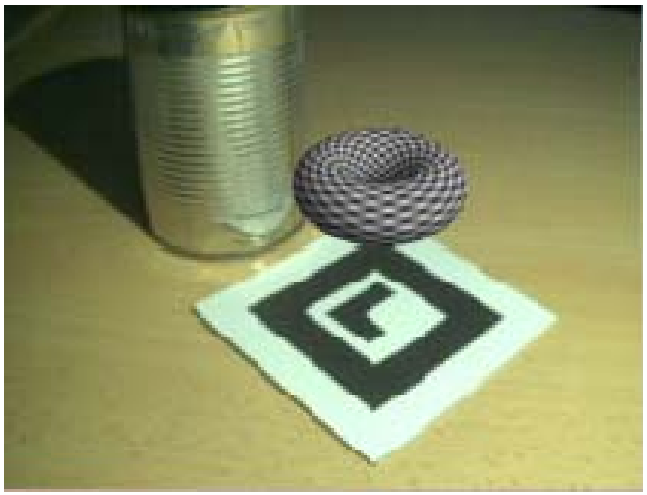
- Simple scene geometry
- Only virtual shadows

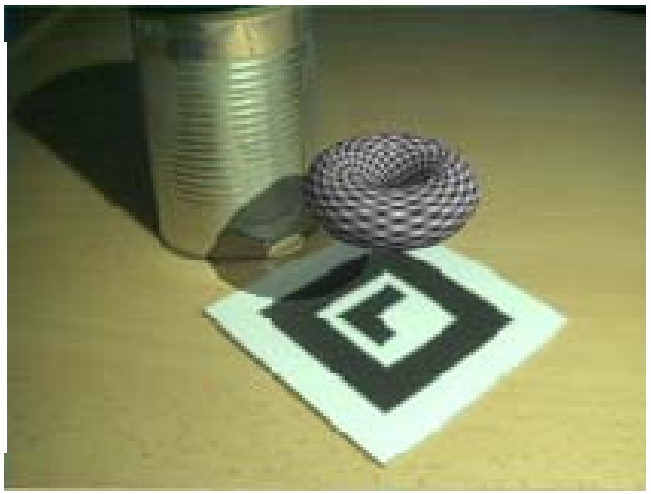


Shadow volume rendering

- Multi-pass shadow volume
  1. Draw real scene from video image
  2. Draw virtual ➔ real shadows
  3. Draw virtual objects
  4. Draw real, virtual ➔ virtual objects

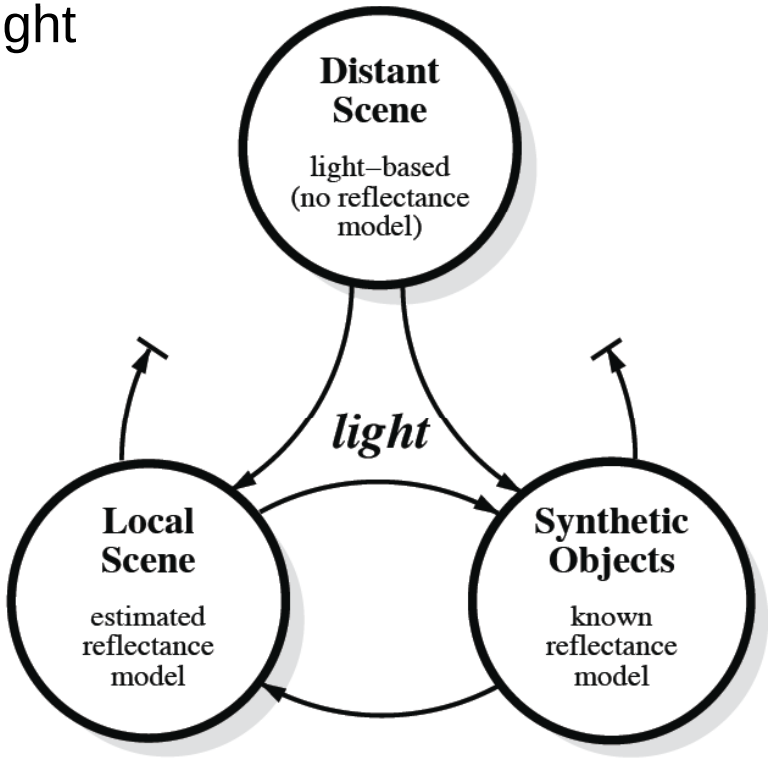






Illumination

- Real scenes have complex illumination
  - More than 1 light source
  - Complex light sources - area, inter-reflection, color
  - Global illumination through indirect light
- Measure illumination
  - Light probe

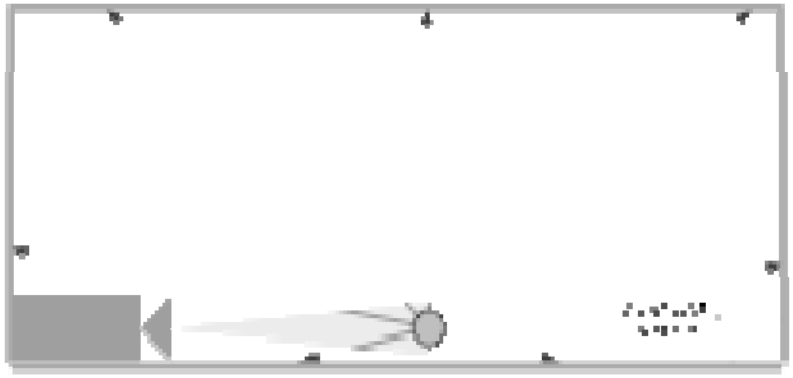


Light probe for environment maps

- Light probe
  - Shiny crome sphere

- Create HDR environment map

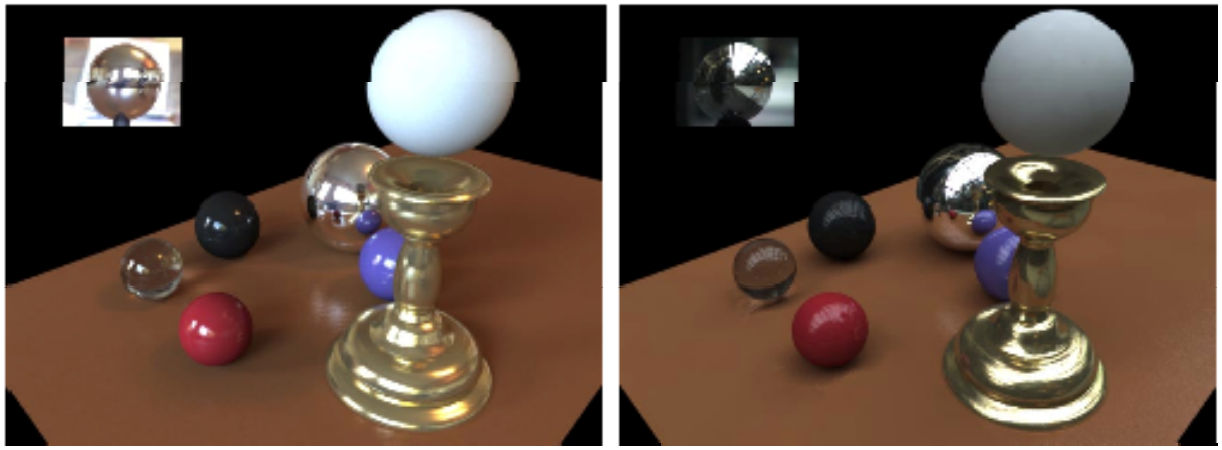
- Captures most directions



[Devebec98]

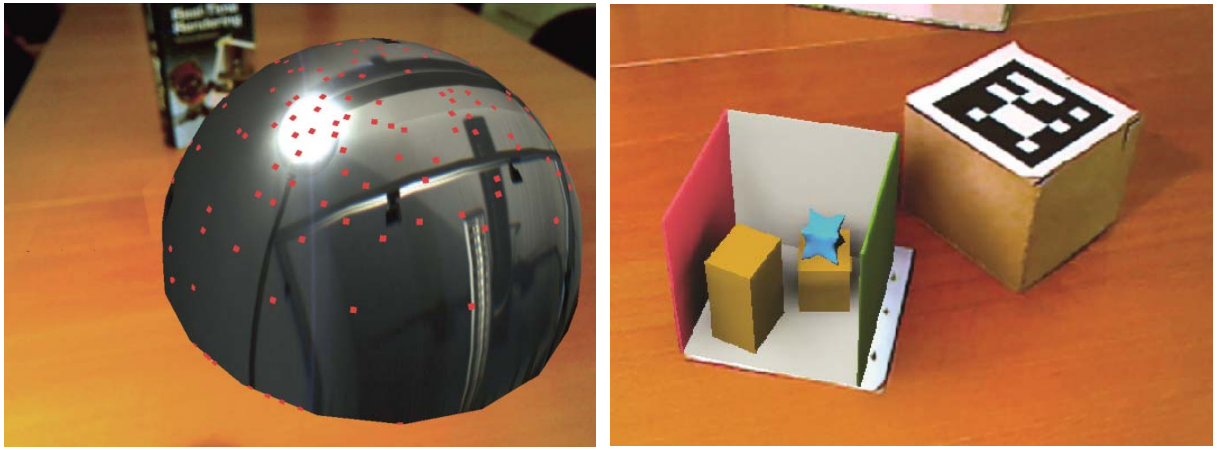
Image-based Lighting and shadows

- Illuminate surfaces with environment map



[Devebec98]

- Approximate with light field and blend multiple shadows

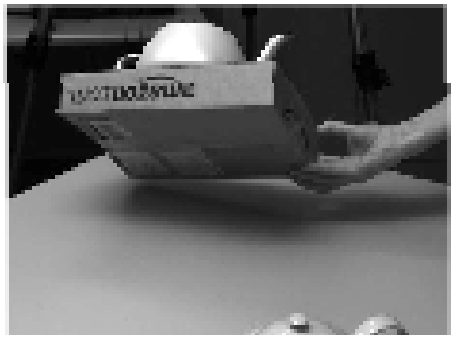


[Knecht10]

Image-based Lighting and shadows

Light acquisition without a sphere



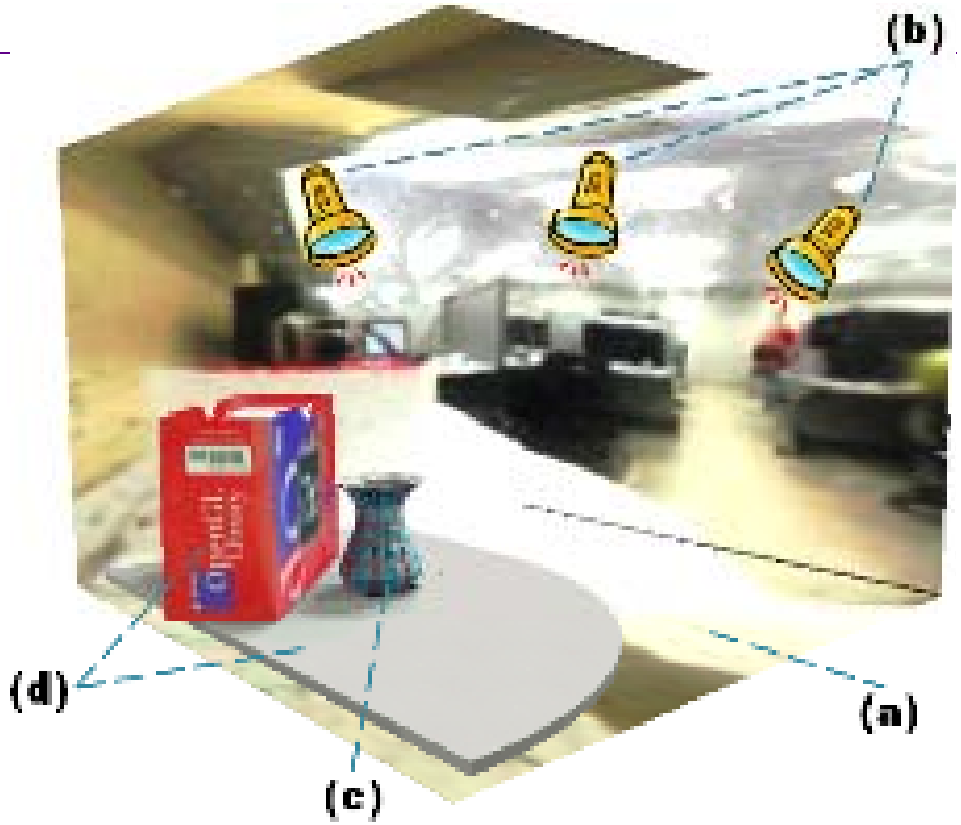



[Pilet06]

- Calibrate multi-camera system
  - planar target with feature point recognition
- On-line and off-line estimation of a light map
  - no light occlusions
  - depends only on normal direction

Now all together

- Interaction between close real and virtual objects
  - light bleeding
  - shadowing
  - refraction
- Render dynamic environment maps per virtual object



(a) Environment map
(b) Light sources
(c) Virtual objects
(d) Impostors

[Pessoa10]

Now all together

Camera Effects

- Small & cheap cameras & lenses
- Many problems
  - Distortion
  - Vignetting
  - Chromatic aberration
  - Softness in corners
  - Bayer pattern artifacts
  - Color compression
  - Arbitrary image processing
  - Motion blur

