Video in Video

Project of CS4243

Group Member:

Zhao Shanheng

Ding Feng

Wang Zhiyu

Outline

- Introduction
- Tracking the Photo Frame and Reflection
- Anti-Aliasing
- Transformation of Image
- Reflected Image
- Lighting Effect
- Demonstration
- Summary

Project Requirements

- Seamlessly embed one video into the photo frame of another video
- Reproduce the ambient lighting effects for the video within the photo frame
- Automatic or semi-automatic detection of photo frame in sequence

Objectives

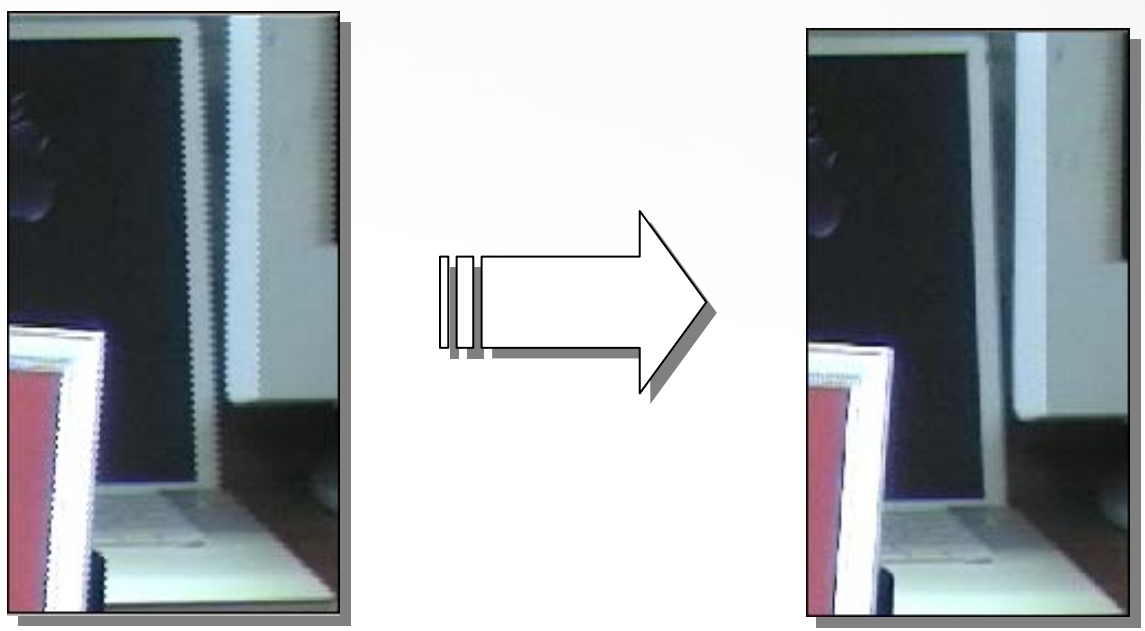
- Tracking of the photo frame and the reflection
- Video in video with multi-layer
- Video in video with reflected image (multi-layer)
- Reflected image with textile of desktop
- Anti-Aliasing
- Video with lighting effect

Shoot the Video

- “Red Screen”
- Cast a shadow on the frame

Pre-processing

- Discretize the video into images
- De-interlacing the images to reduce the *combing* effect



Before de-interlacing

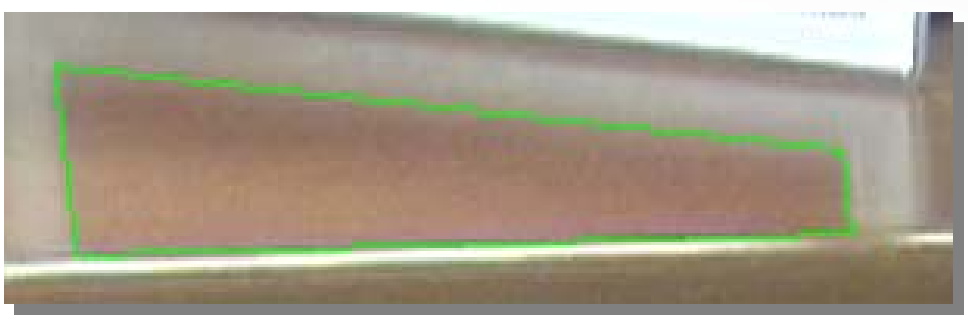
After de-interlacing

Track the Frame

- Manually select 4 points at the corner of the frame
- Use Lucas & Kanade Method to track those points

