Illusion of Motion

Amir H. Moin
amir.moin@dfki.de

Seminar: **C**omputer **V**ision **f**or **C**omputer **G**raphics (CVfCG)

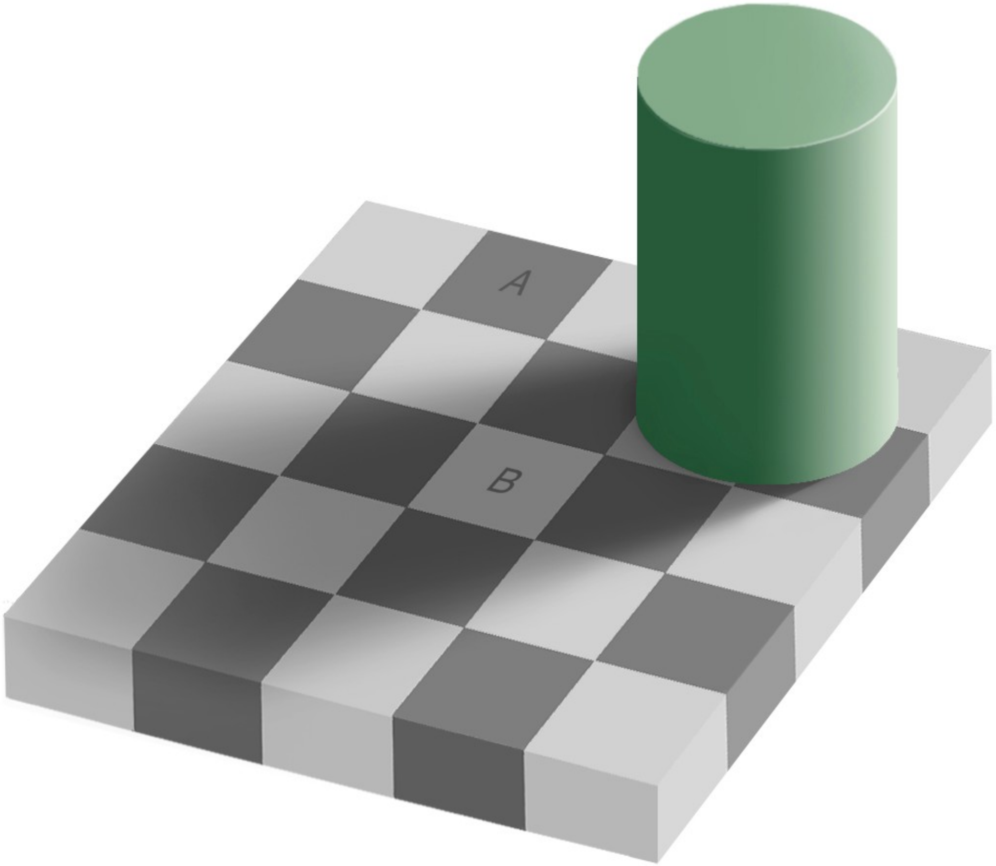
Max **P**lanck **I**nstitute for **I**nformatics (MPII)
Saarbrücken
May 7, 2013

Outline

- Introduction to illusory motion
- Exploring Photobios (Shlizerman et al. 2011)
- Video Magnification (Wu et al. 2012)
- Summary & Conclusion
- Questions & Discussion

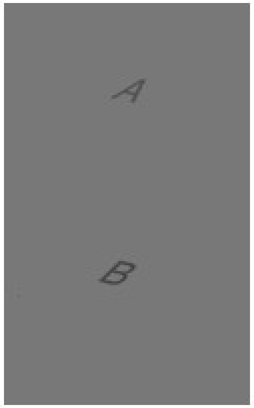
Optical Illusion, Example 1

- Checker Shadow Illusion
  - Prof. Edward H. Adelson, 1995



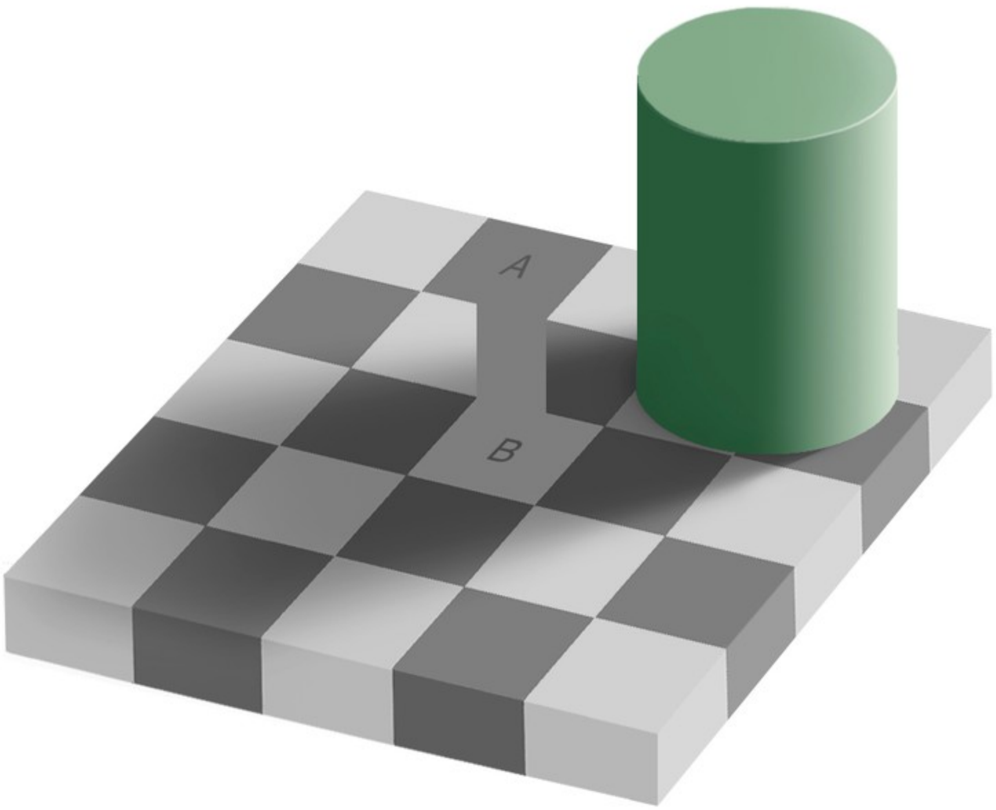
Optical Illusion, Example 1

- Checker Shadow Illusion
  - Prof. Edward H. Adelson, 1995



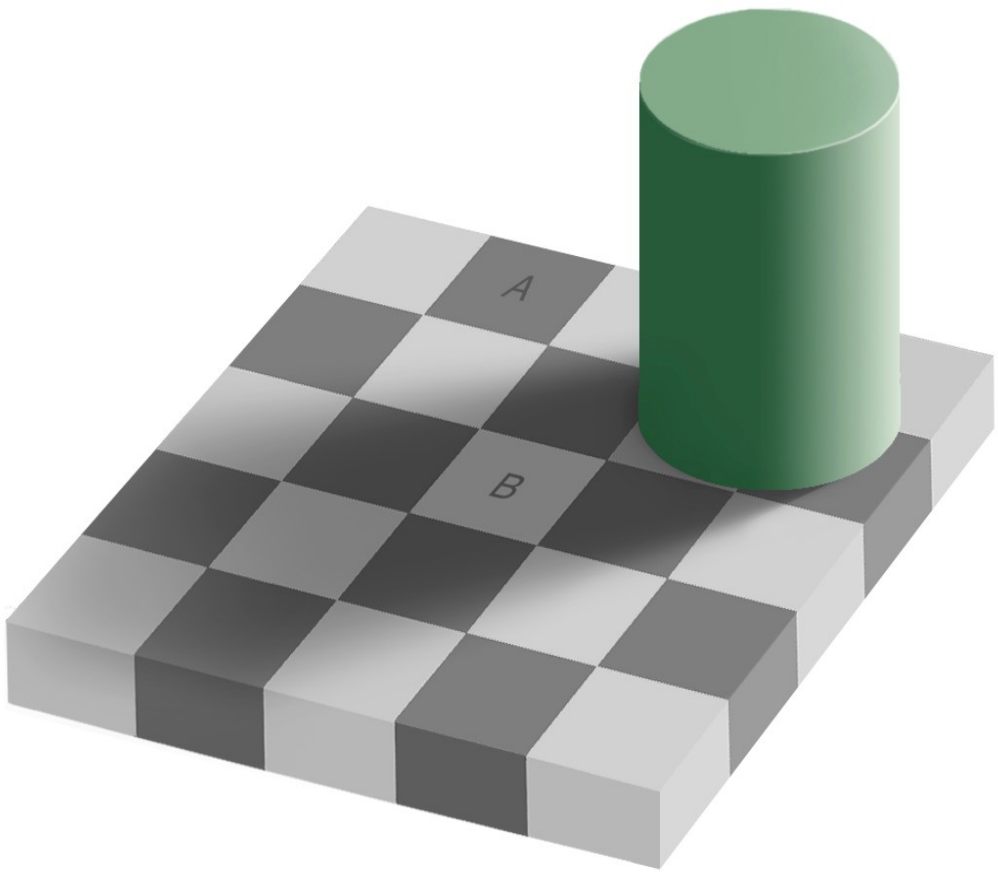
Optical Illusion, Example 1

- Checker Shadow Illusion
  - Prof. Edward H. Adelson, 1995



Optical Illusion, Example 1

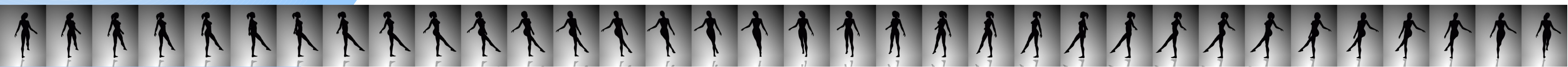
- Checker Shadow Illusion
  - Prof. Edward H. Adelson, 1995



Optical Illusion, Example 2

- Spinning Dancer (silhouette illusion)
  - by web designer Nobuyuki Kayahara, 2003