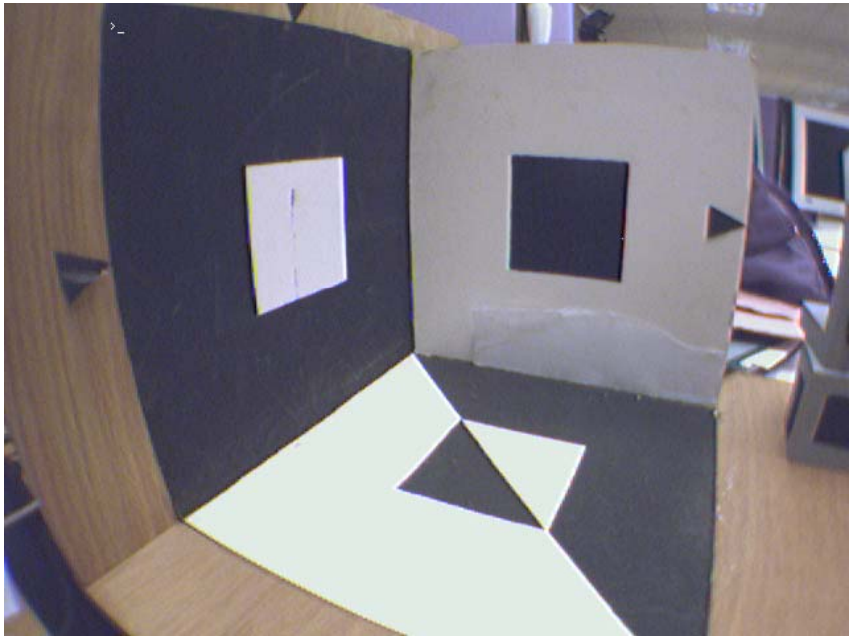
f=2.1 mm, f/2.0, around $10

A terrible image

Graphics don’t match

Lens distortion

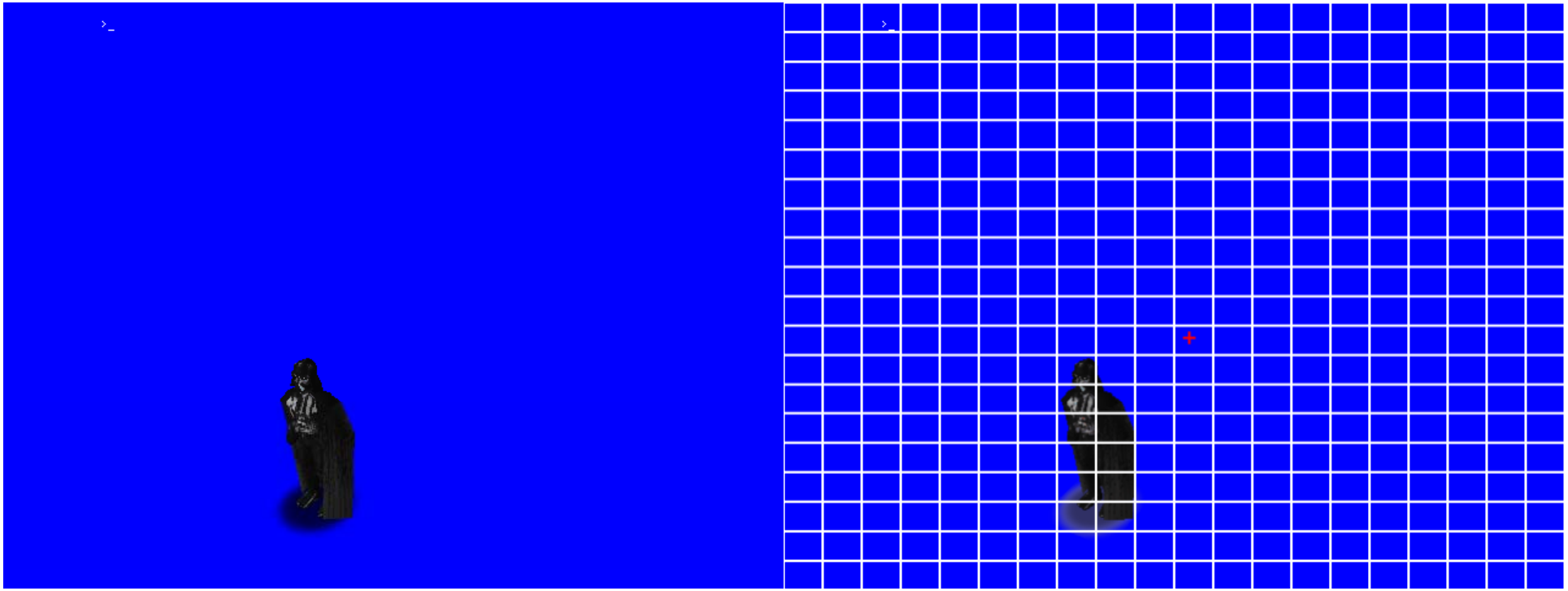
- Wide angle lenses
  - good for tracking
  - not well modeled with a pinhole camera



- Texture-based Approach
  - Watson & Hodges ’95
  - Render undistorted initially
  - Use texture mapping to distort rendered image

- Geometry-based Approach
  - Vertex shader to tweak geometry

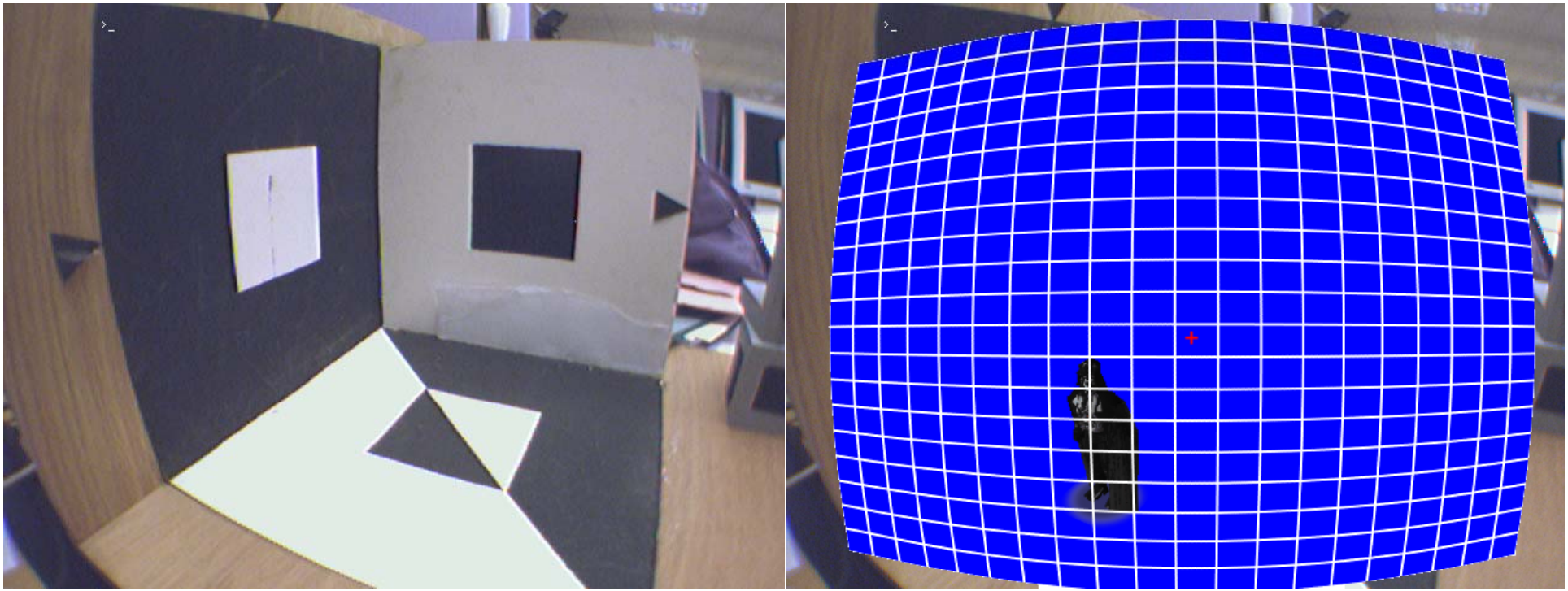
Render un-distorted visuals



[Klein05]

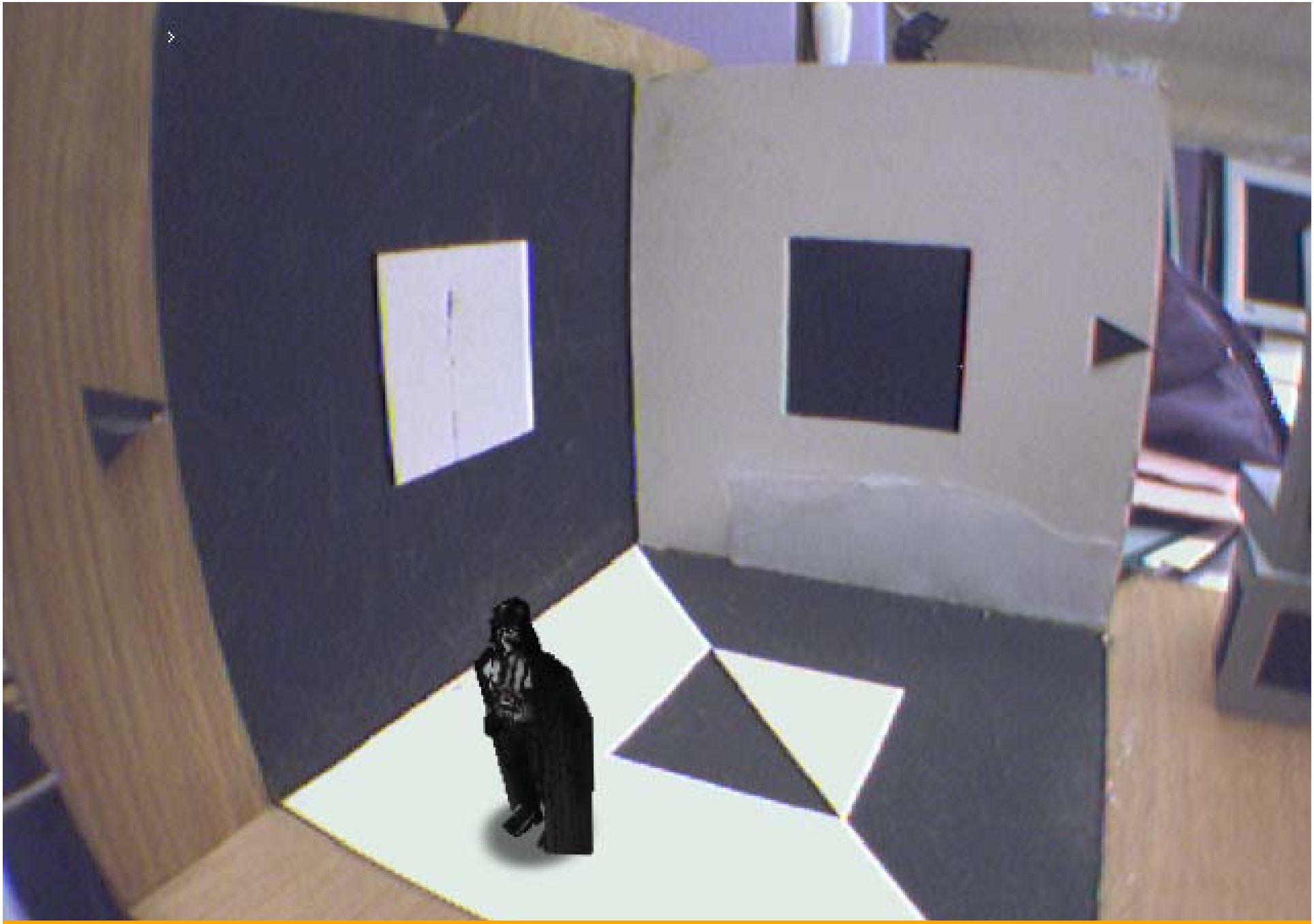
- Render into off screen buffer
- Prepare a grid for texture mapping

Compositing with video background



[Klein05]

- Draw video background
- Overlay distorted texture



Final Image

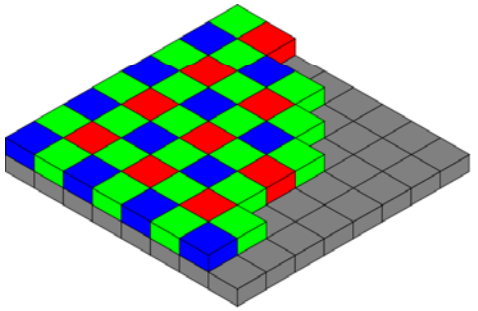
Geometry-based Approach

- Vertex shader to move vertices in projected image
  - Requires geometry subsampling
  - Artifacts at the image edge

More effects

- Lens
  - Vignetting
  - Chromatic aberration
  - Softness in corners
- Image acquisition
  - Antialiasing
  - Sharping
  - Quantization
  - Bayer CFA artifacts
  - Color compression