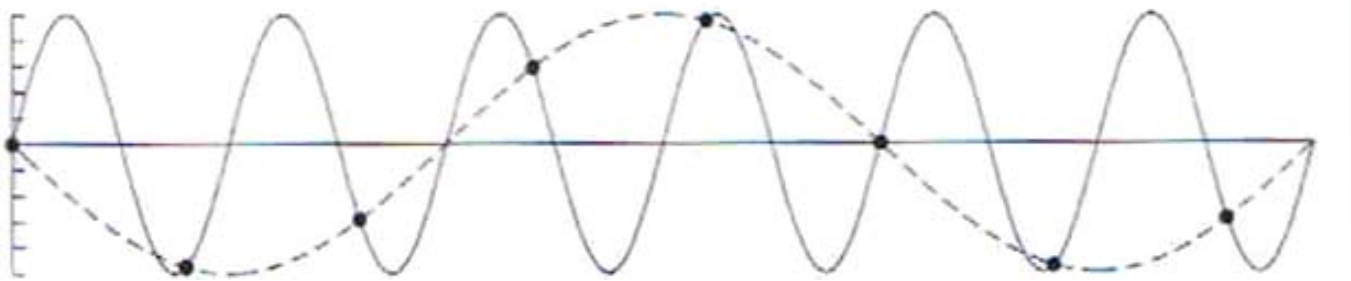
Track the Reflection

- Use the same method as tracking the frame
- Tracking points may drift
- Correct them manually