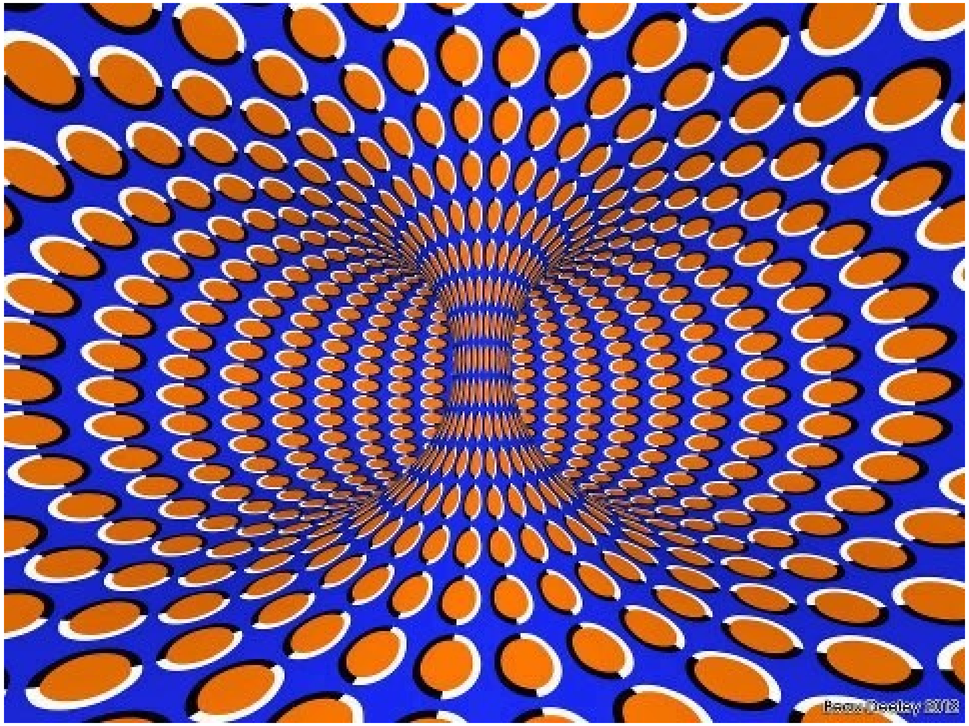
Optical Illusion, Example 2



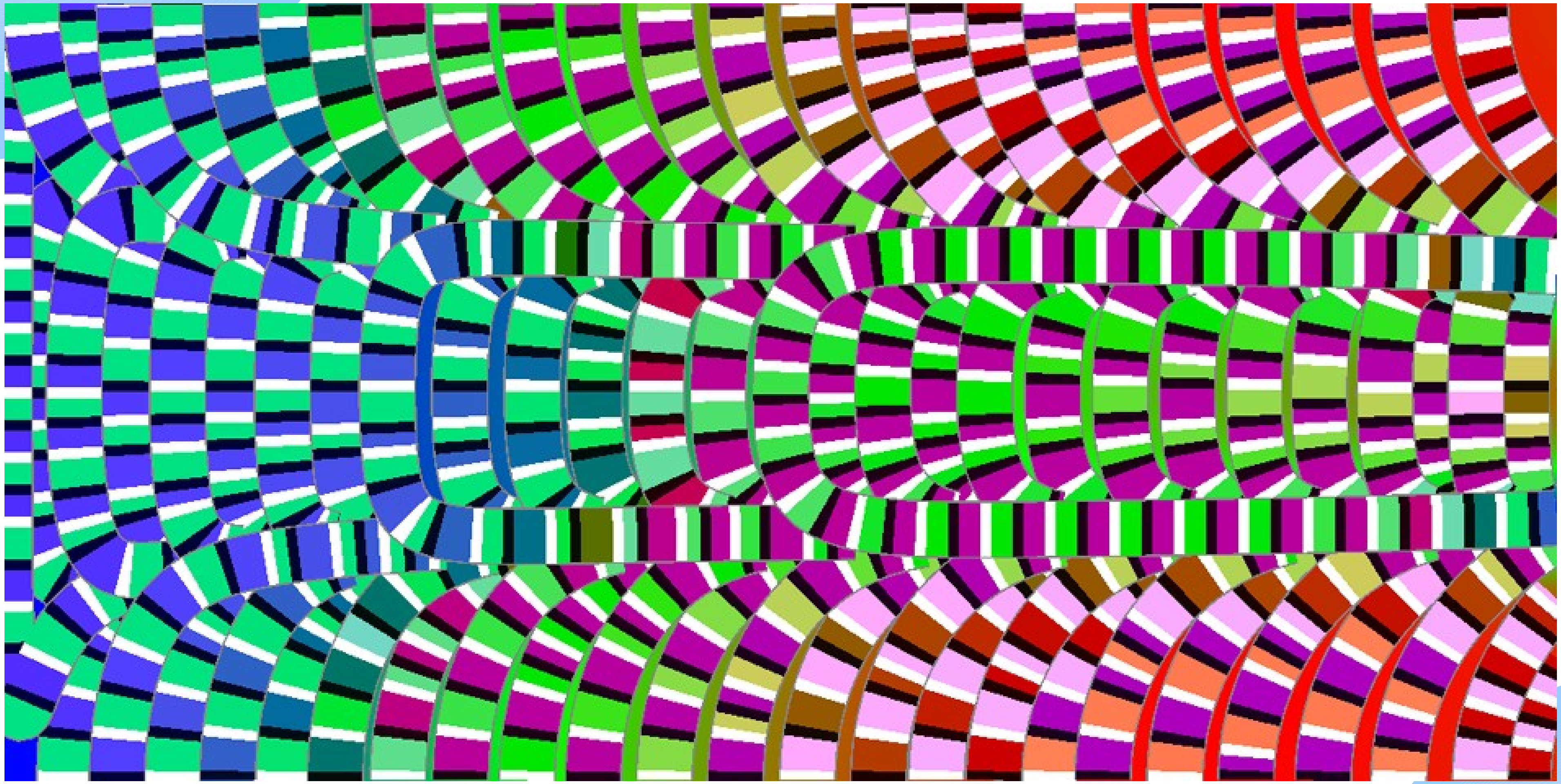
clockwise

counter-clockwise

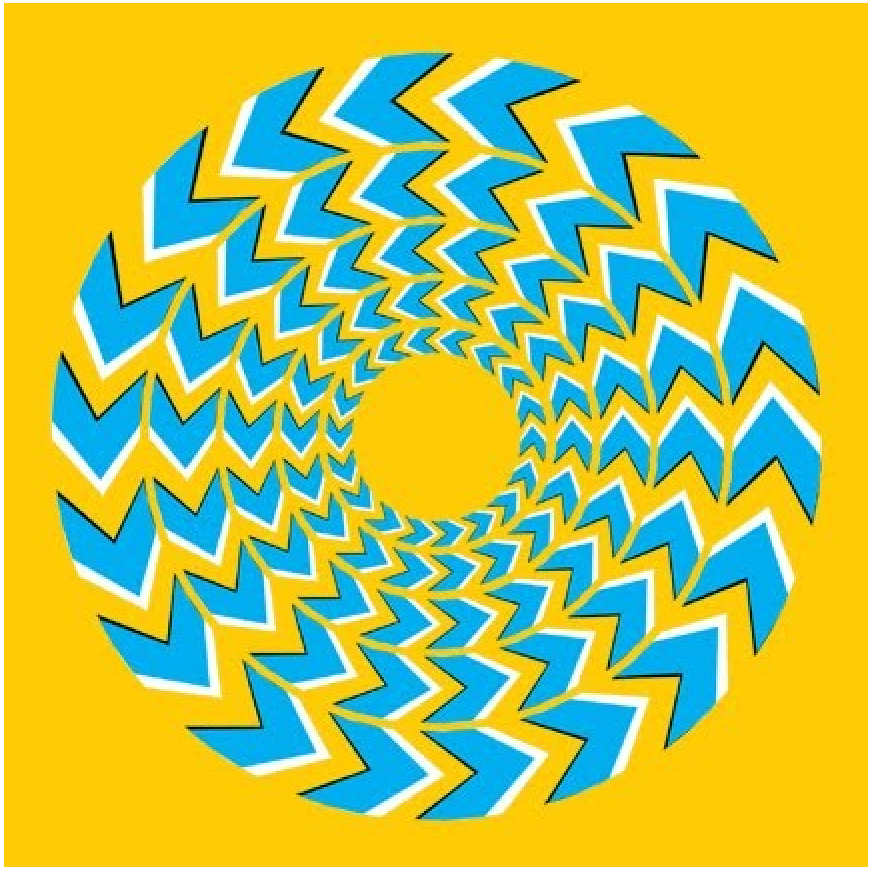
Optical Illusion, Example 3: Illusory Motion