Sony 1/4” CCD
640x480@30Hz
Bayer mask

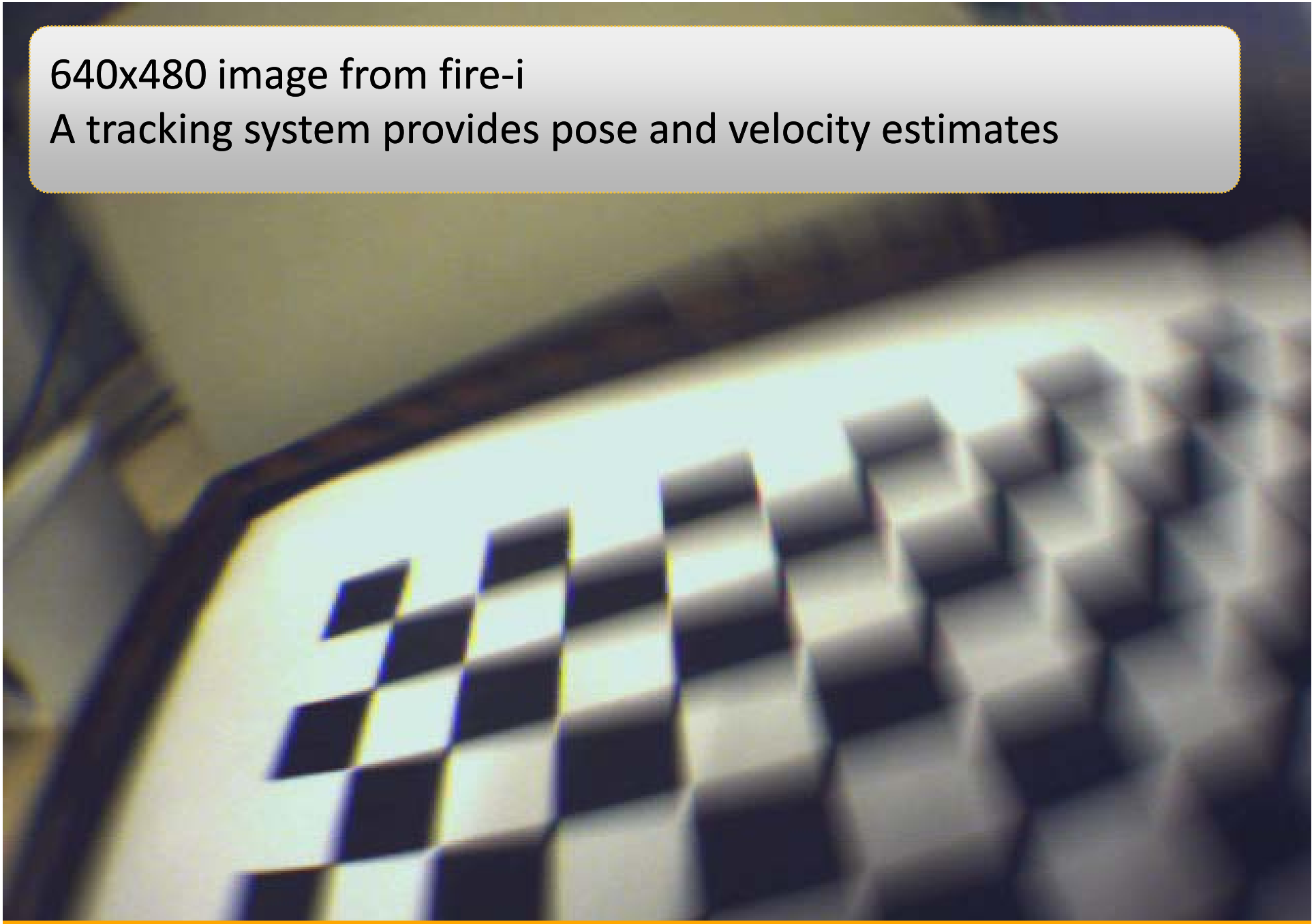


Emulating camera effects

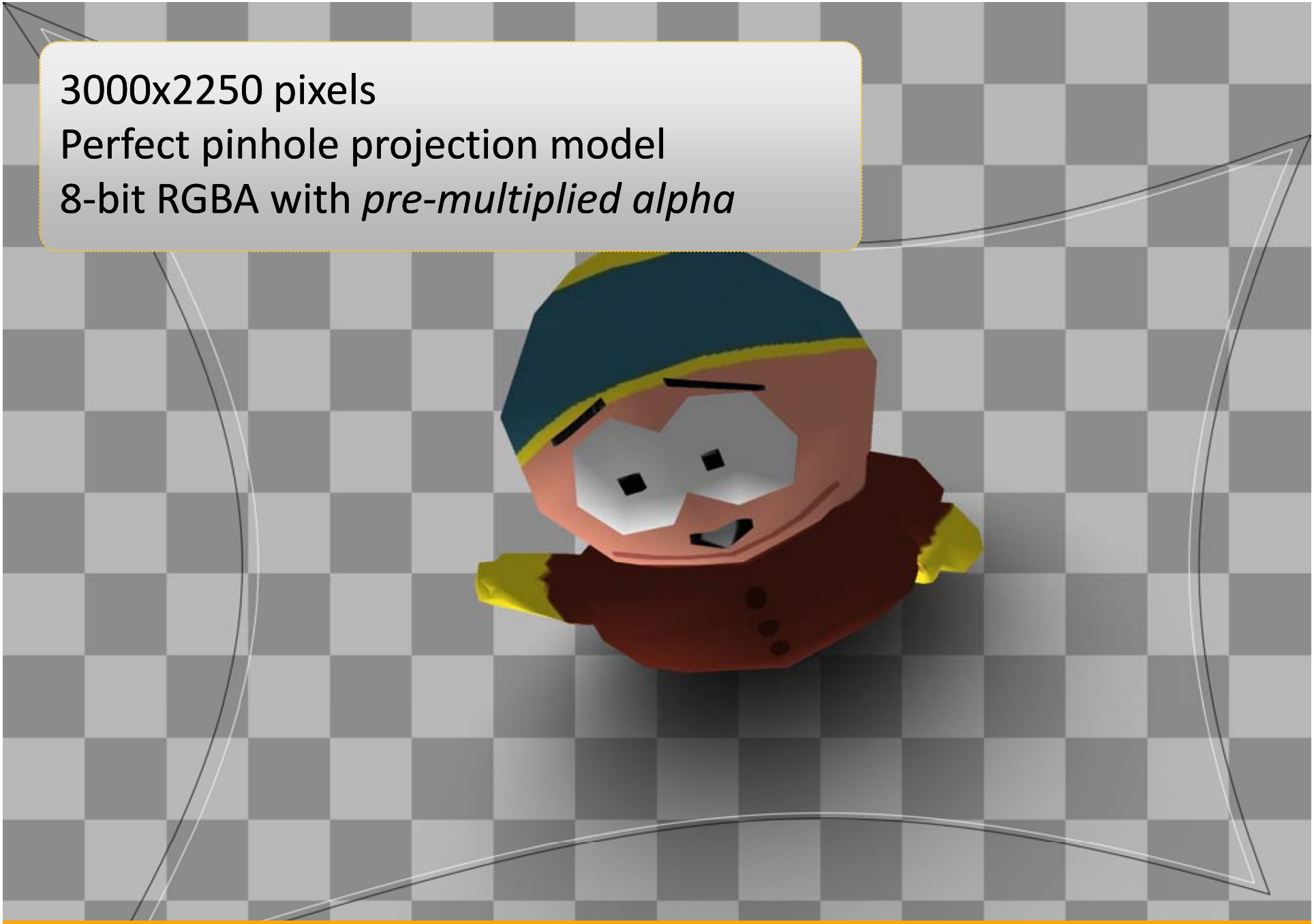
- Some effects require changes to 3D rendering
  - E.g: Depth defocus, proper motion blur
- Expensive !

- Focus on post-processing:
  - Take single RGBA image rendered in OpenGL
  - Modify this image to emulate camera effects
- Limitations
  - No full motion blur

[Klein08]