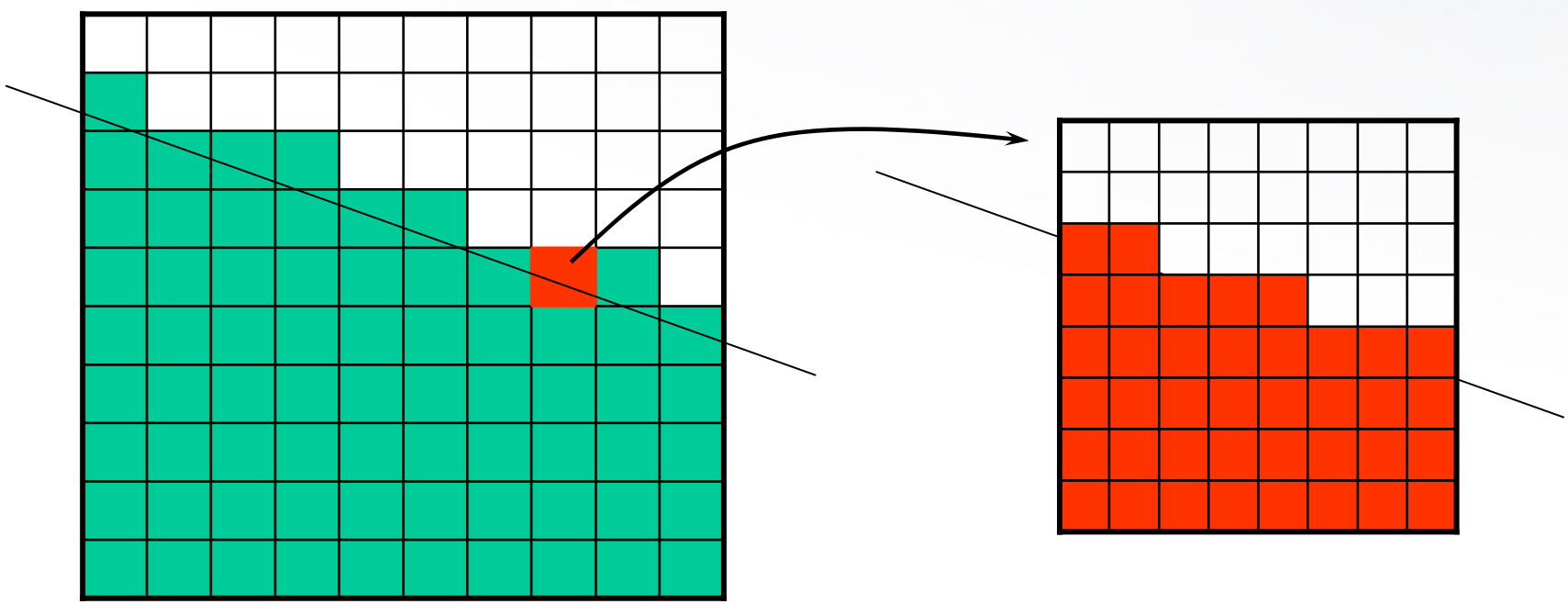
Anti-Aliasing

- Aliasing
  - *Jigsaw* effect due to under-sampling



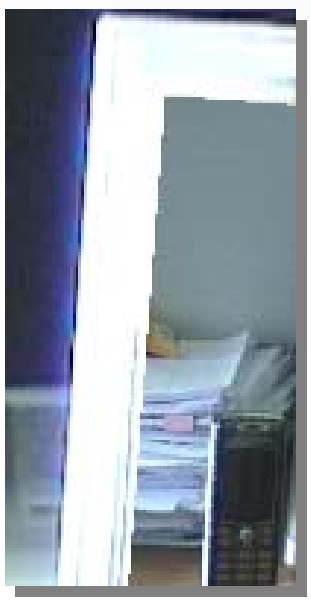
Anti-Aliasing

- Algorithm
  - Divide pixel into sub-pixels
  - Count sub-pixels

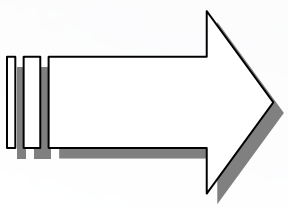


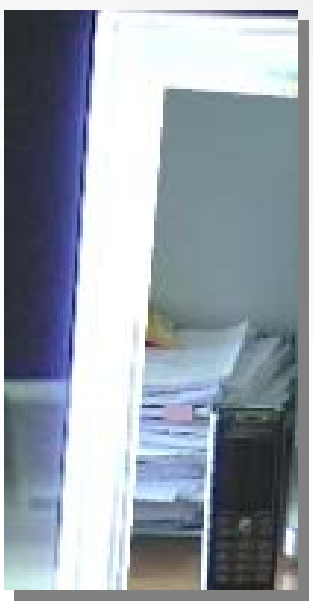
Anti-Aliasing

- Result



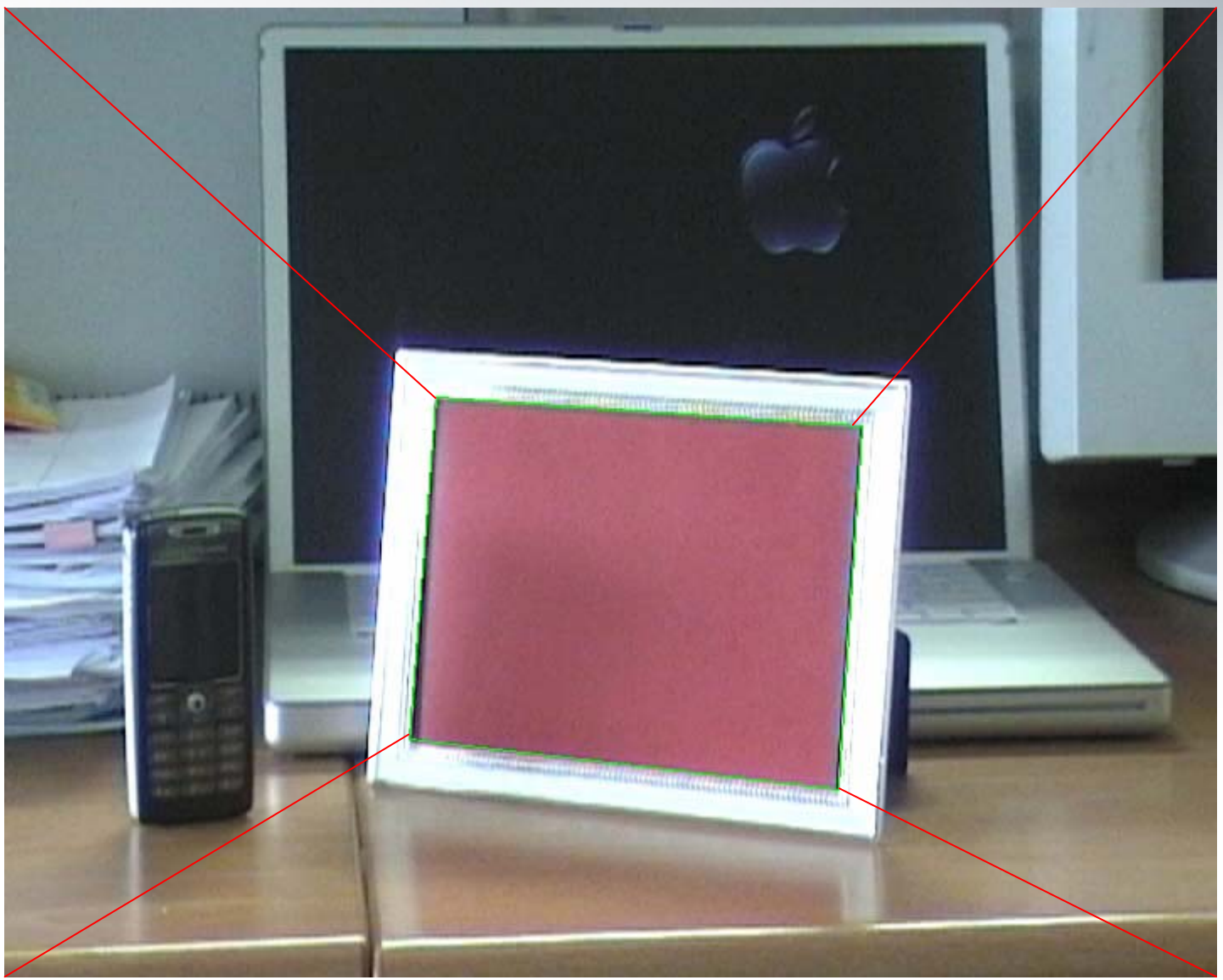
Before anti-aliasing





After anti-aliasing

Point Correspondences



Transformation Matrix

$$\begin{bmatrix} u_j \\ v_j \\ 1 \end{bmatrix} = \begin{bmatrix} a & b & c \\ d & e & f \\ g & h & i \end{bmatrix} \begin{bmatrix} x_j \\ y_j \\ 1 \end{bmatrix}$$

$$\begin{bmatrix} x_1 & y_1 & 1 & 0 & 0 & 0 & -u_1x_1 & -u_1y_1 \\ x_2 & y_2 & 1 & 0 & 0 & 0 & -u_2x_2 & -u_2y_2 \\ x_3 & y_3 & 1 & 0 & 0 & 0 & -u_3x_3 & -u_3y_3 \\ x_4 & y_4 & 1 & 0 & 0 & 0 & -u_4x_4 & -u_4y_4 \\ 0 & 0 & 0 & x_1 & y_1 & 1 & -v_1x_1 & -v_1y_1 \\ 0 & 0 & 0 & x_2 & y_2 & 1 & -v_2x_2 & -v_2y_2 \\ 0 & 0 & 0 & x_3 & y_3 & 1 & -v_3x_3 & -v_3y_3 \\ 0 & 0 & 0 & x_4 & y_4 & 1 & -v_4x_4 & -v_4y_4 \end{bmatrix} \begin{bmatrix} a \\ b \\ c \\ d \\ e \\ f \\ g \\ h \end{bmatrix} = \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \\ v_1 \\ v_2 \\ v_3 \\ v_4 \end{bmatrix}$$

Transformation Matrix – cont.