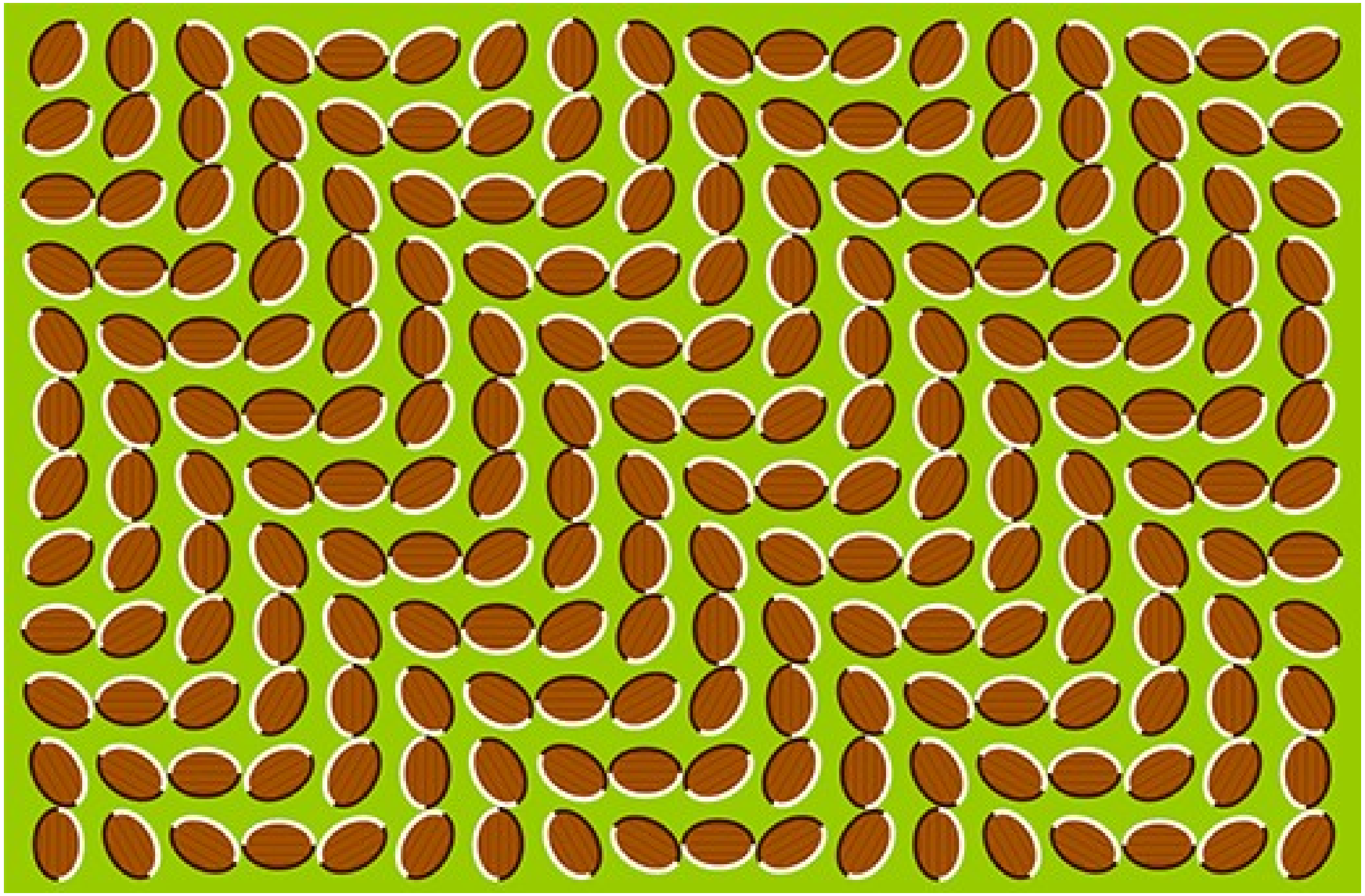
Optical Illusion, Example 4: Illusory Motion



Optical Illusion, Example 5: Illusory Motion



Optical Illusion, Example 6: Illusory Motion



Illusory Motion

- A.k.a. motion illusion (illusion of motion)
- Kind of optical illusion
- **Static** image *appears* to be moving
- Cognitive effects:
  - Color contrasts
  - Shape positions

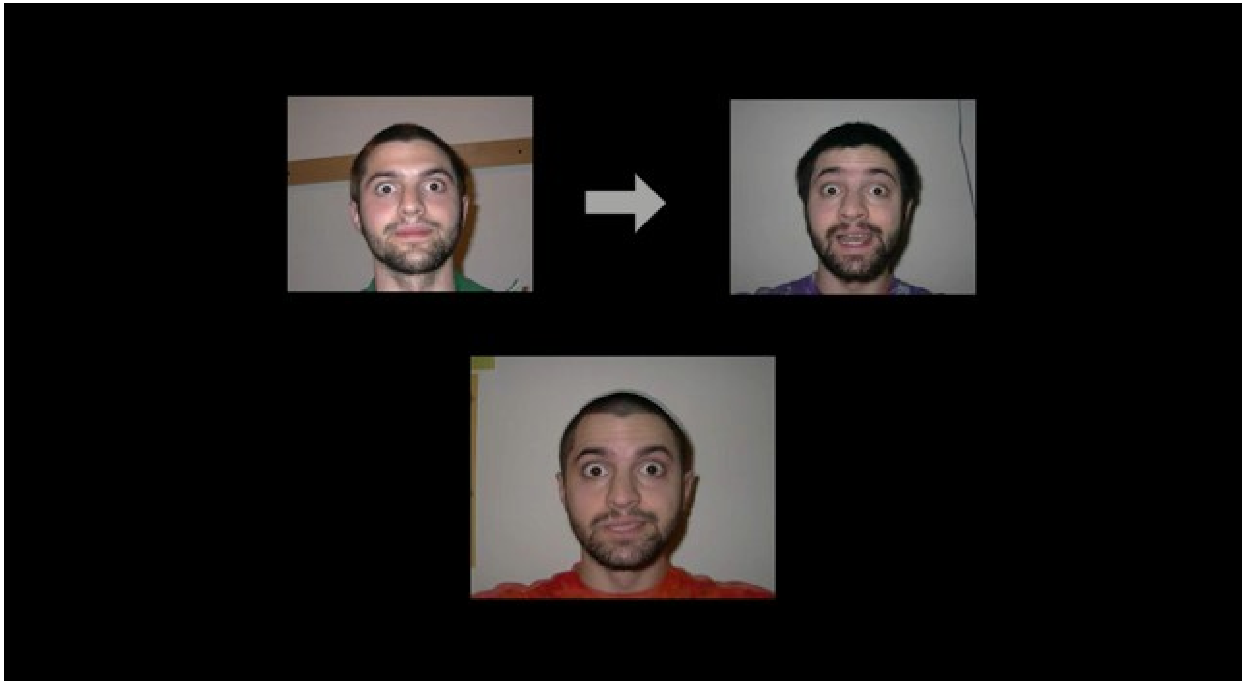
Outline

- Introduction to illusory motion
- Exploring Photobios (Shlizerman et al. 2011)
- Video Magnification (Wu et al. 2012)
- Summary & Conclusion
- Questions & Discussion

Photobios (Shlizerman et al. 2011)

- Photobio: large image collection of the same person
- Samples the appearance space of the person over time

Sample Face Animation



Generating Face Animations

- Optimizing the order
- Cross dissolving

Optimizing the Order

Choose among a very large image collection (several thousands)

Cross Dissolve (a.k.a. Cross Fade or Linear Intensity Blend)

- Gradual transition from one image to another
- Fade-out vs. Fade-in

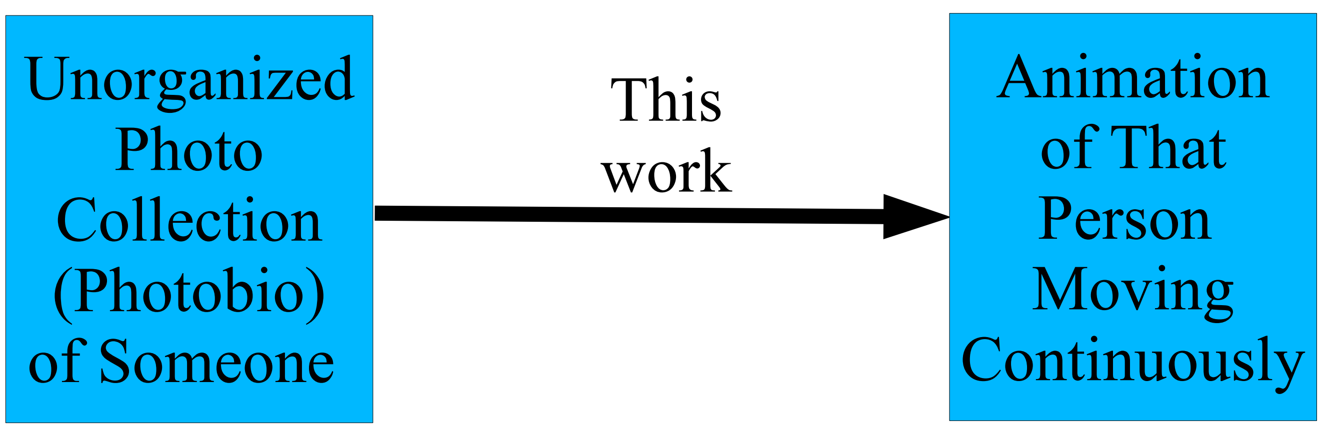
Source: http://en.wikipedia.org/wiki/Dissolve_%28filmmaking%29

The Key Idea of This Work

- *Cross dissolving* well-aligned images produces a very strong motion sensation
- Not only *illusion* of motion, but *true* motion!