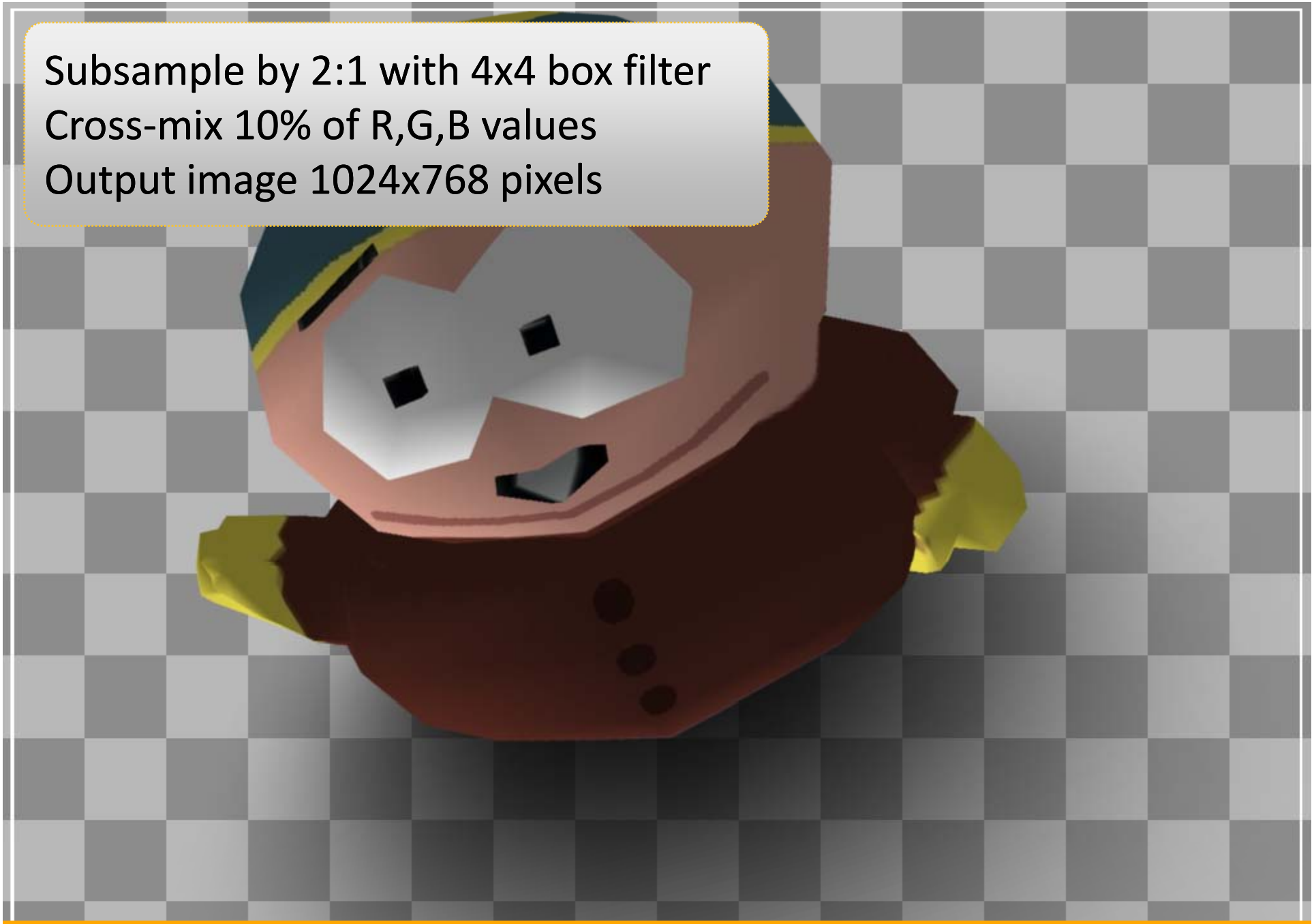
0. Input image



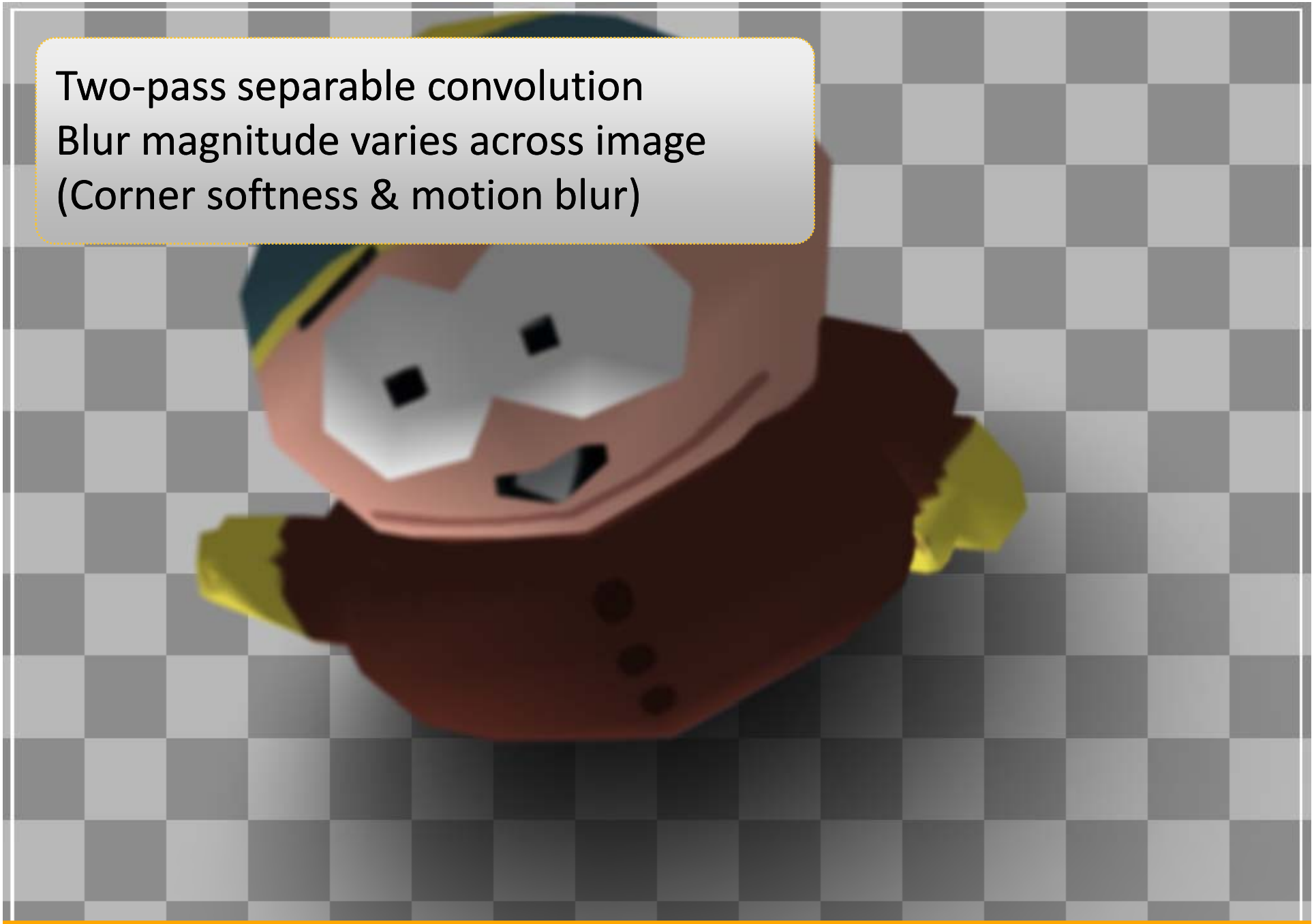
1. Rendered Image

Distort image with texture-mapped grid
Output 2048x1536

2. Apply lens distortion

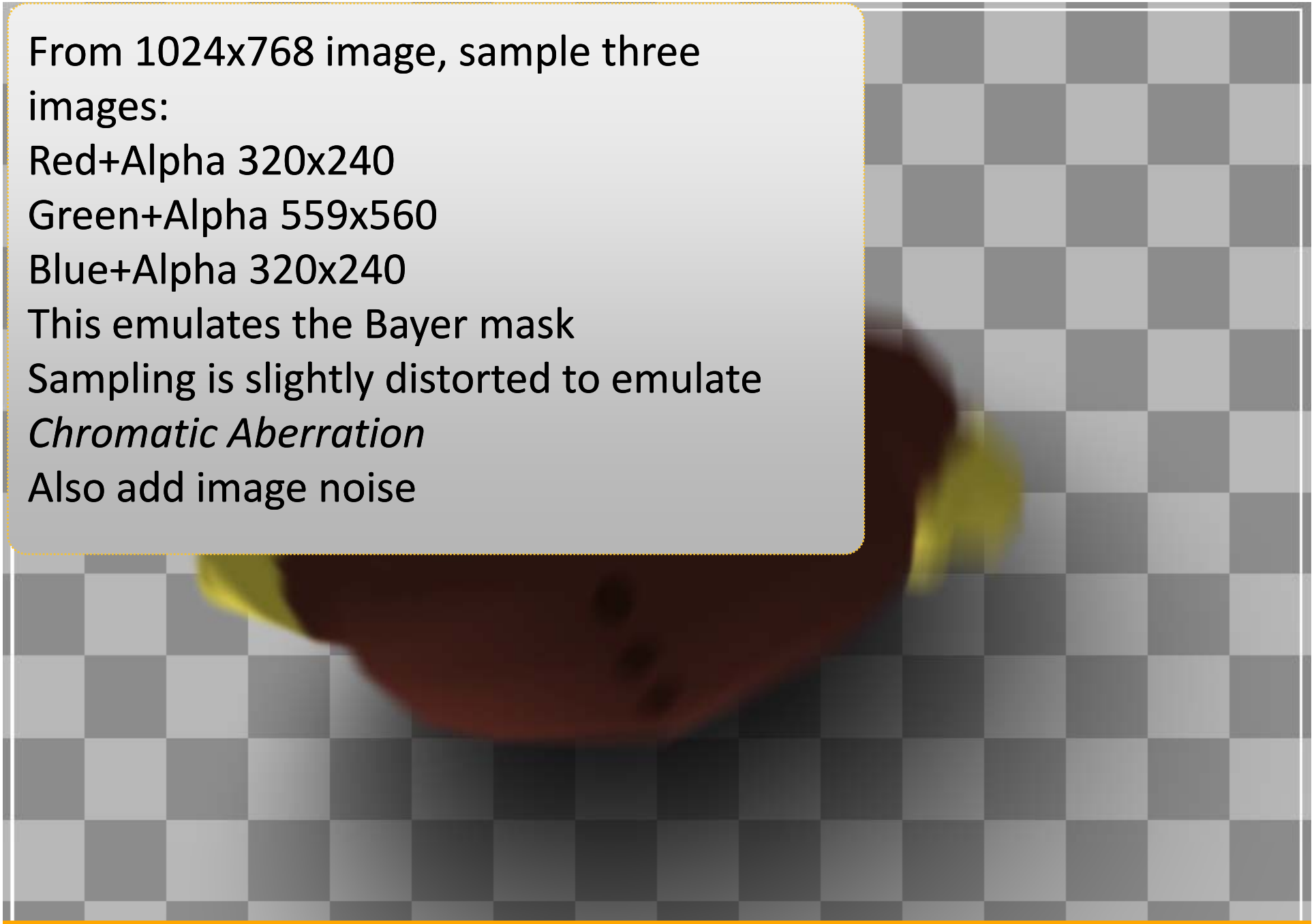


3. Half-sample and colour mix



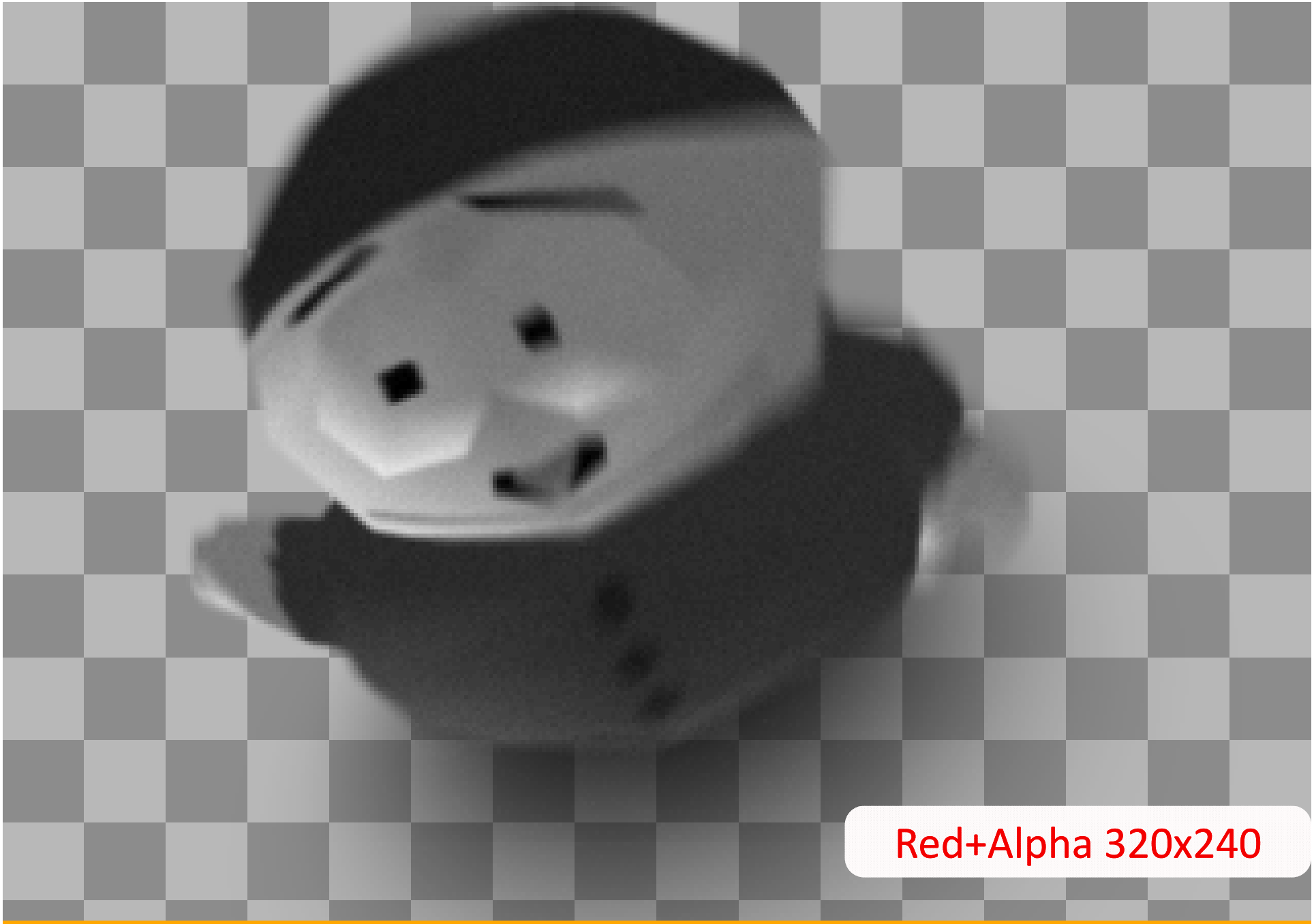
4. 7-tap Gaussian Blur

5. 16-tap motion blur

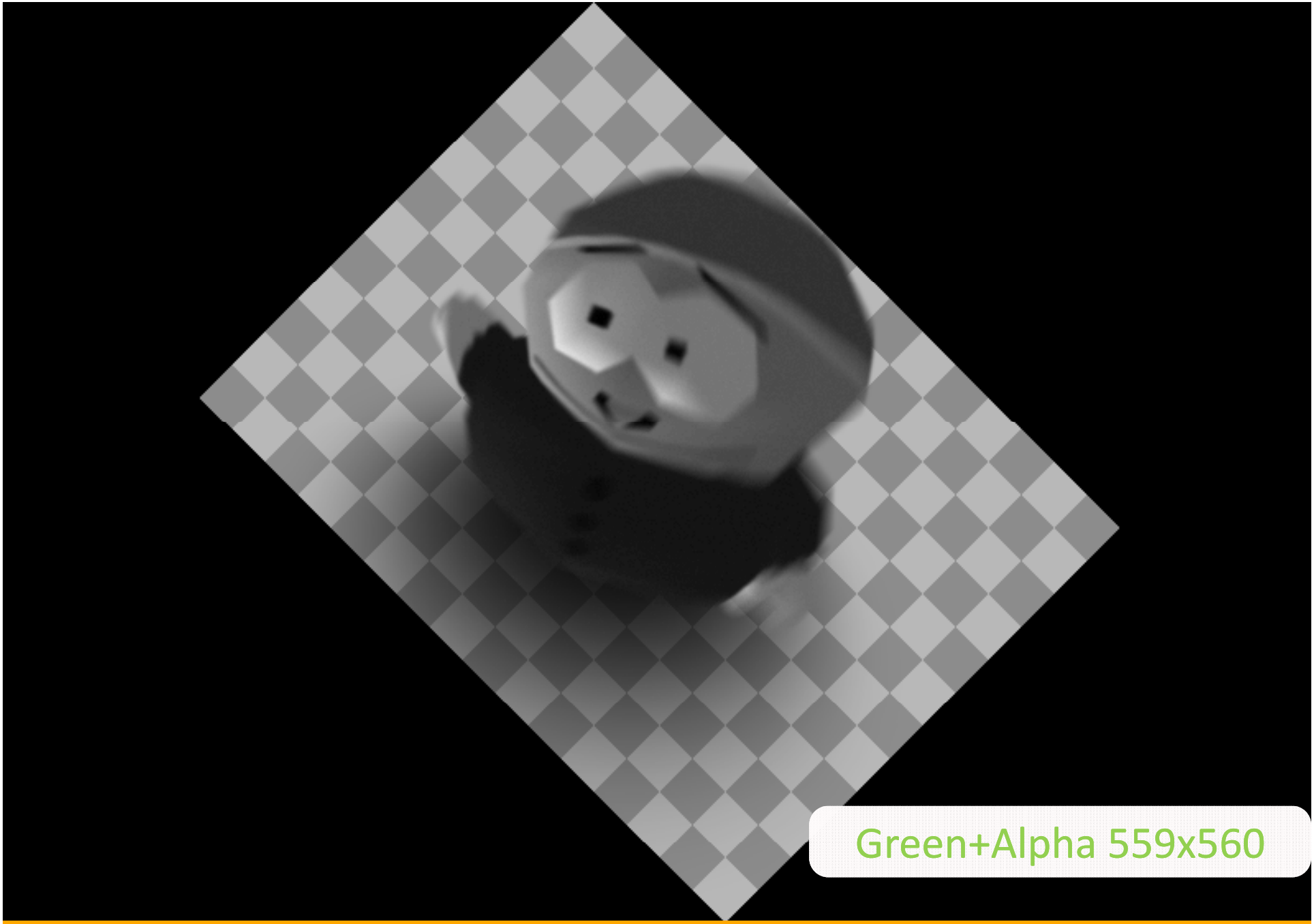


From 1024x768 image, sample three images:
Red+Alpha 320x240
Green+Alpha 559x560
Blue+Alpha 320x240
This emulates the Bayer mask
Sampling is slightly distorted to emulate *Chromatic Aberration*
Also add image noise