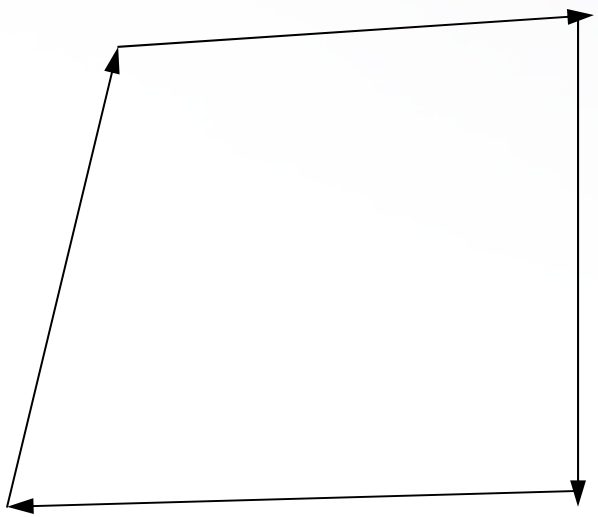
$$Ap = b$$

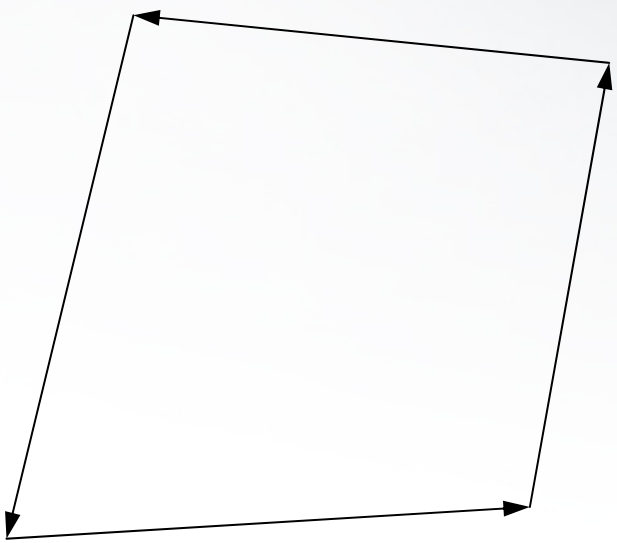
$$p \rightarrow H$$

$$H \rightarrow H_{Inv}$$

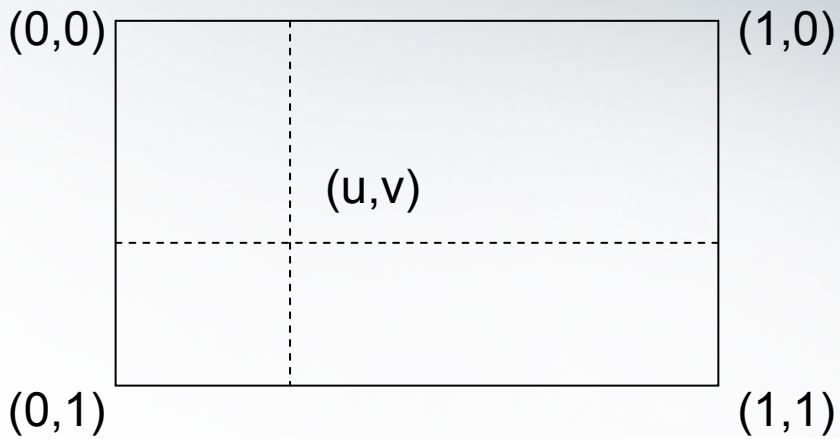
Point in Polygon

- $(y - y_0)(x_1 - x_0) - (x - x_0)(y_1 - y_0)$



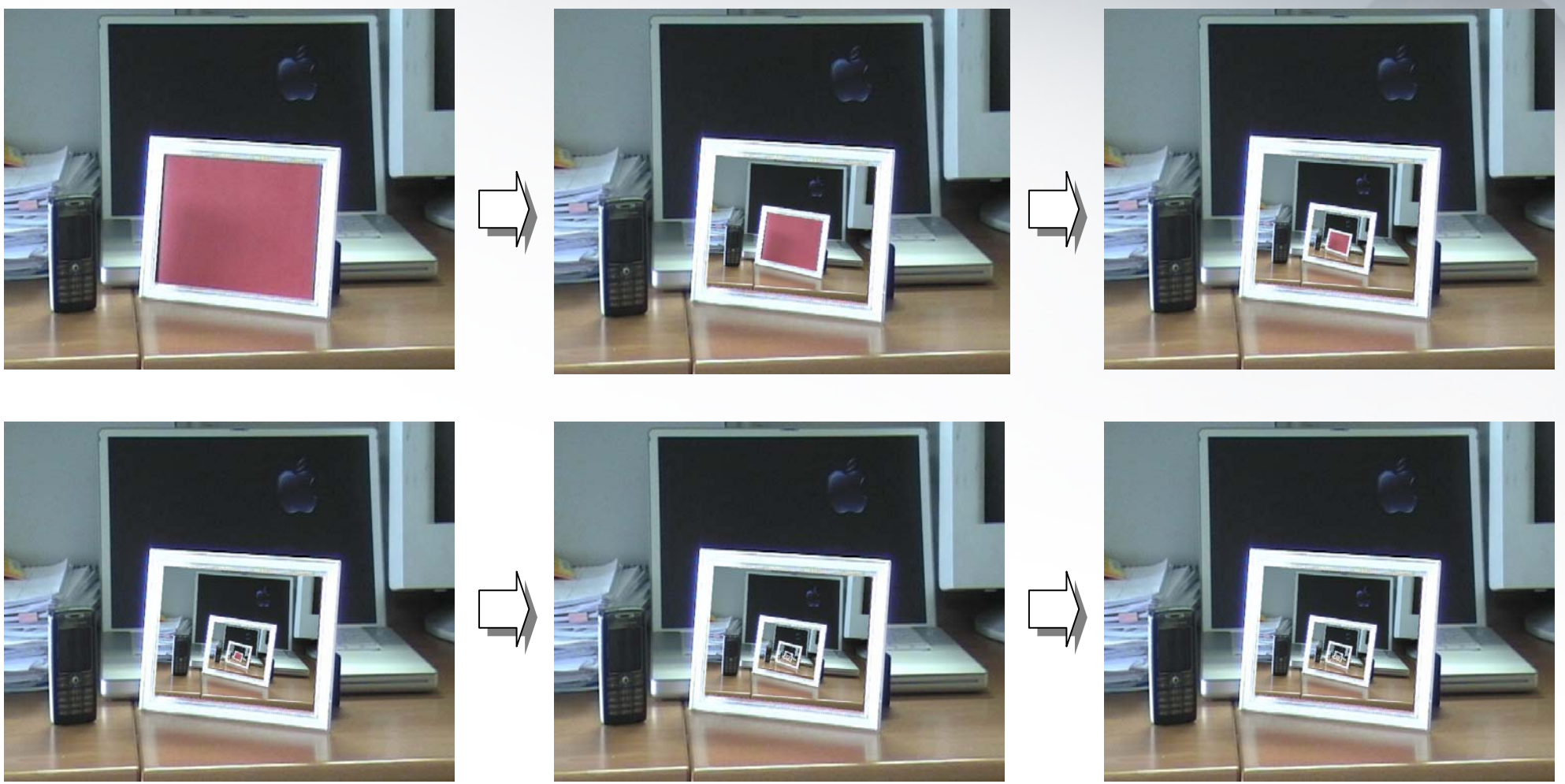


Bilinear Interpolation