The Aim

To create interactive animated viewing experiences from a person's photobio



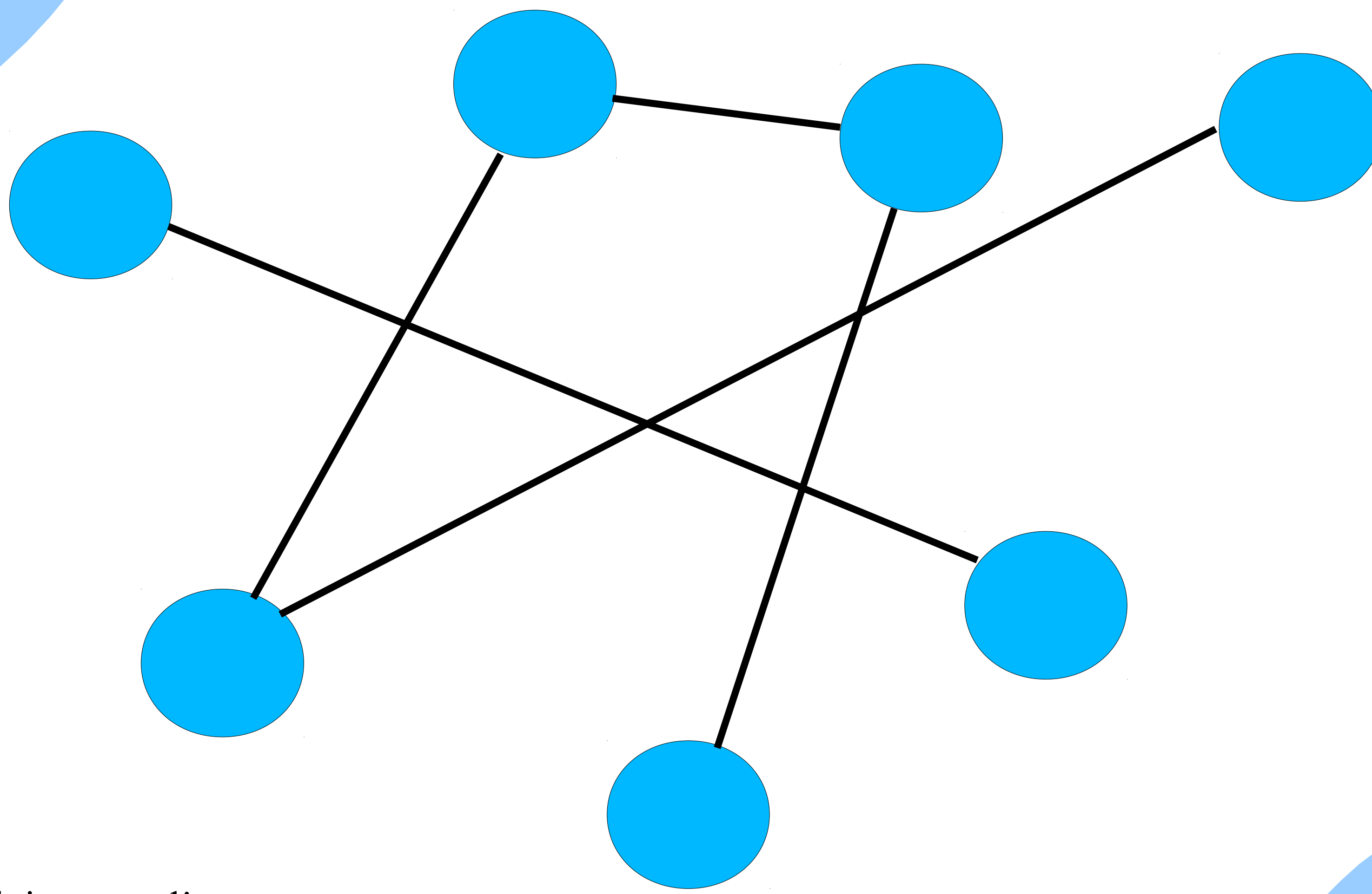
The Specific Problem

- View Interpolation:
  - Rendering a seamless transition between two images

Challenges

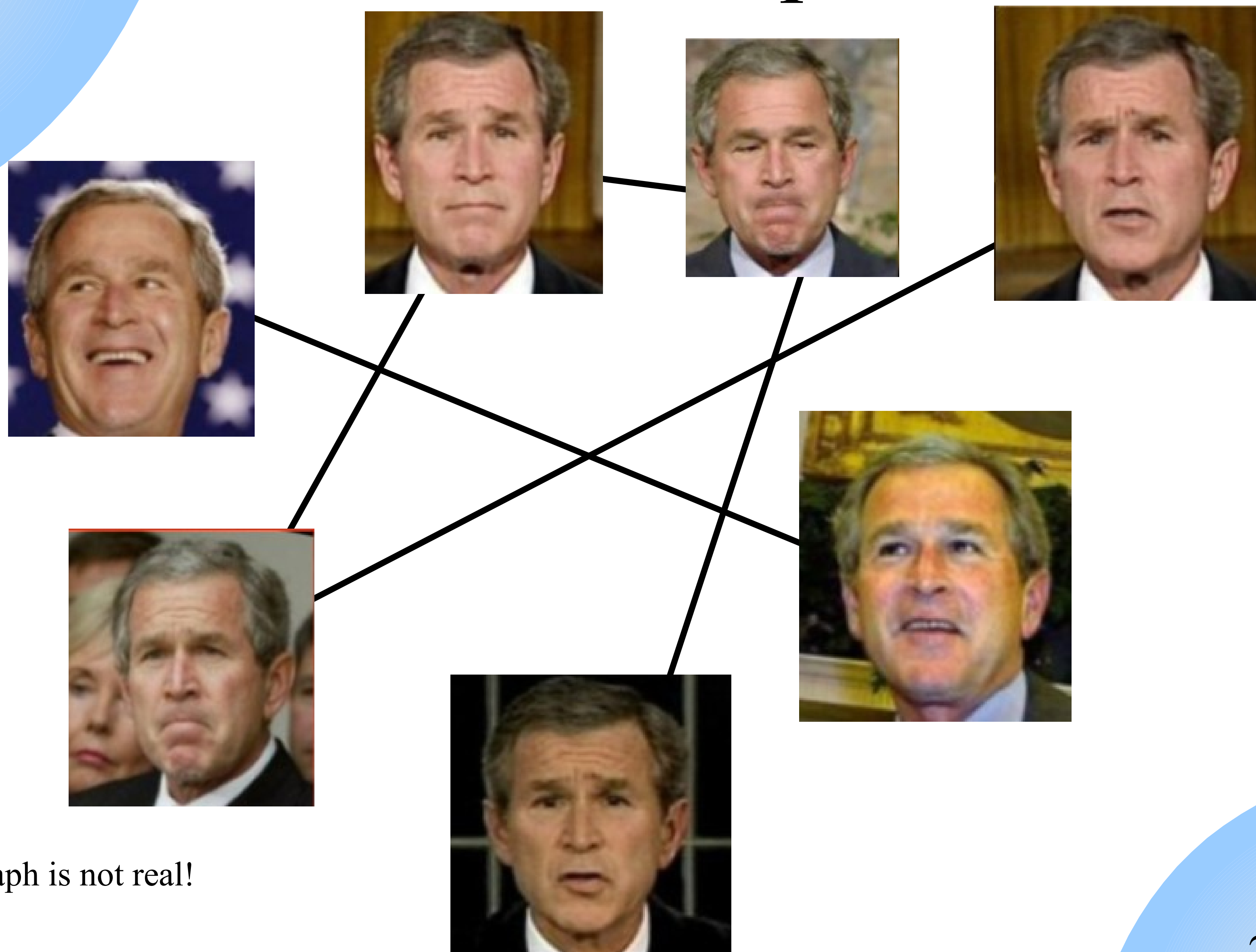
- Face appearance space: extremely *high-dimensional*
- Limited access: only a *sparse* sample space
- The exact mapping of each image to pose, expression, etc. is not known!

The Face Graph



Note: This graph is not real!

The Face Graph



Note: This graph is not real!

The Face Graph

- Nodes: Face Images
- Edges: Relative distances (dissimilarities)
- Problem: smooth transition between well-aligned images
- Equivalent to traversing the shortest path on the face graph

How to Find the Distances?

Comparing the face images!

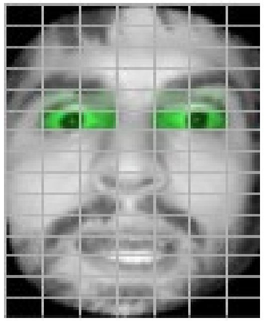
The Pre-Processing Pipeline

- Face detection
  - Locating eyes, nose, mouth, hair
  - Ignore photos with low detection confidence
- Pose detection
  - Aligning to a 3D template model
  - Warping to frontal views

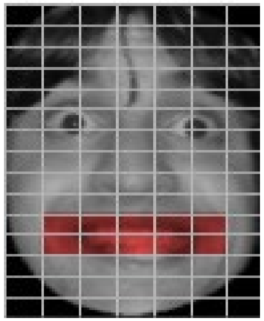
Comparing Images to Find Distances

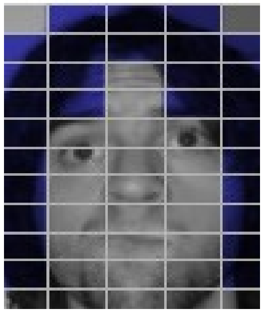
- Local Binary Patterns (LBP) Histograms
- Already proven to be useful for:
  - Image classification
  - Face recognition
  - Expression identification
  - Etc.

Local Binary Pattern (LBP) Histograms



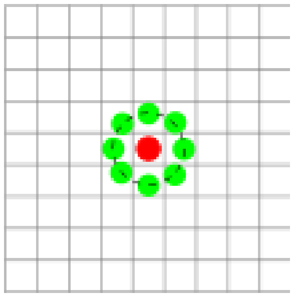
- Divide an image to a grid of cells
- Convert each pixel in a cell to a **binary code**





How to Calculate the Per-Pixel Binary Code?

- Compare each pixel to its 8 neighbor pixels
- For each neighbor pixel:
  - if brighter than the center pixel → 1
  - if darker than the center pixel → 0



  - 8-digits binary code for each pixel in a cell

Local Binary Pattern (LBP) Histograms

The histogram of these codes for each cell is the descriptor of that cell.