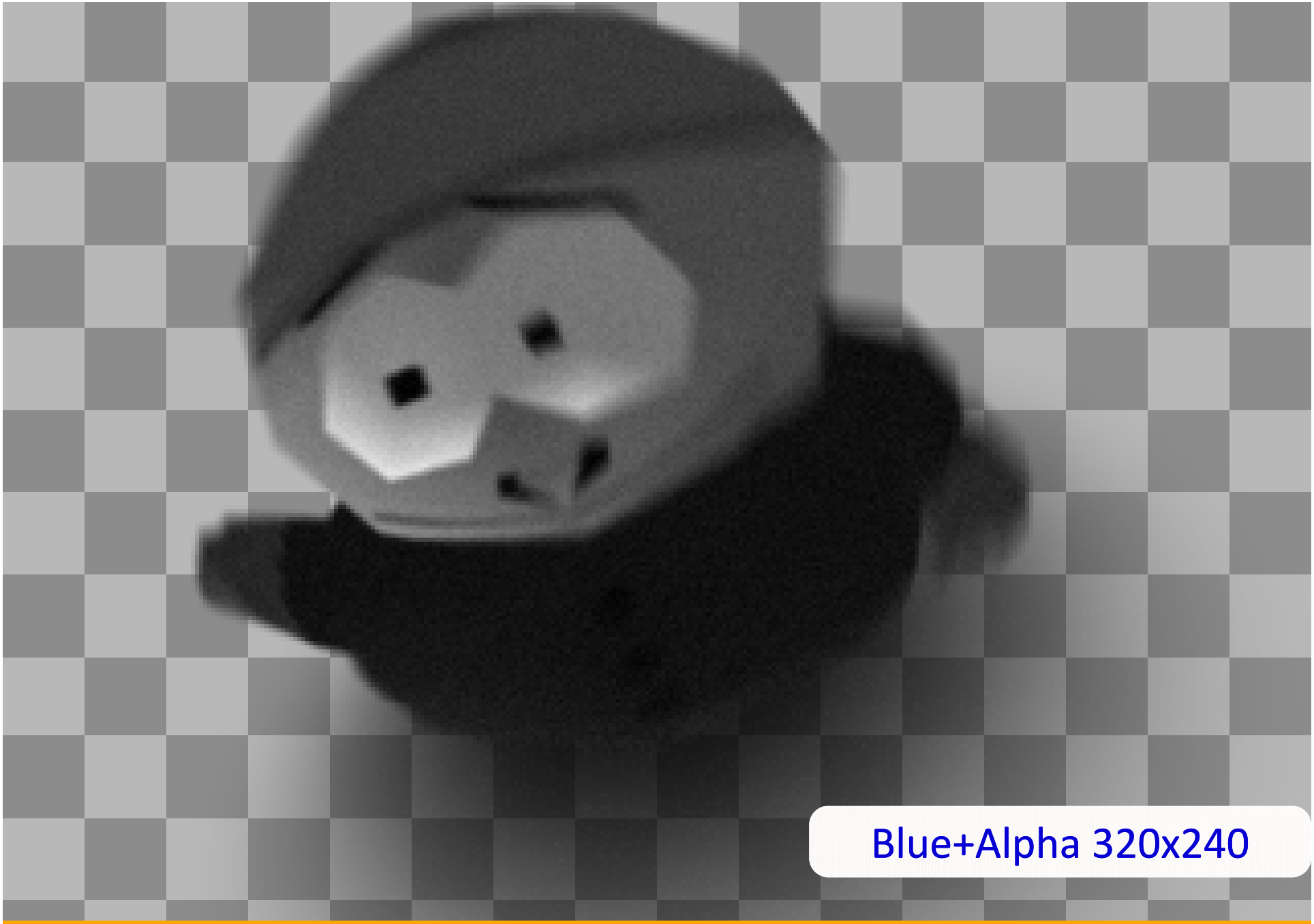
6. Bayer sampling



6. Bayer sampling

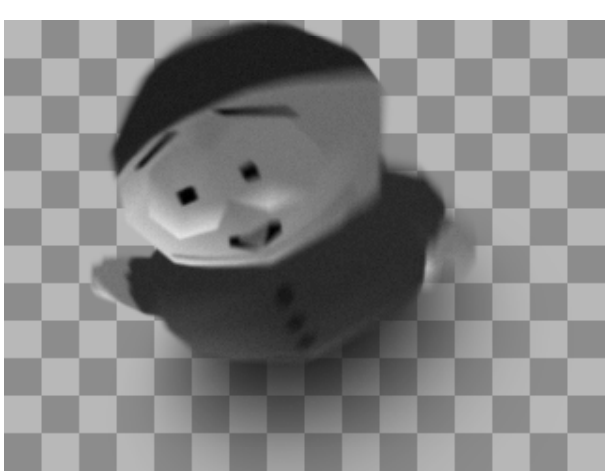
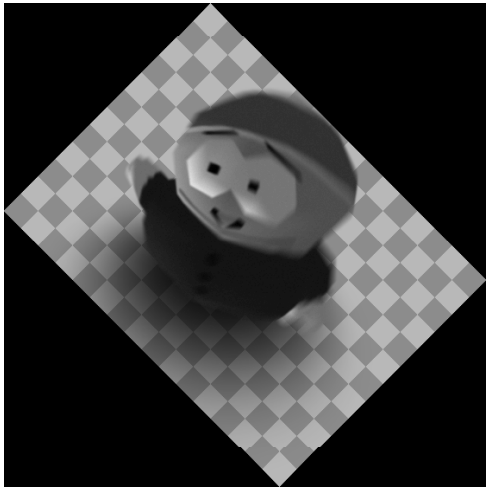
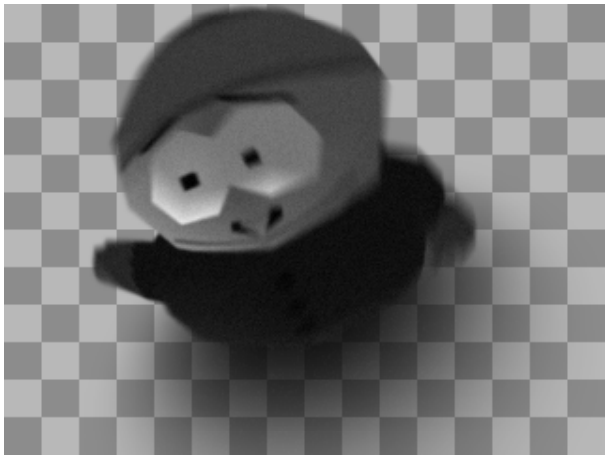


6. Bayer sampling



6. Bayer sampling

Bayer Samples

- Raw bayer image available?
  Blend now, then de-Bayer - done!