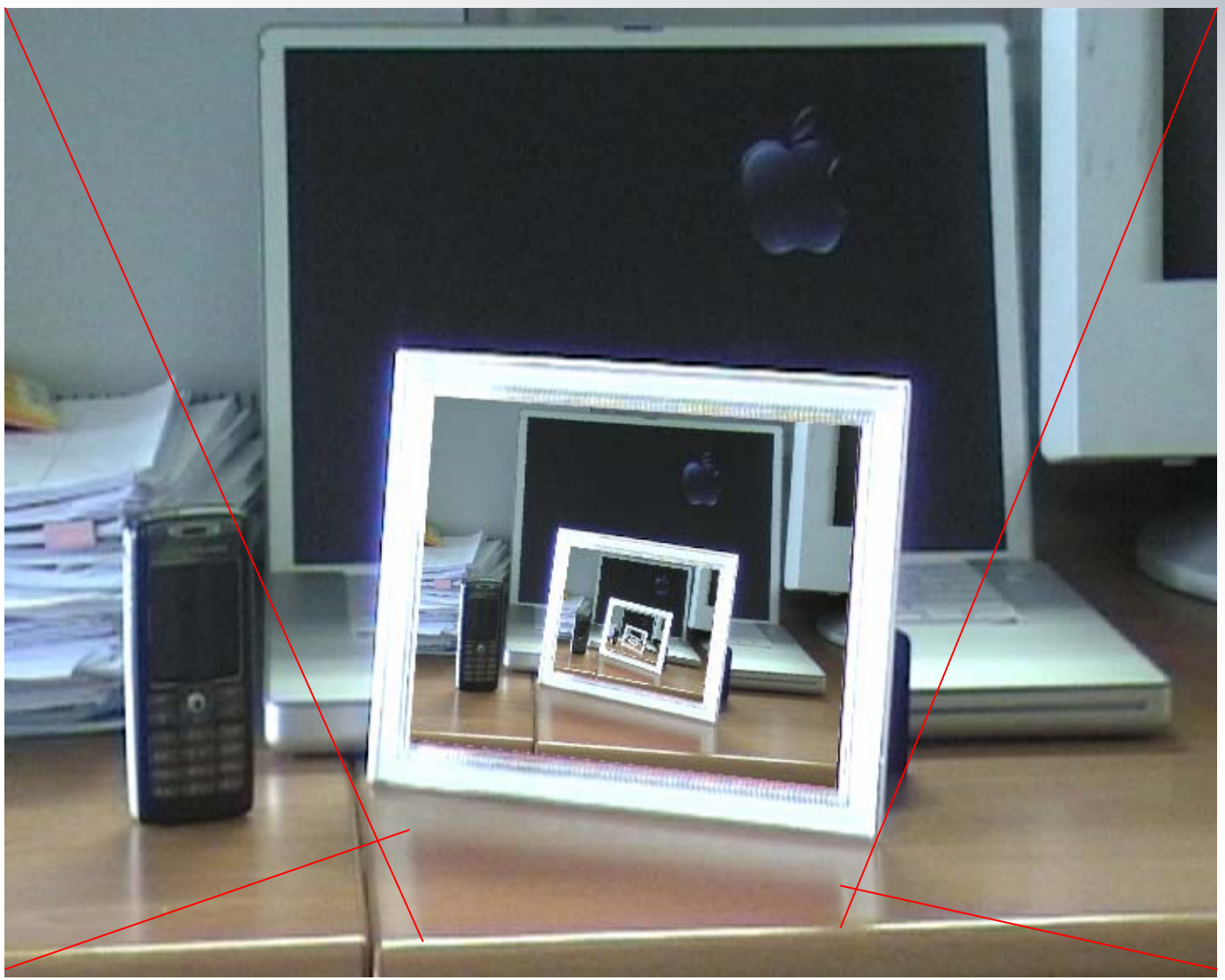


- $f_{uv} = (1-u)(1-v)f_{00} + u(1-v)f_{10} + (1-u)vf_{01} + uvf_{11}$
- $f_{u0} = (1-u)f_{00} + uf_{10} = f_{00} + u(f_{10} - f_{00})$
- $f_{u1} = (1-u)f_{01} + uf_{11} = f_{01} + u(f_{11} - f_{01})$
- $f_{uv} = (1-v)f_{u0} + vf_{u1} = f_{u0} + v(f_{u1} - f_{u0})$