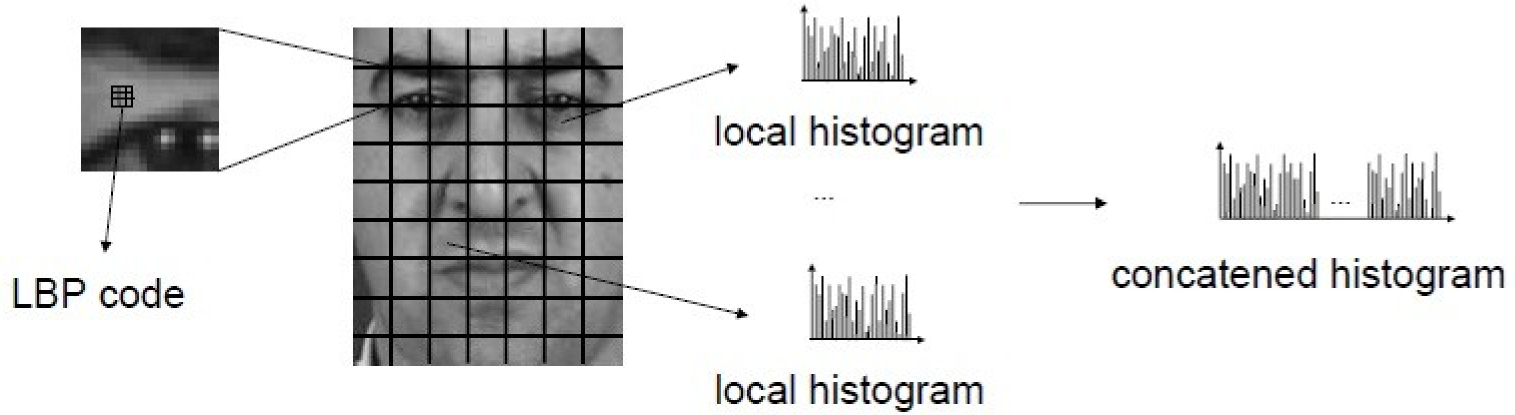


Image Source: http://www.intechopen.com/source/html/17176/media/image24.jpg

Face Distances

- Combination of difference in:
  - Appearance
  - Pose
  - Time (if timestamps are available)

The Appearance Difference of Face Images i and j

- Distance between the corresponding descriptors in face images i and j
- Normalized using a robust logistic function

$$D_{appearance}(i,j) = 1 - (1 - \lambda^m d_{ij}^m)(1 - \lambda^e d_{ij}^e)(1 - \lambda^h d_{ij}^h)$$

The Appearance Difference of Face Images i and j

$$D_{appearance}(i,j) = 1 - (1 - \lambda^m d_{ij}^m)(1 - \lambda^e d_{ij}^e)(1 - \lambda^h d_{ij}^h)$$

LBP histogram distance for mouth

The Appearance Difference of Face Images i and j

$$D_{appearance}(i,j) = 1 - (1 - \lambda^m d_{ij}^{\,m})(1 - \lambda^e d_{ij}^{\,e})(1 - \lambda^h d_{ij}^{\,h})$$

LBP histogram distance for mouth

for hair

for eyes

The Appearance Difference of Face Images i and j

$$D_{appearance}(i,j) = 1 - (1 - \lambda^{m} d_{ij}^{m})(1 - \lambda^{e} d_{ij}^{e})(1 - \lambda^{h} d_{ij}^{h})$$

Weights for the regions

The Appearance Difference of Face Images i and j

$$D_{appearance}(i,j) = 1 - (1 - \lambda^m d_{ij}^m)(1 - \lambda^e d_{ij}^e)(1 - \lambda^h d_{ij}^h)$$

Weights for the regions

$$\lambda^m = 0.8,\ \lambda^e = 0.1,\ \lambda^h = 0.1$$

Face Distances

- Combination of difference in:
  - Appearance: $D_{appearance}(i,j)$
  - Pose: $D_{yaw}(i,j)$, $D_{pitch}(i,j)$
  - Time (if timestamps are available): $D_{time}(i,j)$

- For pose and time → absolute values

Recall: Euler Angles

- Yaw
- Pitch
- Roll

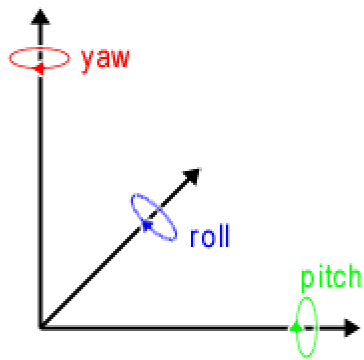


fig. 1: Euler angles

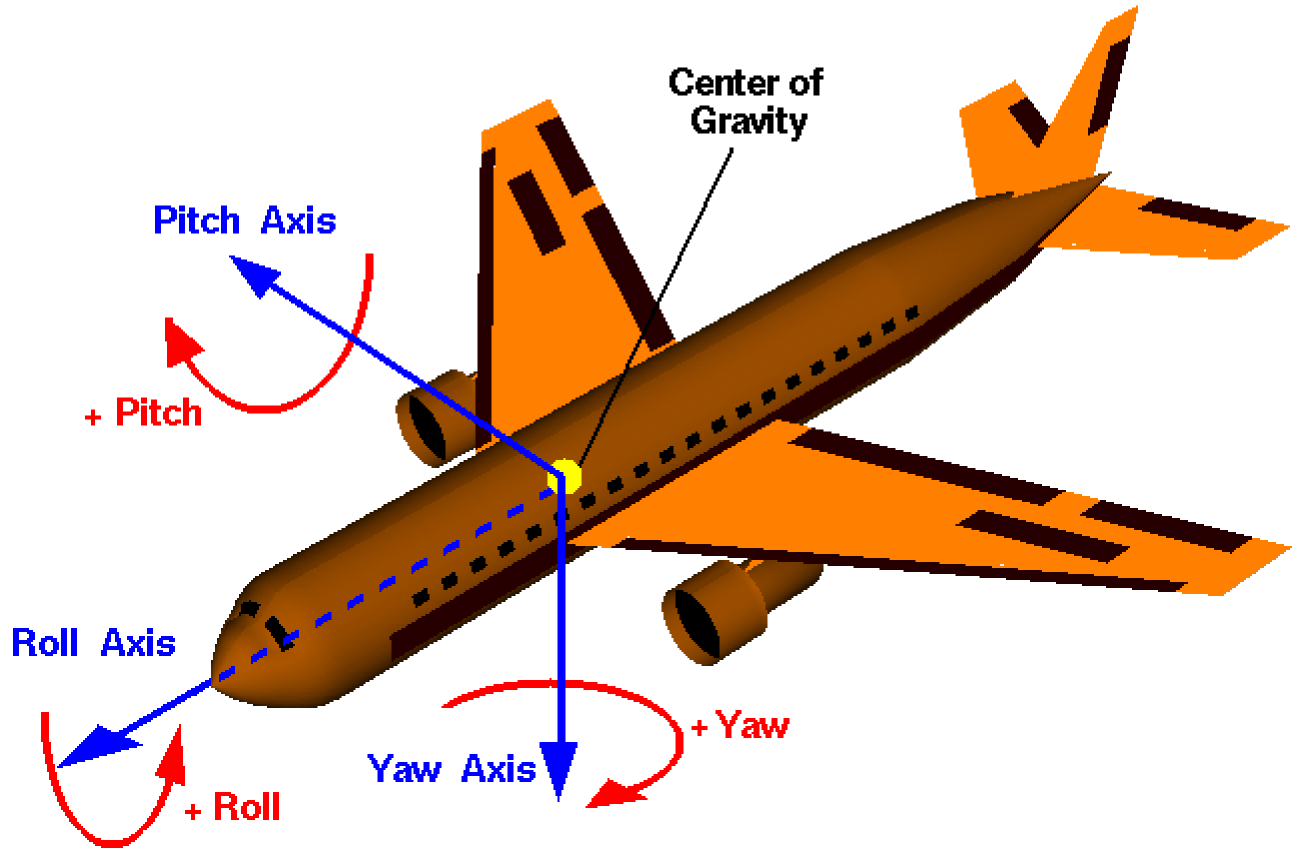


Image Sources: http://www.gameprogrammer.net/delphi3dArchive/viewing.htm
http://copterix.perso.rezel.net/wp-content/uploads/2011/04/rotations1.gif

The Face Graph

- Face images i and j → Nodes i and j
- Edge (i,j) has weight D(i,j) defined as:

$$D(i,j) = [1 - \Pi_{s \in \{appearance,yaw,pitch,time\}} (1 - D_s(i,j))]^{\alpha}$$

The number of in-between images → the α parameter

Using The Face Graph

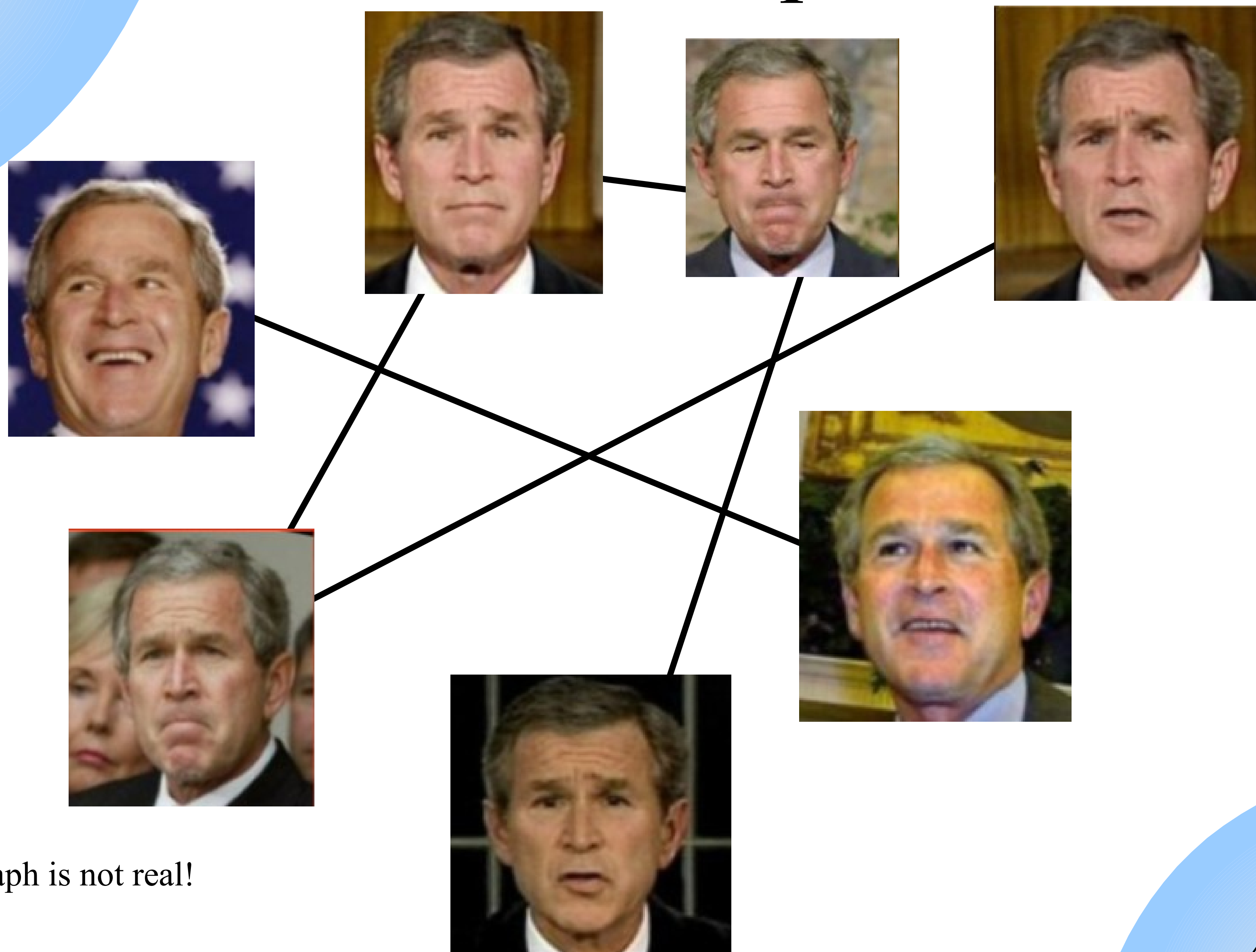
Smooth continuous image transitions

$$\equiv$$

Traverse the shortest path on the face graph

- Dijkstra's algorithm

The Face Graph



Note: This graph is not real!