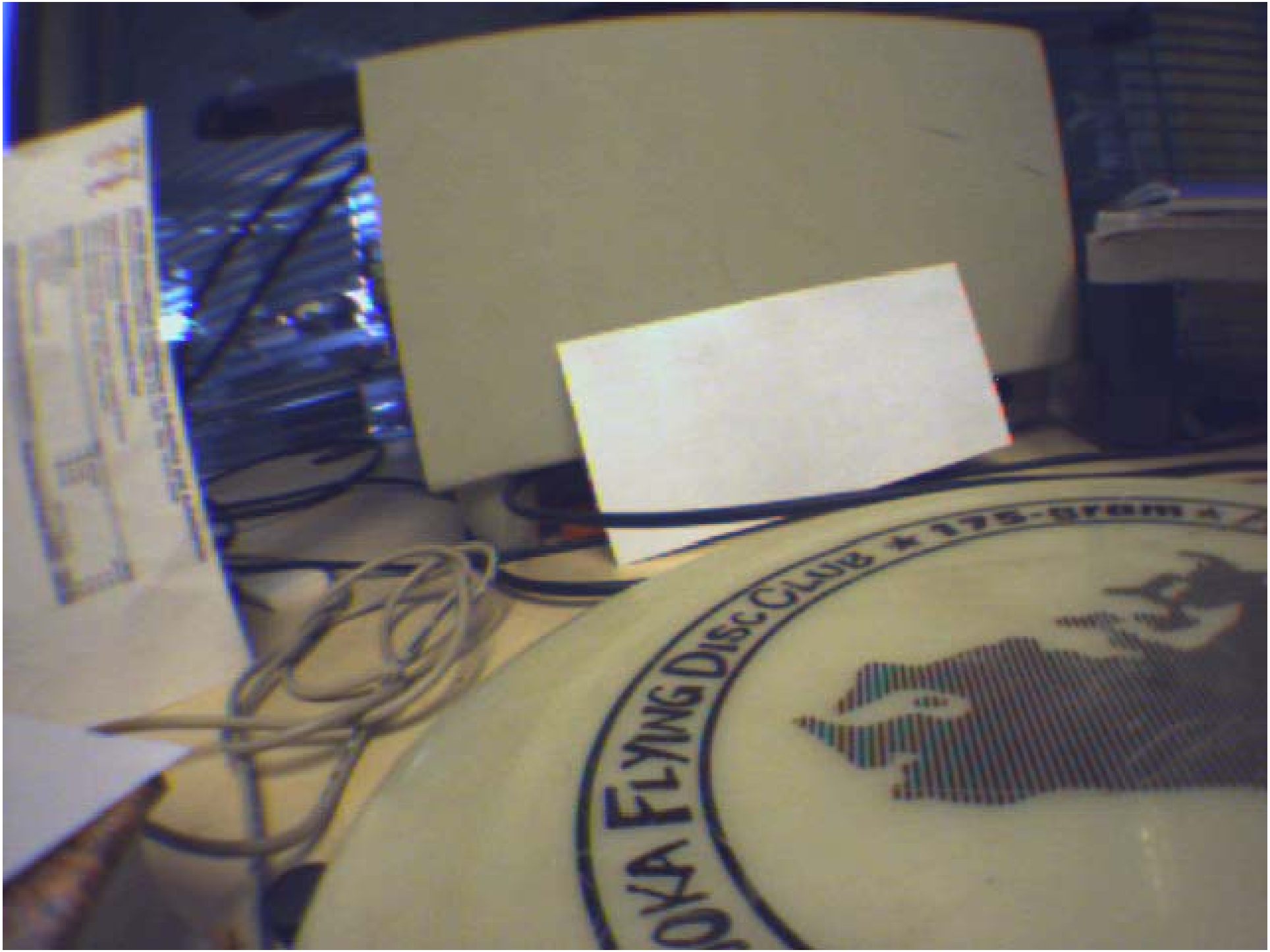
YUV Composite

Chroma Squash

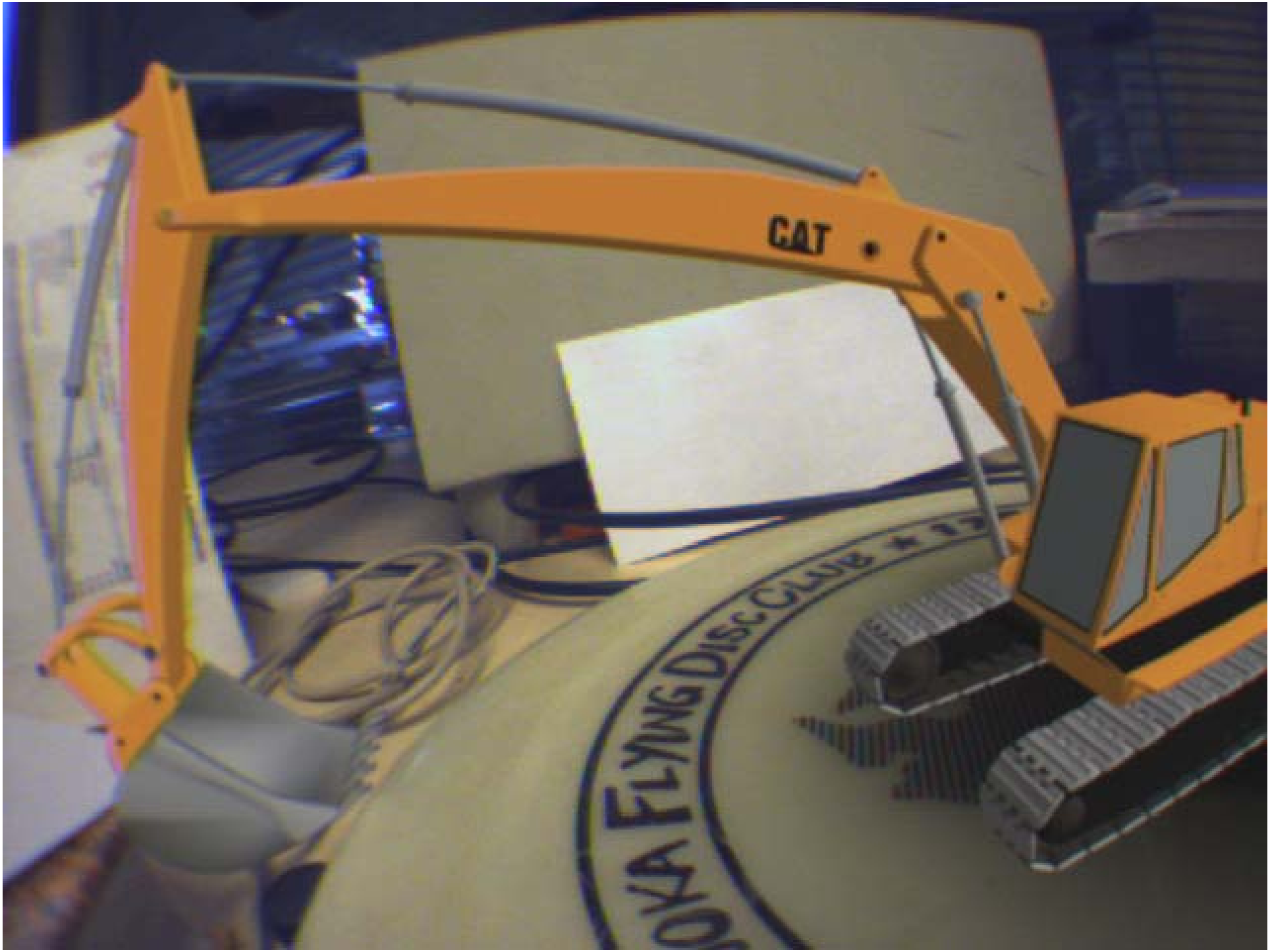
Chroma Squash

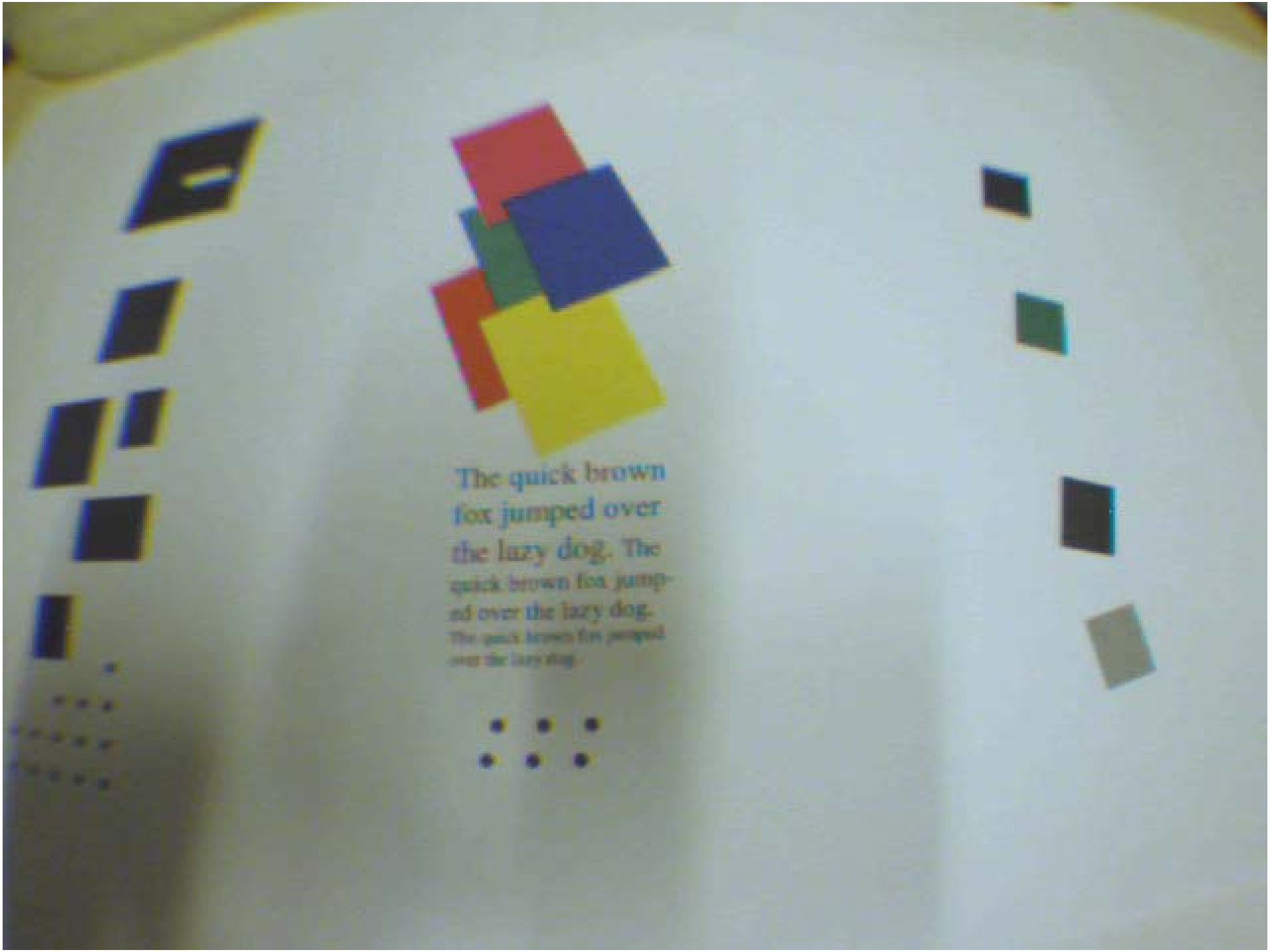
Finished Composite

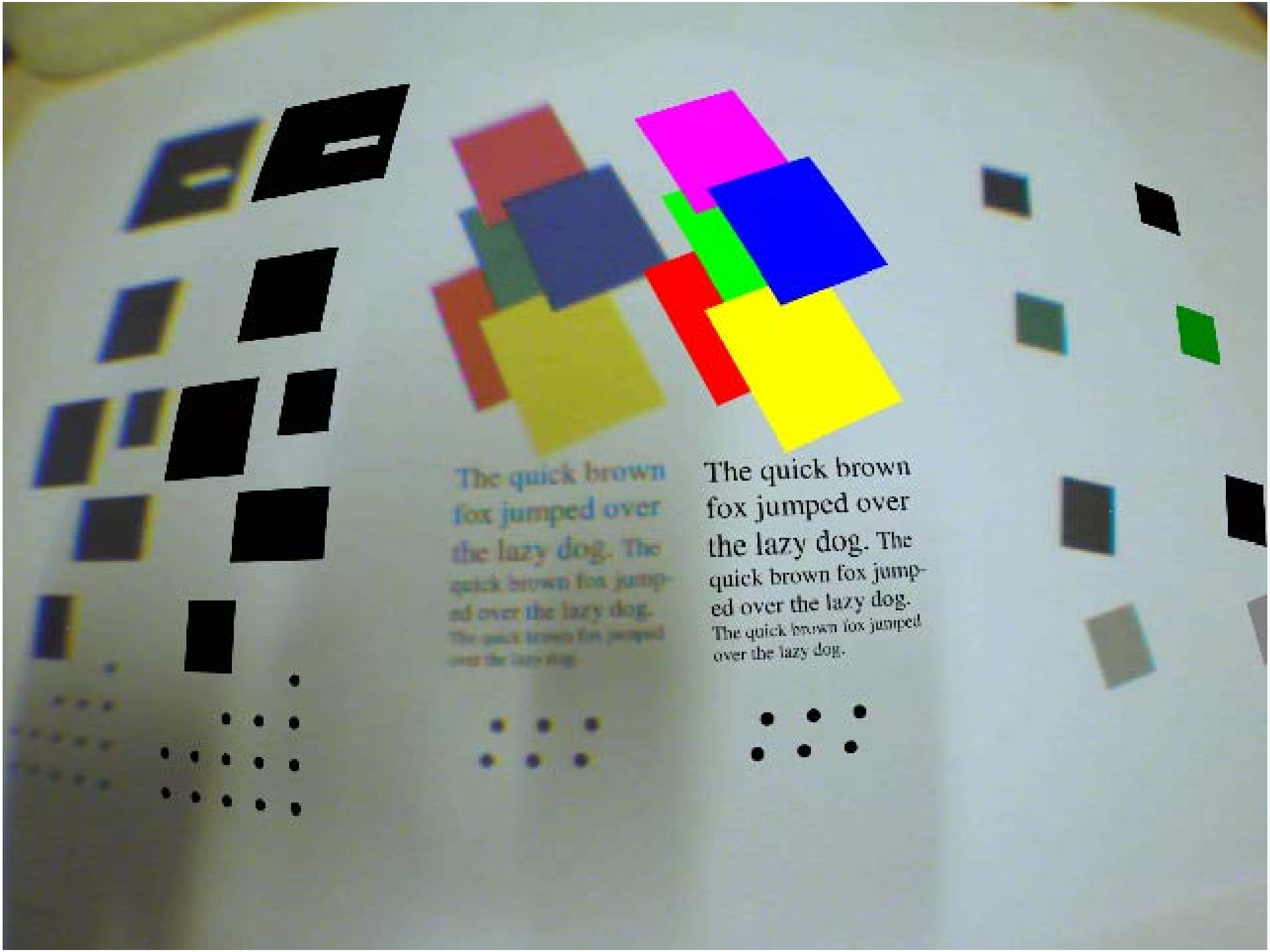
Old Method

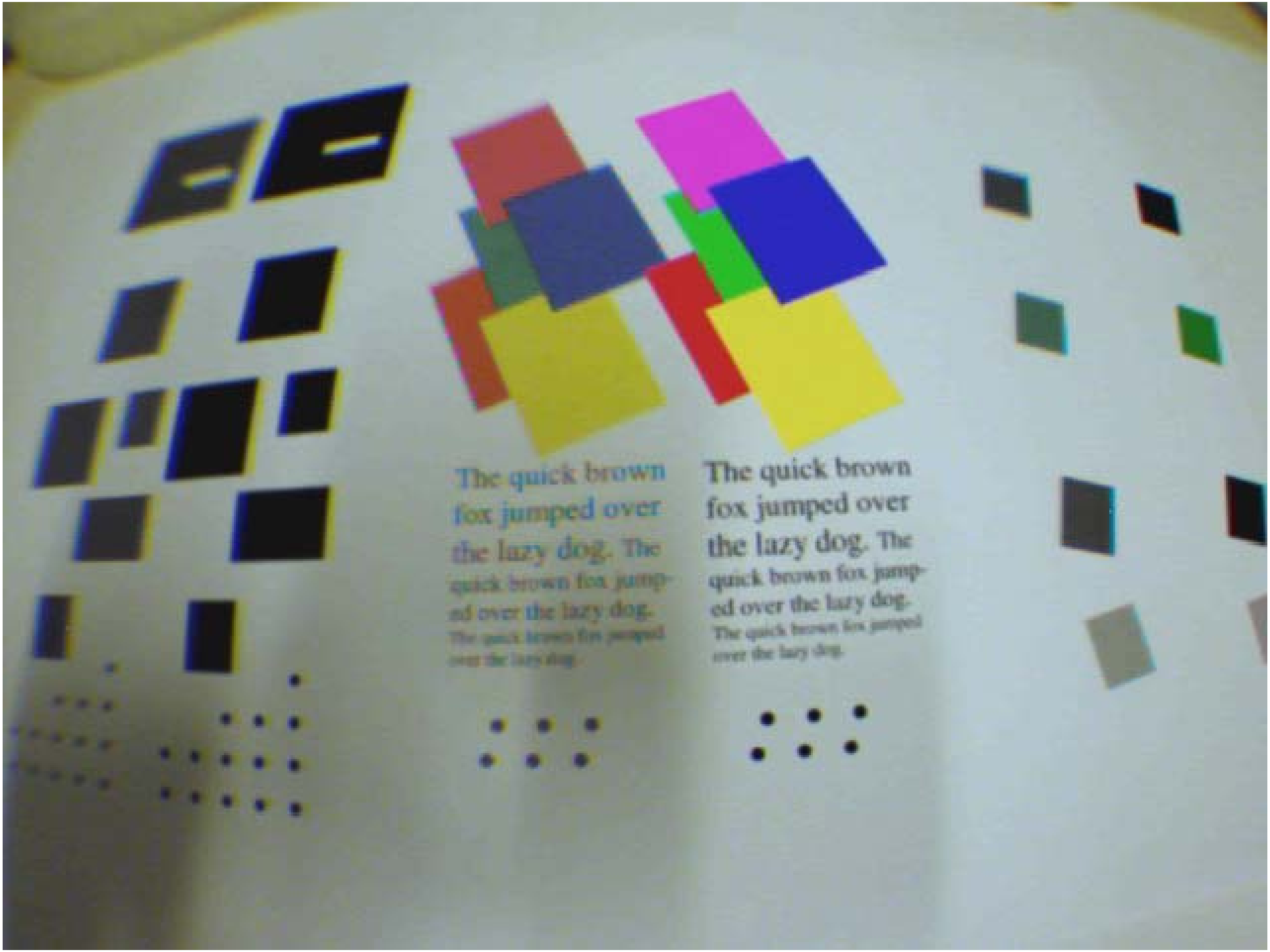

OKA FLYING DISC CLUB


CAT


CAT


The quick brown fox jumped over the lazy dog. The quick brown fox jumped over the lazy dog.
The quick brown fox jumped over the lazy dog.


The quick brown fox jumped over the lazy dog. The quick brown fox jumped over the lazy dog. The quick brown fox jumped over the lazy dog.
The quick brown fox jumped over the lazy dog. The quick brown fox jumped over the lazy dog. The quick brown fox jumped over the lazy dog.

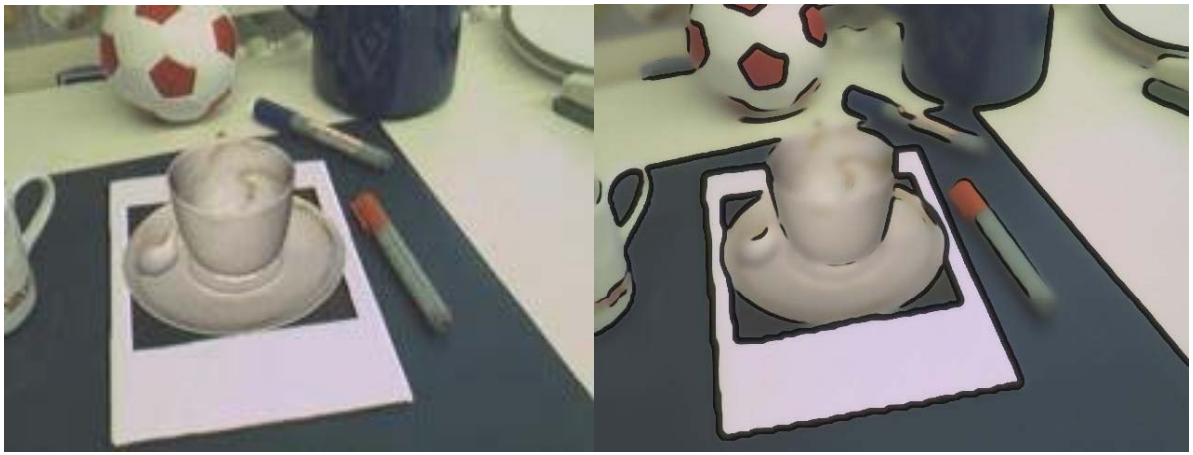

The quick brown fox jumped over the lazy dog. The quick brown fox jumped over the lazy dog.
The quick brown fox jumped over the lazy dog.
The quick brown fox jumped over the lazy dog. The quick brown fox jumped over the lazy dog.
The quick brown fox jumped over the lazy dog.