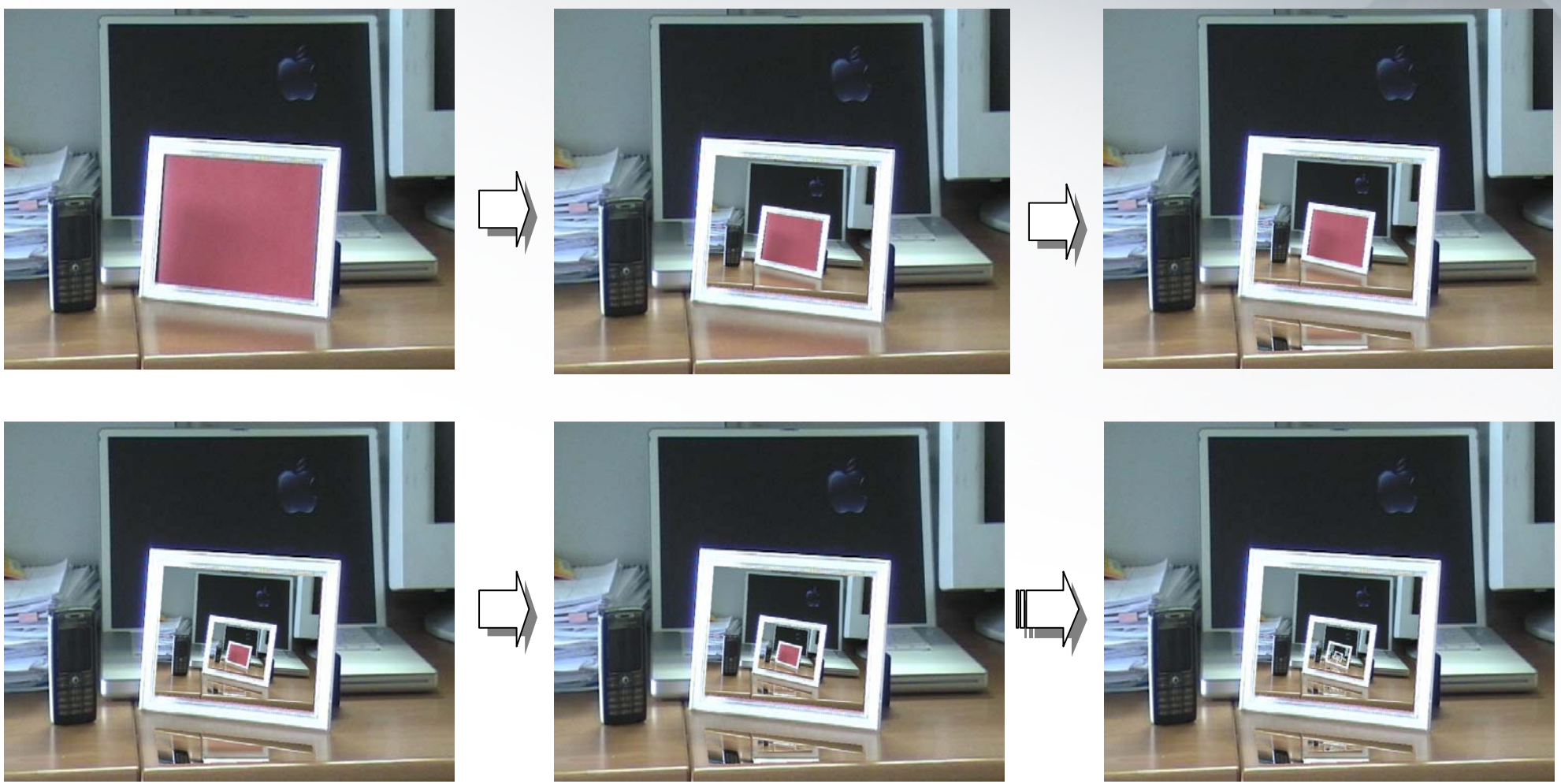
Image After Transformation



Realistic?



Iterative Transformation



Alpha Blending

Original Image

Edited Image

RGB vs. YUV

- YUV is the color space used in the PAL (Phase Alternation Line) system of television broadcasting which is the standard in most of Europe and some other places.
- Y stands for the luminance (brightness) component.
- U and V are the chrominance (color) components.

RGB ⇔ YUV

$$Y = 0.299 * R + 0.587 * G + 0.114 * B$$

$$U = (B - Y) * 0.565$$

$$V = (R - Y) * 0.713$$

$$R = Y + 1.403 * V$$

$$G = Y - 0.344 * U - 0.714 * V$$

$$B = Y + 1.770 * U$$

How to keep the shadow?

In the frame of the original image,

1. Convert RGB to YUV, and get Yo;
2. Calculate the meanYo;

In the frame of the edited image,

3. Convert RGB to YUV, and get Ye;
4. Update the Y by

   $Ye = Ye + \alpha * (Yo - meanYo)$,

   in practice, $\alpha = 1.5$;
5. Convert YUV back to RGB.

Shadowed Image

Edited Image

Noise Alert!

Original Image

How to remove the noise?