What do we have? What do we need?

- We have got: a proper sequence of images
- We need: a way to render smooth & compelling transitions from one photo to the next

A Classic Solution: Morphing

- Change (morph) one image to another through a seamless transition
  - E.g. one person turning to another one

A morph from George W. Bush to Arnold Schwarzenegger
Source: http://en.wikipedia.org/wiki/Morphing

A Simpler Solution: Cross Dissolve (a.k.a. Cross Fade)

Fade-out one image and fade-in another one simultaneously.

$$I_{out}(t) = (1-t)\, I_{in1} + t\, I_{in2}$$

A Simpler Solution: Cross Dissolve (a.k.a. Cross Fade)

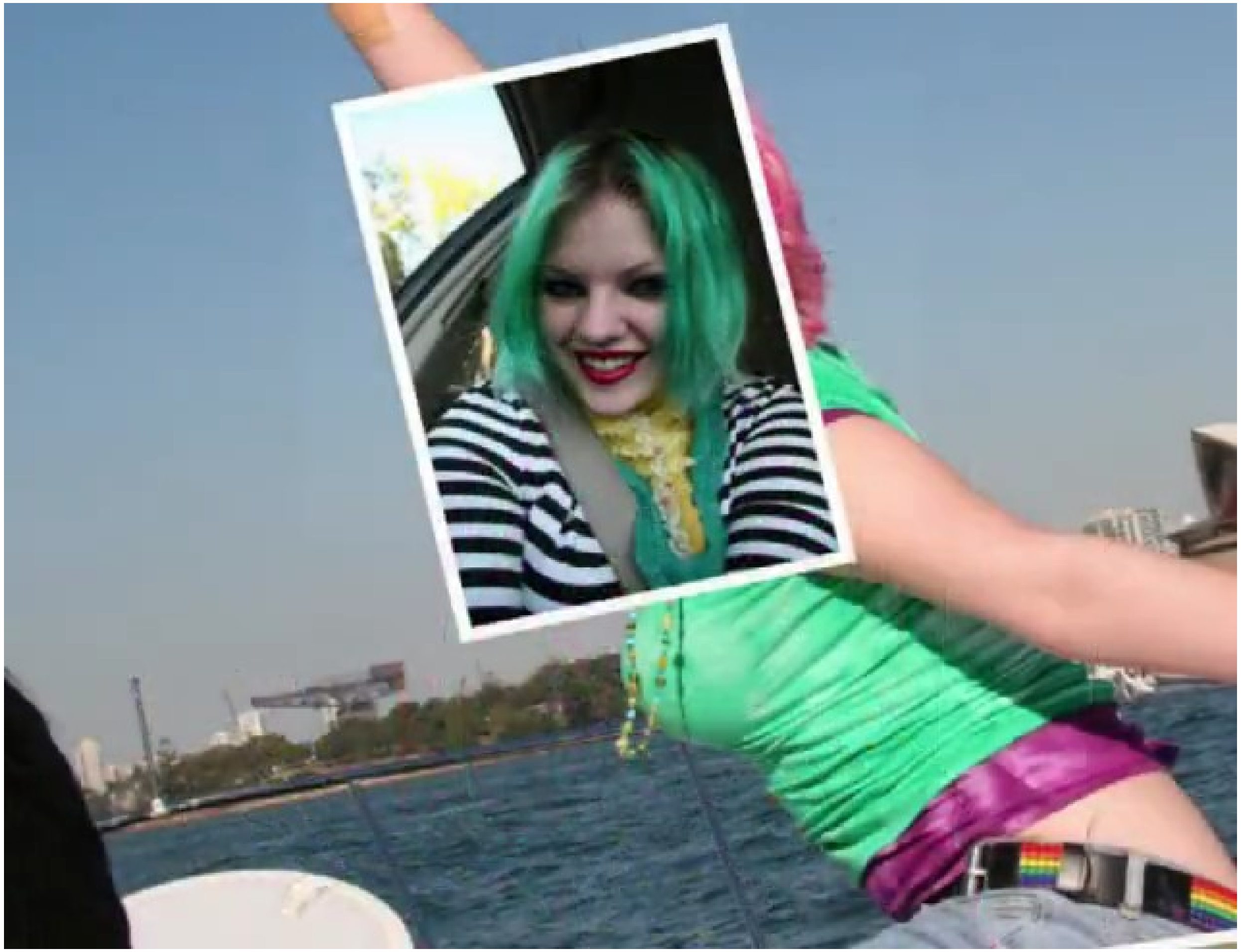
Fade-out one image and fade-in another one simultaneously.

$$I_{out}(t) = (1-t)\, I_{in1} + t\, I_{in2}$$

Output Image

Input Images

Strong Motion Effect by Cross Dissolve



How Could Cross Dissolve Produce Motion?

- **Edge Motion:**
  - *Image edges* move smoothly, with nonlinear *ease-in ease-out* dynamics
- **Physical illumination changes:**
  - The light source direction moves realistically during the transition

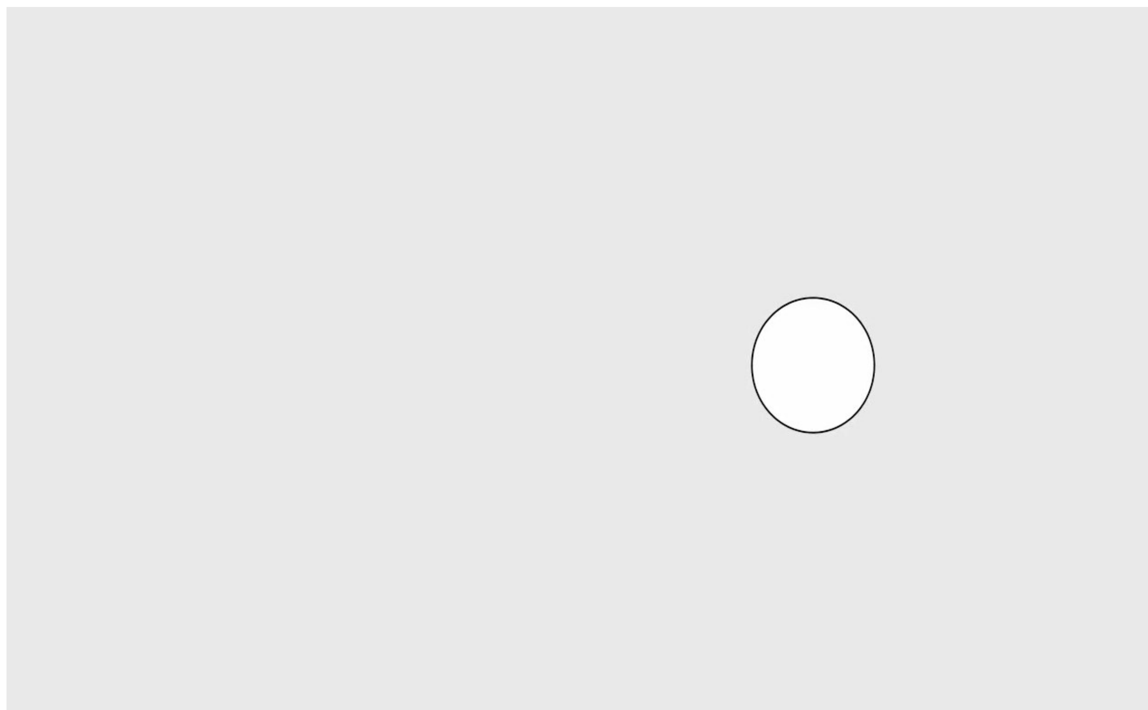
Basics: Image Edges

Points on a digital image where the brightness changes sharply (i.e. has discontinuities)