Visual Effects

- Realism difficult ➔ Change reality instead
- Diminished reality
  - Hide real objects
  - Instrumentations

[Mann01]

- Artistic effects
  - Cartoon style
  - Abstraction



[Fischer06]

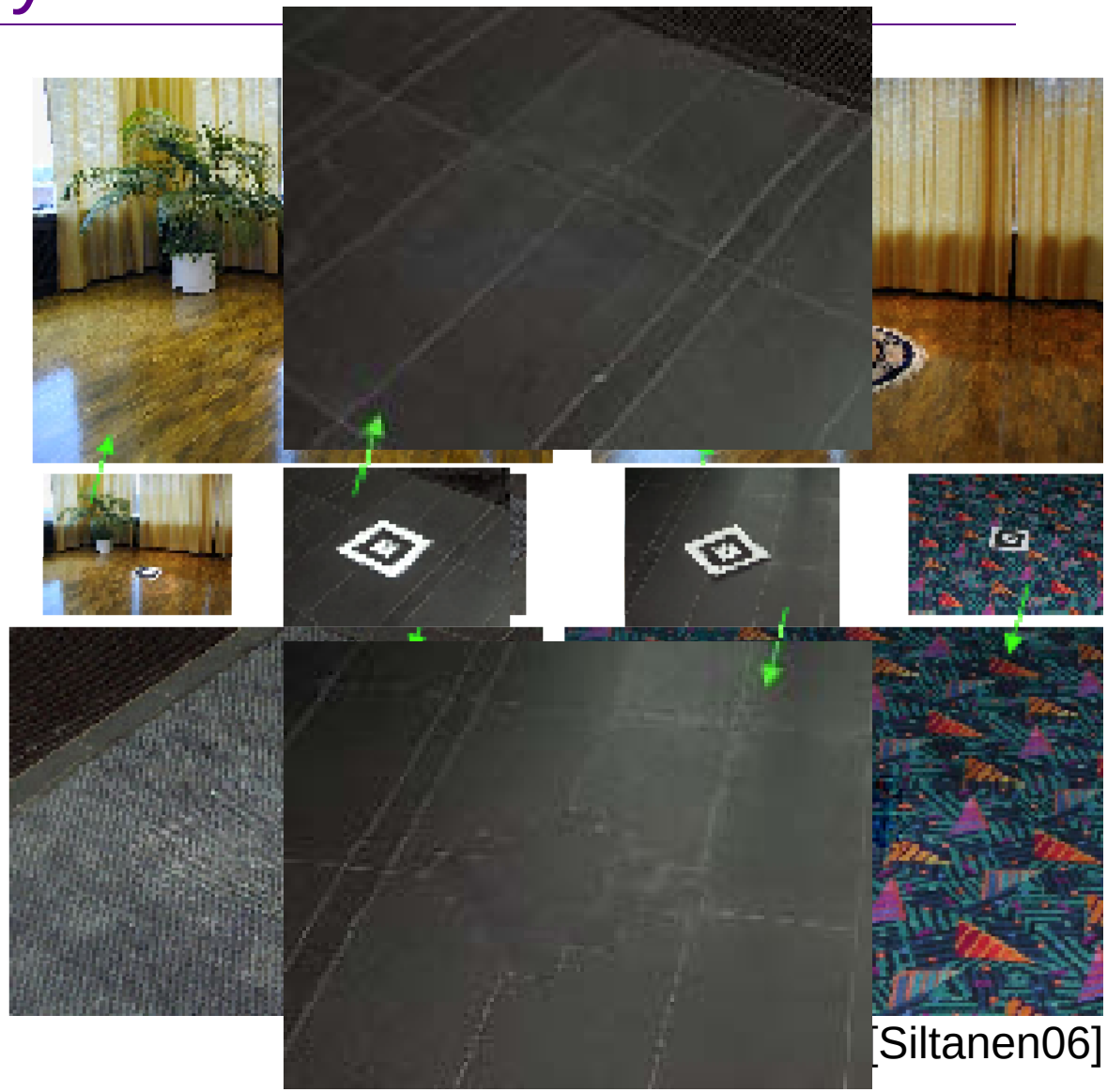
Diminished reality

- Remove fiducial markers
- Texture synthesis from area around the marker
- Repeats band of video background

[Siltanen06]

Diminished reality

- Remove fiducial markers
- Texture synthesis from area around the marker
- Repeats band of video background



[Siltanen06]

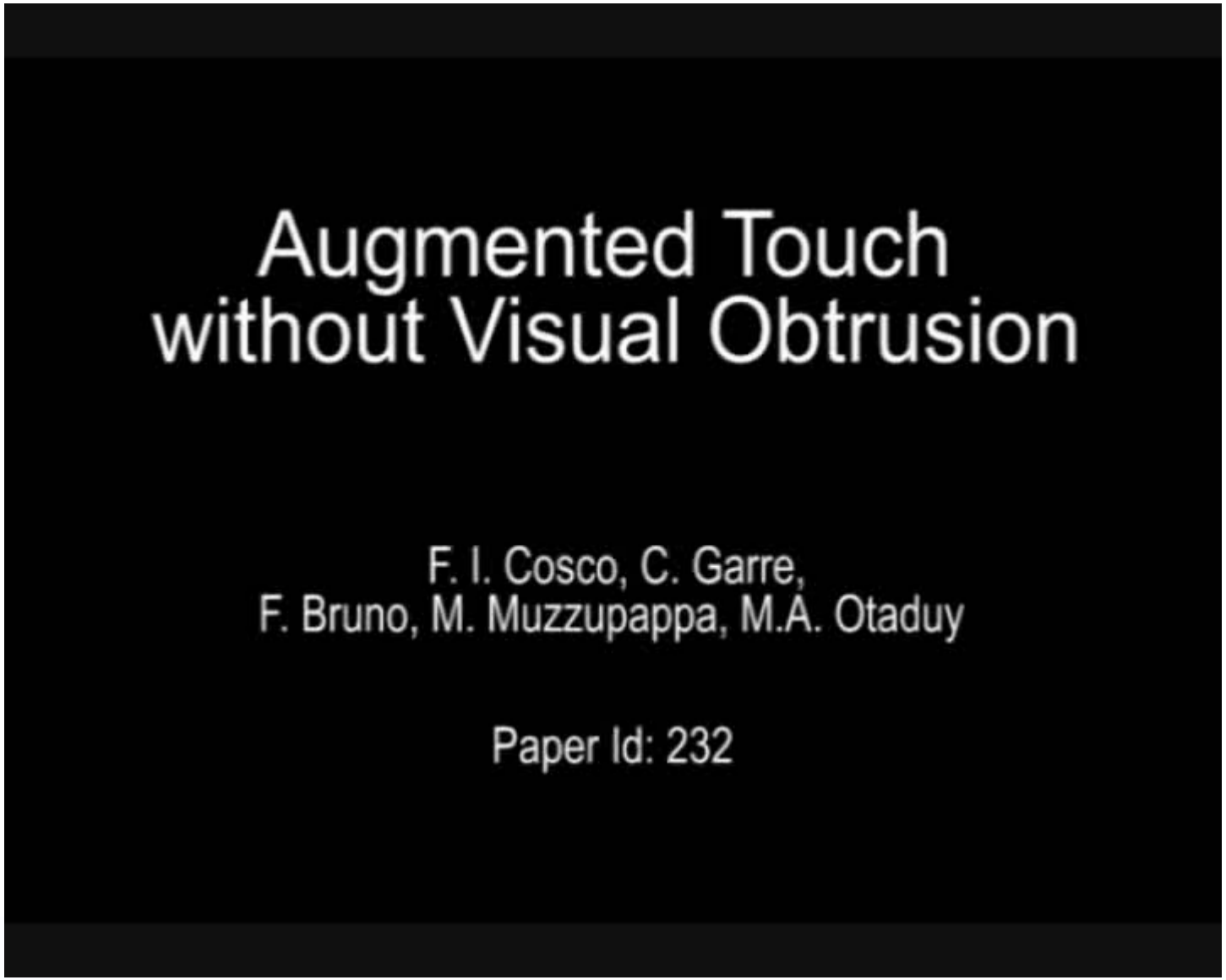
Diminished Reality

- Block/remove objects in the real environment

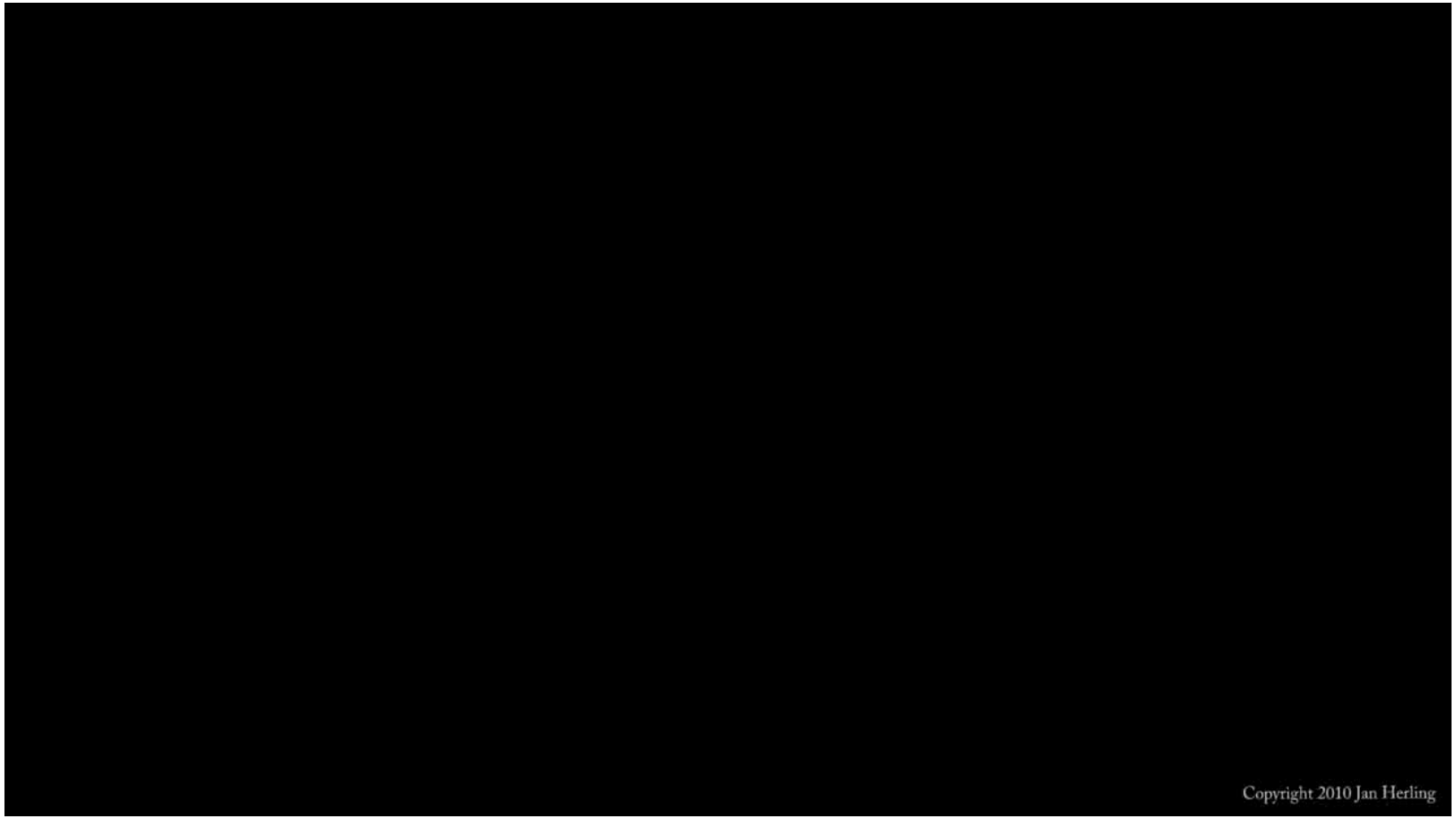
- Haptic device removed with image-based rendering

[Cosco09]

Diminished Reality



Diminished reality



Stylized Augmented Reality

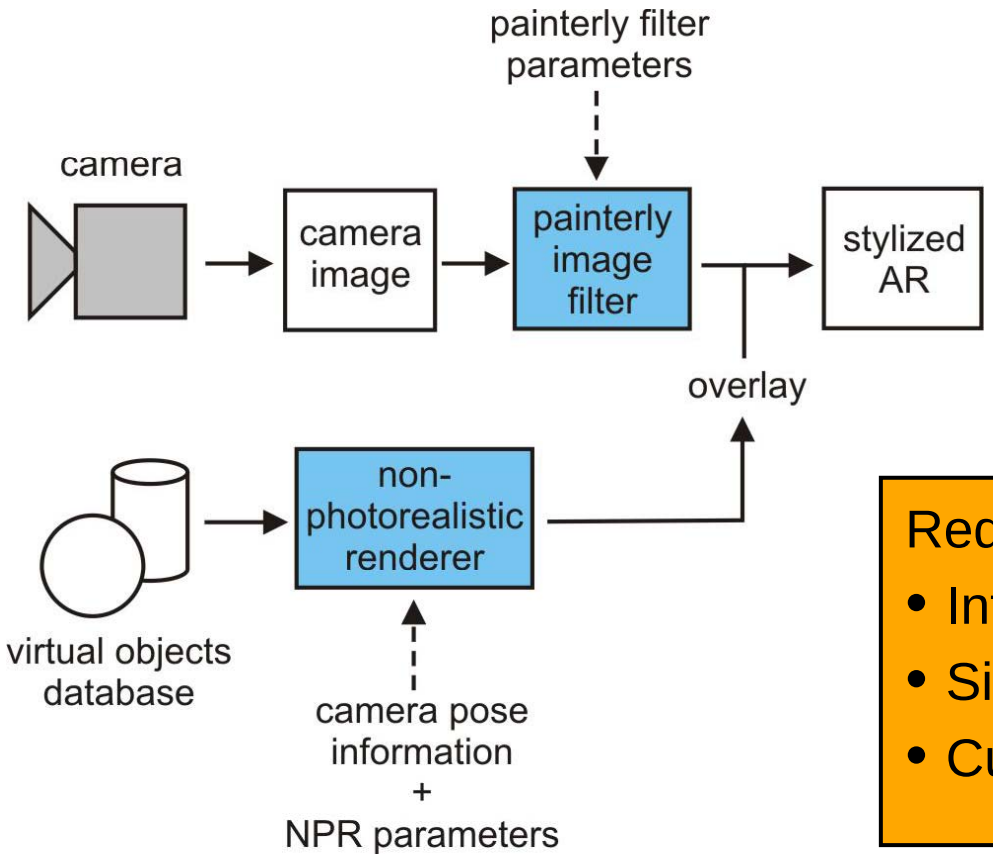
- Non-photorealistic display of both the camera image and virtual objects.
- ➔ Adapt levels of realism

Conventional AR

Stylized Augmented Reality

Stylized AR Pipeline

- Cartoon-like stylization of augmented reality images

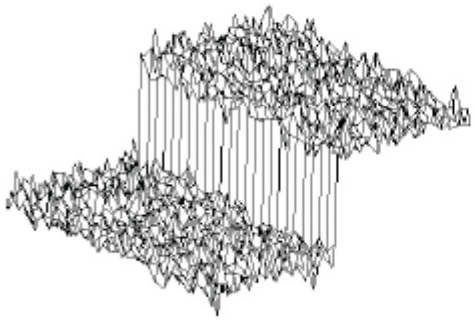


Requirements:

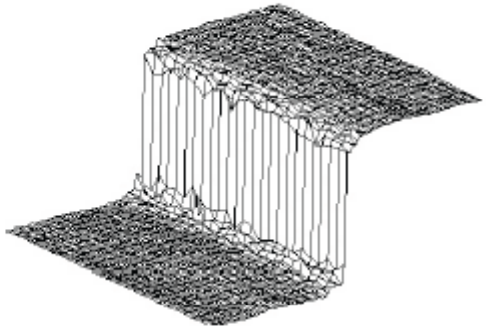
- Interactive frame rate
- Similar-looking output images
- Customizable