- Box-filter Method:

$$f(x,y) \otimes k(x,y) = \frac{1}{w^2} \sum_{i=-w/2}^{w/2} \sum_{j=-w/2}^{w/2} f(x+i, y+j)$$

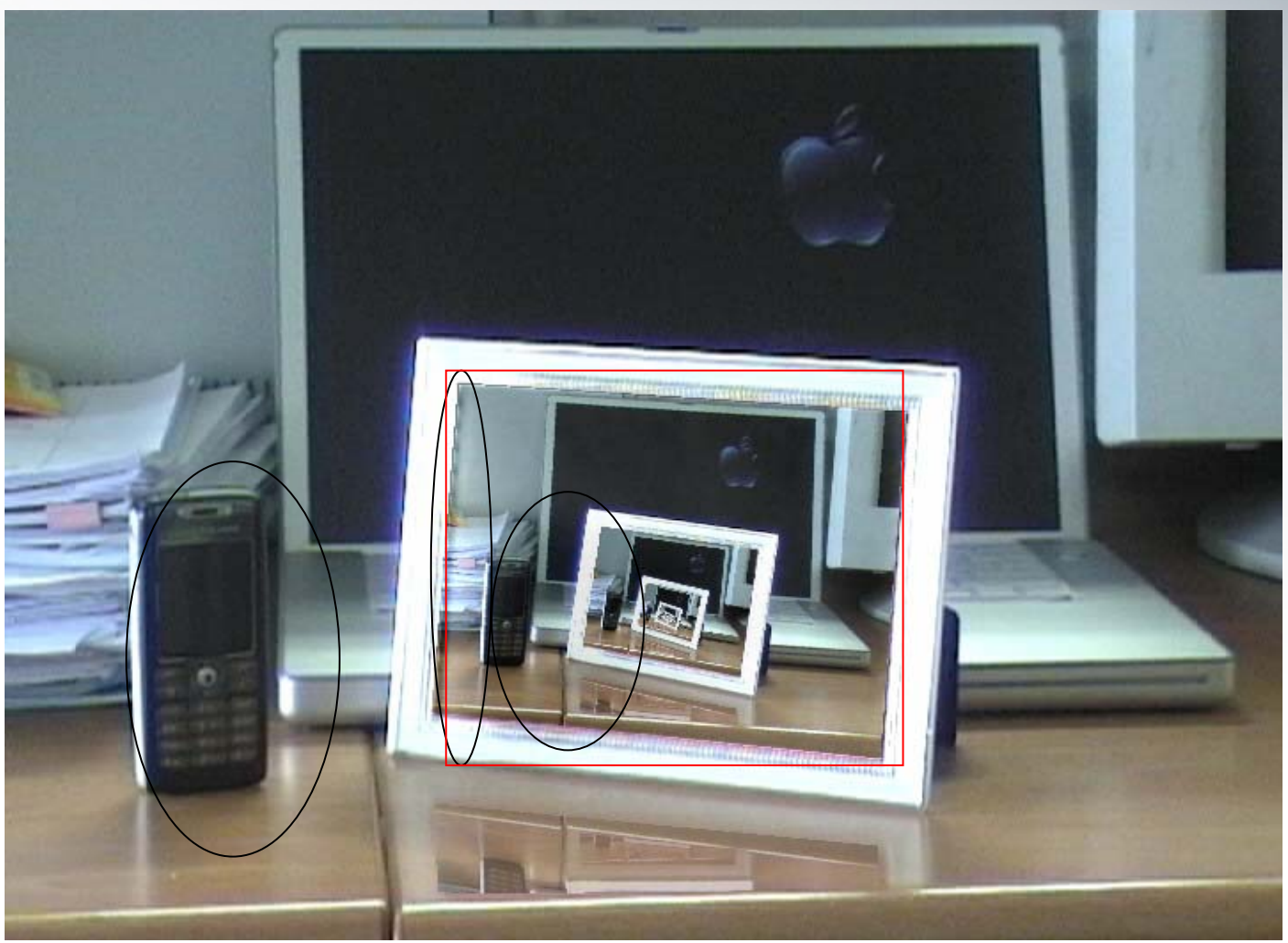
- Gaussian-filter Method:

$$f(x,y) \otimes g(x,y) = \sum_{i=-w/2}^{w/2} \sum_{j=-w/2}^{w/2} f(x+i, y+j) g(i,j)$$

1-D (normalized) Gaussian:

$$g'(i,j) = \frac{1}{\sum_{p=1}^{w^2} g(i,j)} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right)$$

Smoothed & Shadowed Image



With Gaussian-filter, mask size = 5

Putting All Together

Summary

- Tracking of photo frame and reflection, anti-aliasing
- Video in video with multi layer and reflection
- Video with lighting effect

THANK YOU