Image Source: http://en.wikipedia.org/wiki/File:EdgeDetectionMathematica.png

Basics: Ease-in vs. Ease-out

Few drawings → faster action
More drawings → slower action

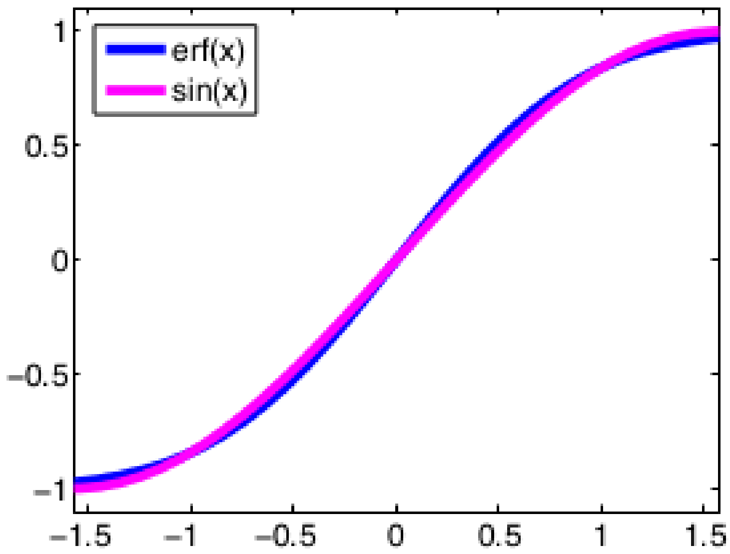


Video Source:
http://www.youtube.com/watch?v=yQ-NC0bHTYs

Edge Approximation

Approximate the image edges by the sine function

$\mathrm{erf}(x) = \int_0^x e^{-t^2}\, dt$



Edge Motion

- Cross dissolving 2 images (signals) represented by

  $\alpha\ sin(mx)$

  $sin(mx+d)$

  - d is phase shift (spatial translation)
  - α is amplitude scale

$$\mathbf{I_{out}\,(t) = (1\text{-}t)\ I_{in1} + t\ I_{in2}}$$

$$(1\text{-}t)\ \alpha\ \sin(mx) + t\ \sin(mx+d) = c\ \sin(mx+k)$$

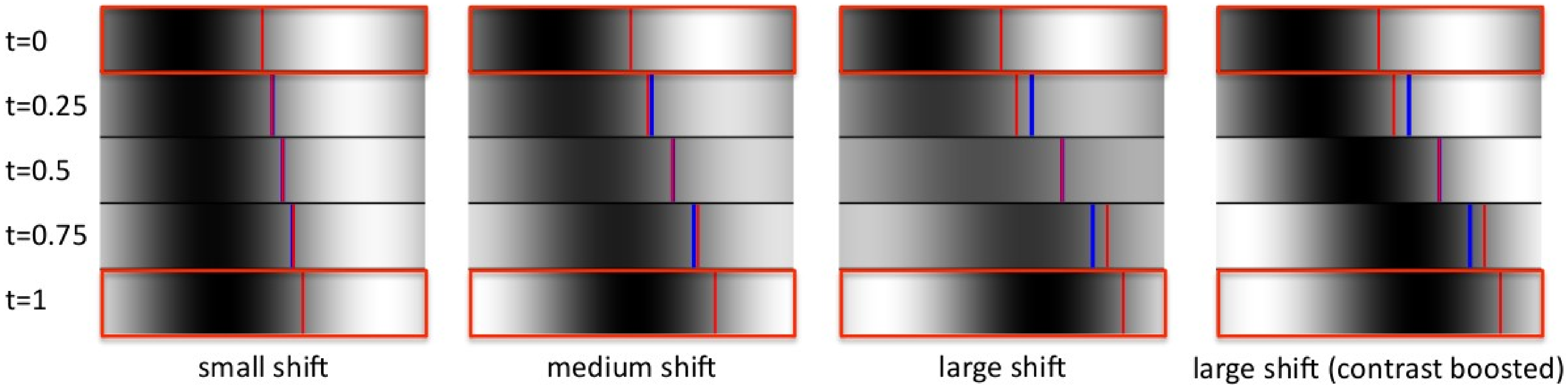
$$t \in [0,1]$$

Important Results

$k = \arctan(t \sin d / ((1-t)\alpha + t \cos d))$

- (1) Phase k is smoothly interpolated. The speed of the motion is determined by phase k.
- (2) k is not linear, but, it perfectly resembles the ease-in ease-out curve (i.e. more believable animations)!

Cross Dissolve vs. Linear Motion



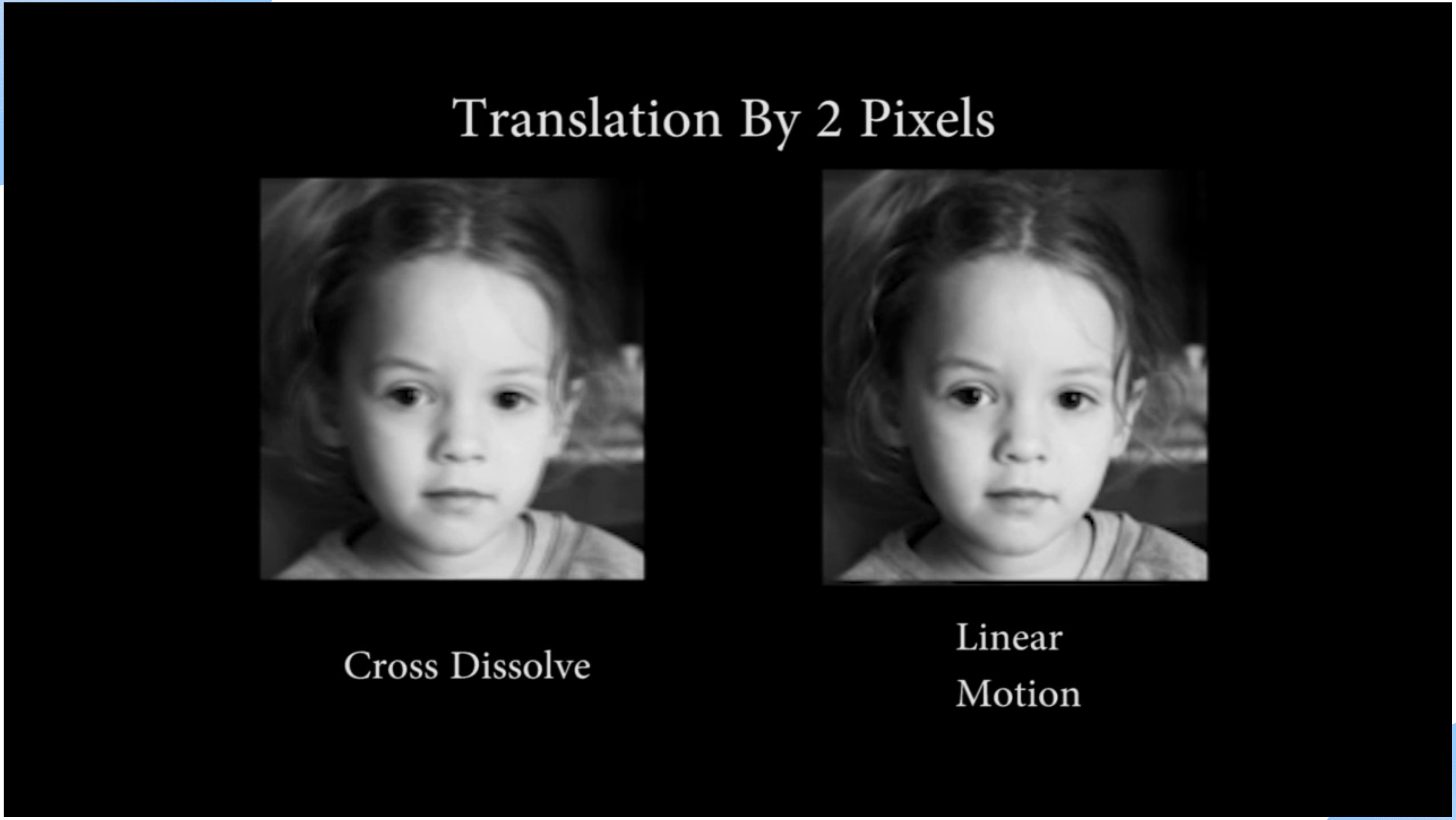
- Location of the edges:
  - in cross dissolve → red
  - in linear motion → blue
- Larger shifts → non-linear ease-in ease-out + decrease in contrast
- Small shifts → imperceptible

Important Results (3)

$(1-t)\,\alpha \sin(mx) + t \sin(mx+d) = c \sin(mx+k)$

- Low frequency edges can move relatively large distances, while high frequency edges can move only slightly.

Cross Dissolve vs. Linear Motion



Important Results (4)

$(1-t)\,\alpha \sin(mx) + t \sin(mx+d) = c \sin(mx+k)$

•When the phase offset reaches π, the edge disappears entirely. This fading away during a transition is called ghosting).

Important Results (5)

$$c^2 = \alpha^2(1-t^2) + t^2 + 2(1-t)\alpha\, t \cos d$$

- According to c in the above equation:
  drop in amplitude of sine → gradual decrease in image contrast

Important Results (6)

The motion effect only works for edges with approximately the same frequency.

Interpolation of Light Sources

In addition to edge motion, cross dissolve could also produce very convincing illumination changes:

- The light source direction appears to move realistically!
- Mathematical details available in the paper

Automation

- The pipeline is almost fully automated
- Exception: more than one person on a photo
  - Future work: using face recognition techniques

Implementation

- Google Picasa 3.8
  - The *Face Movies* feature
- Latest: Picasa 3.9
- Install on GNU/Linux using WINE

Outline

- Introduction to illusory motion
- Exploring Photobios (Shlizerman et al. 2011)
- Video Magnification (Wu et al. 2012)
- Summary & Conclusion
- Questions & Discussion

Video Magnification (Wu et al. 2012)

Revealing subtle changes in the videos that are hard or impossible to see with the naked eyes

- Color variation
- Low-amplitude motions (both periodic and non-periodic)

Why?

- The human visual system → limited spatiotemporal sensitivity
- Many signals below this limit → still useful!