Color Segmentation

- Aim: Smooth similarly colored regions, but preserve high contrast edges
- Bilateral filtering [Tomasi and Manduchi 1998]



Original pixel values



Filtered image

Original

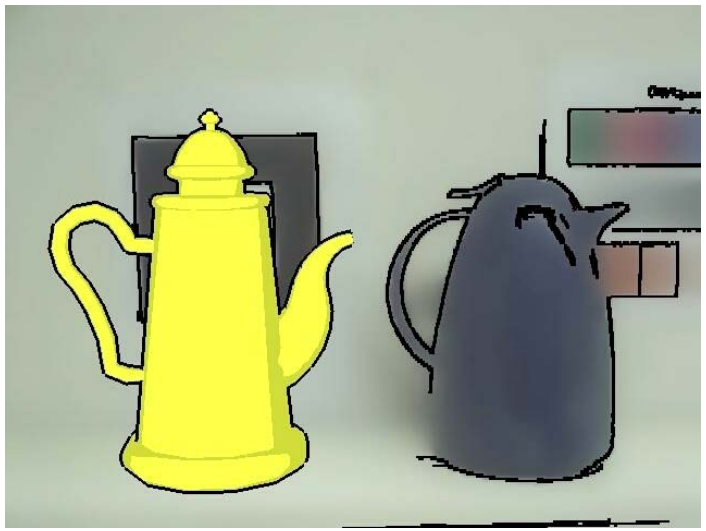
Filtered
[Univ. Edinburgh]

- Takes spatial distance and signal difference into account

Non-Photorealistic Renderer

- Cartoon-like rendering of virtual objects
- Similarity to processed camera image
- Uniformly colored regions enclosed by thick silhouettes
- Two-pass rendering algorithm [Lander 2000]

Conventional



Stylized

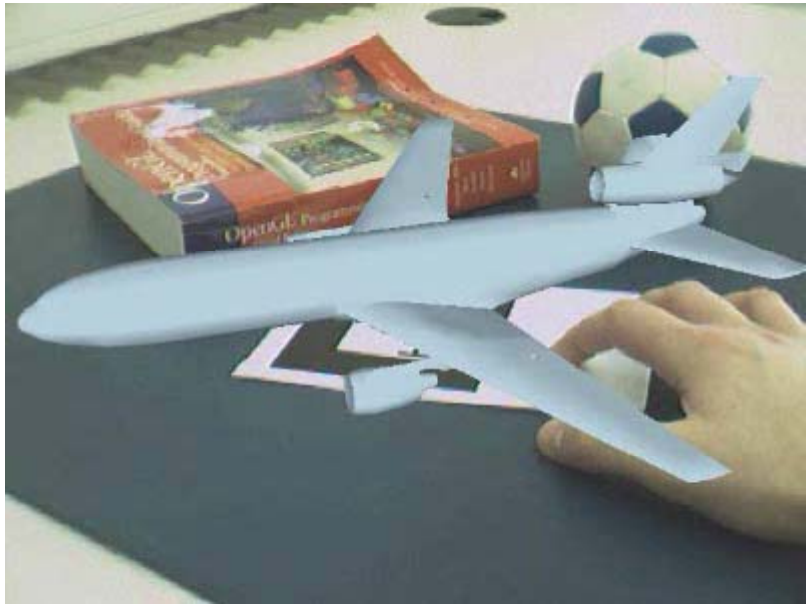
Non-Photorealistic Renderer

Stylized Augmented Reality

"Cartoon-like" Stylization
with virtual bench model

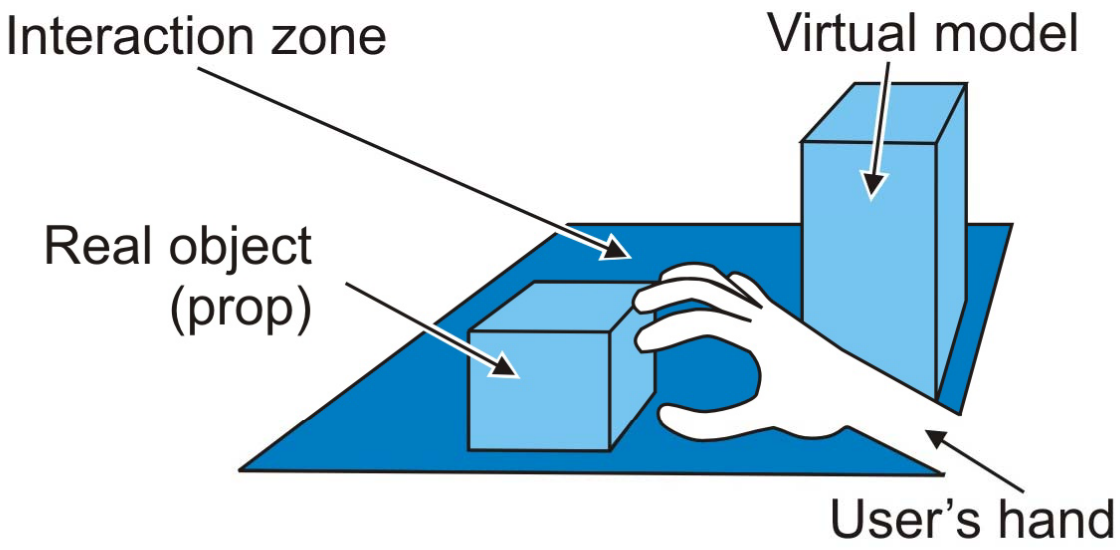
Reality Tooning

- Redesigned algorithm: Pure postprocessing approach
- Implementation on the GPU (OpenGL Shading Language)
- Typically more than 25 fps
- Better visual quality

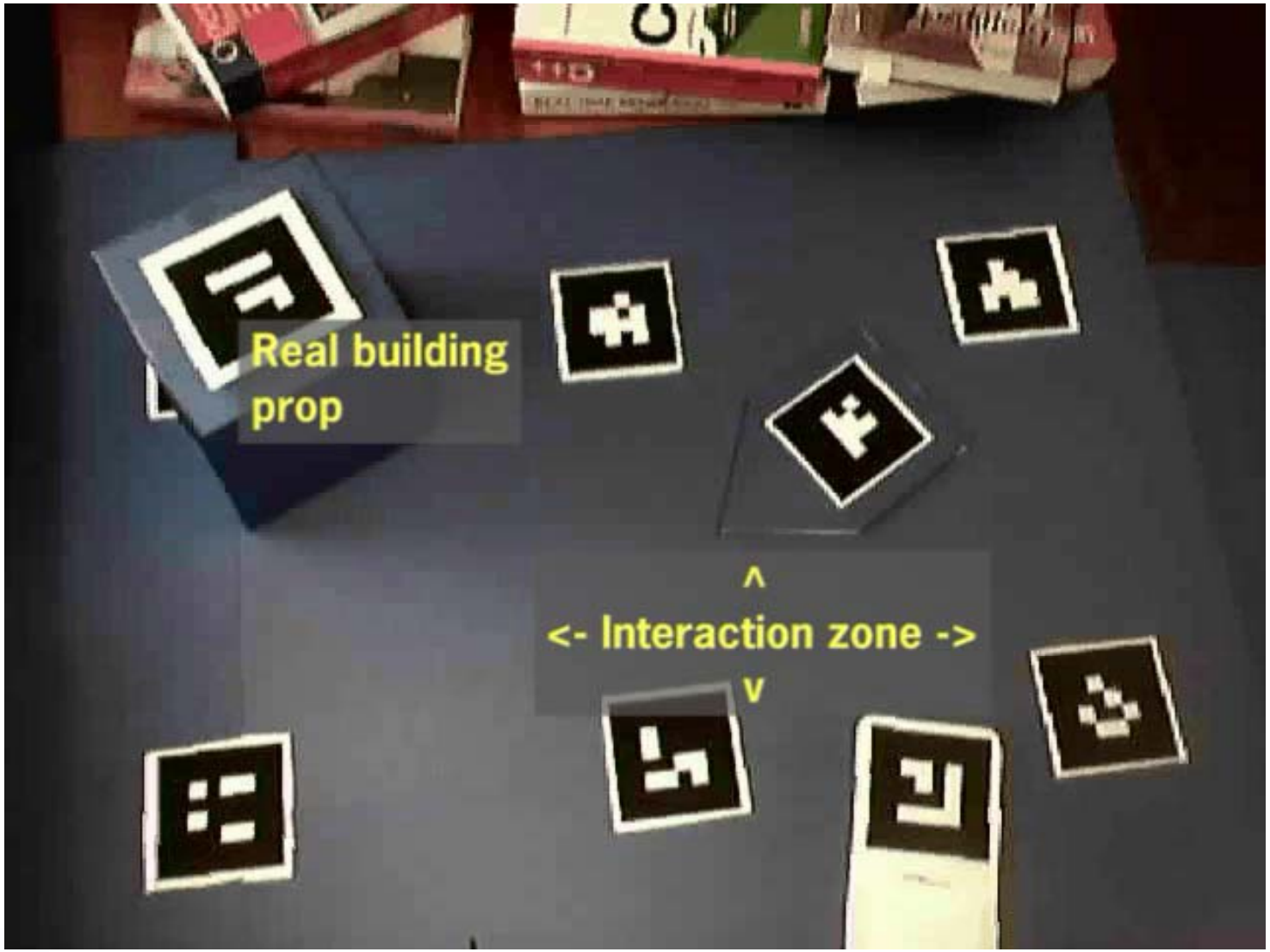


Selective Stylization

- Problem: Entire image is stylized
  - Dissociation of user from observed scene?
  - Why have AR at all ?
- Idea: Select regions where stylization is useful
  - Depends on application case



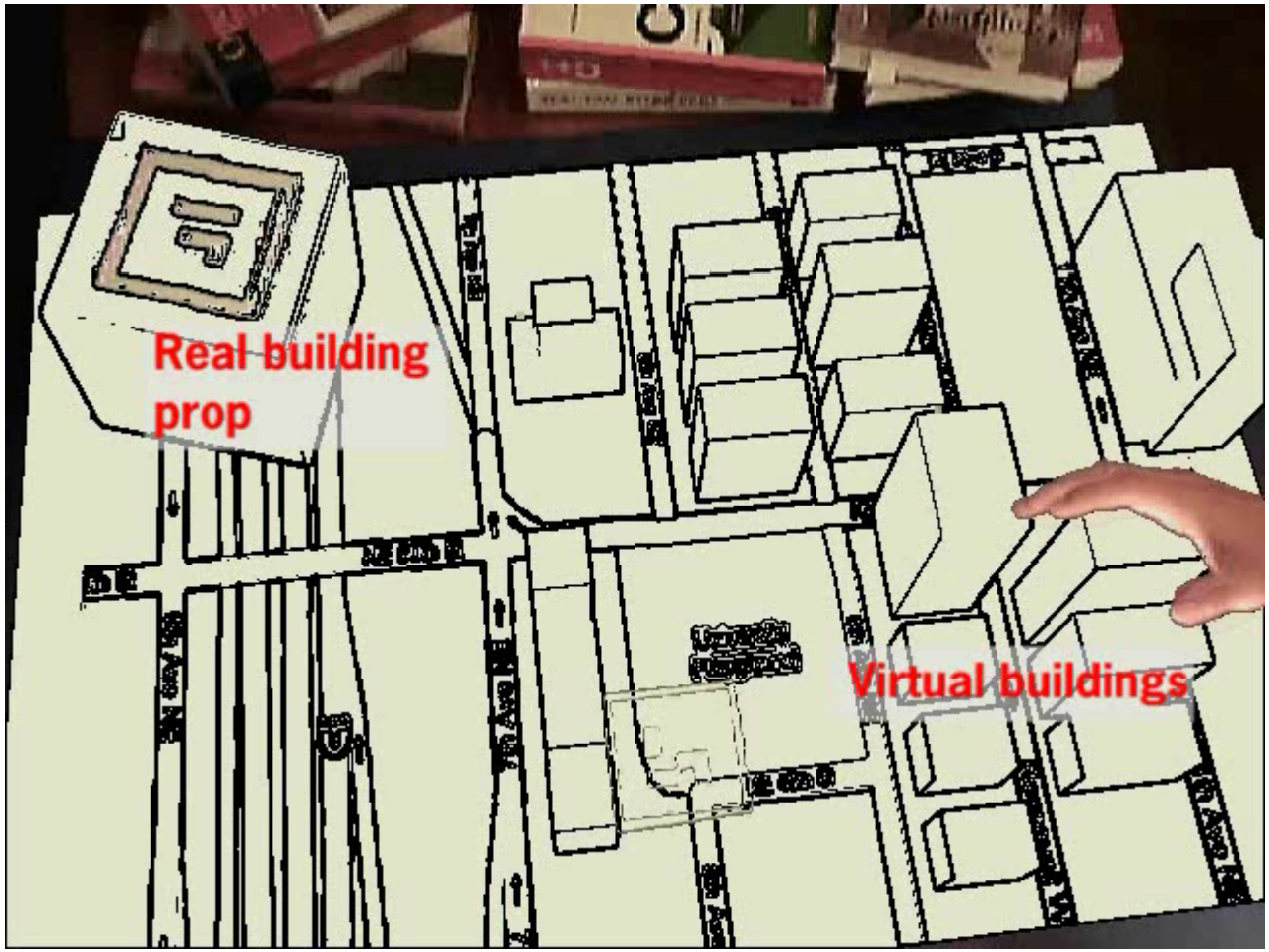
Tangible Interaction Setup



Graphical Augmentations

Stylization Domain - Video

Selective Stylization - Video



Open problems

- Online capture
  - Scene geometry
  - Dynamic Lighting
  - Dynamic objects
  - Mobile ?
- Optical see-through ?
  - Occlusion HMD
  - Maybe a blending display?
- I didn’t mention
  - Projection displays
  - Spatial augmented reality