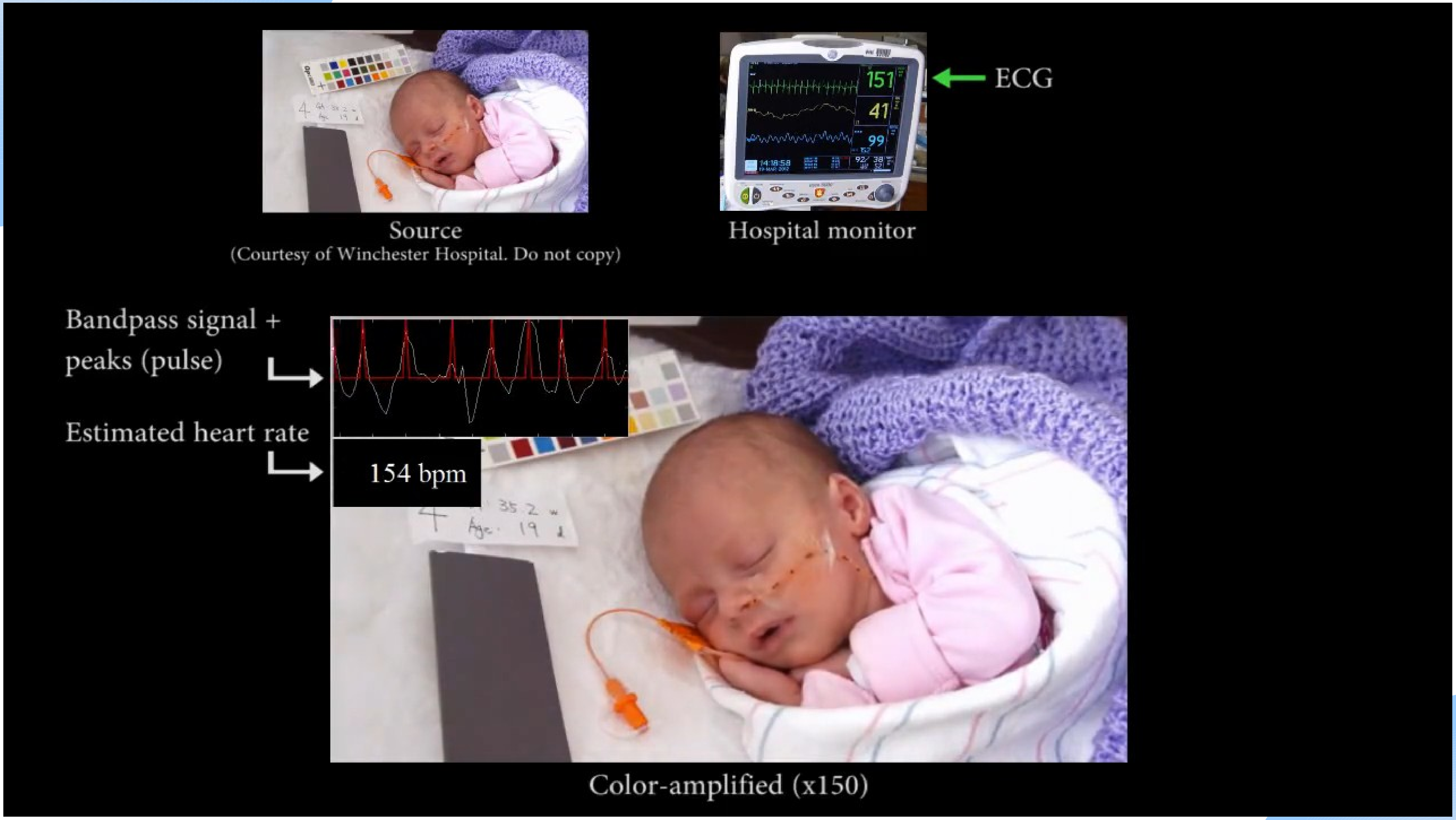
Applications of Video Magnification

- Medicine
- Military
- Architecture
- Law Enforcement
- Etc.

Example 1

Example 2

Example 3



Source
(Courtesy of Winchester Hospital. Do not copy)
Hospital monitor
ECG
Bandpass signal +
peaks (pulse)
Estimated heart rate
154 bpm
Color-amplified (x150)

The Key Idea of This Work

- A combination of spatial and temporal video processing techniques can amplify subtle variations
- To reveal important aspects of the world around us

The Approach

- Consider the time series of color values at any spatial location (i.e. pixel)
- Amplify variation in a given temporal frequency band of interest

Basics: Temporal vs. Spatial Frequency

- Temporal Frequency: No. of occurrences per second
- Spatial frequency

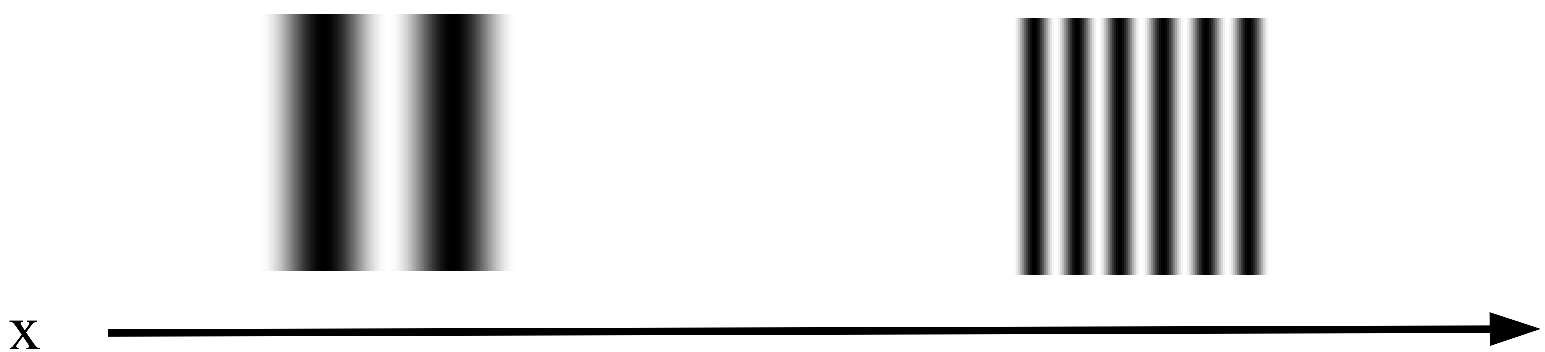


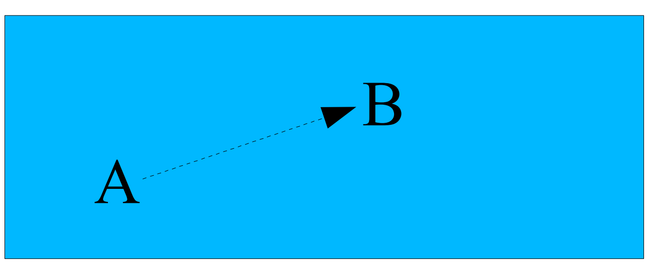
Image source: http://sharp.bu.edu/~slehar/fourier/fourier.html

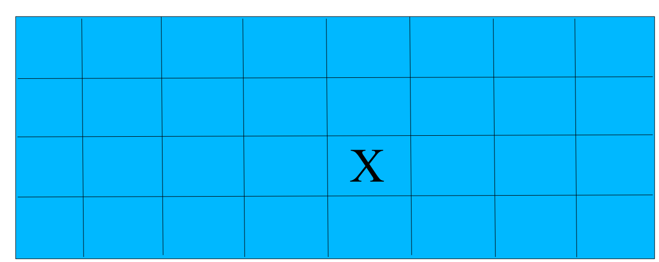
Basics: Fluid Dynamics

- Sub-discipline of Fluid Mechanics
- Deals with fluid flow, i.e. fluids (liquids & gases) in motion
- Sub-disciplines: aerodynamics, hydrodynamics, etc.

2 Main Specifications for Fluid Flow in Fluid Dynamics

- Lagrangian vs. Eulerian
- Lagrangian: Track a particle along its path
- Eulerian: How much fluid passes through a specific point (or cell)





Analogy: The Lagrangian Specification

Image Source:
http://us.123rf.com/400wm/400/400/unnibente/unnibente1112/unnibente111200005/11679189-a-girl-sitting-in-a-small-boat-on-a-river-or-lake.jpg
(with modifications)

Analogy: The Lagrangian Specification

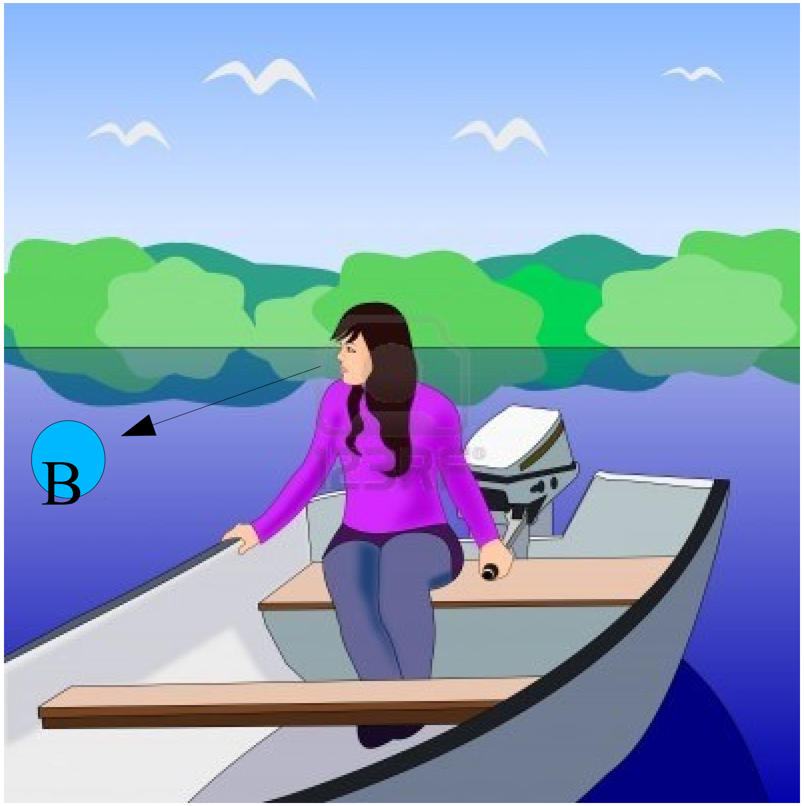


Image Source:
http://us.123rf.com/400wm/400/400/unnibente/unnibente1112/unnibente111200005/11679189-a-girl-sitting-in-a-small-boat-on-a-river-or-lake.jpg
(with modifications)

Analogy: The Eulerian Specification

Image Source: http://8020.photos.jpgmag.com/1227591_193457_e6f5212c66_p.jpg
(with modifications)

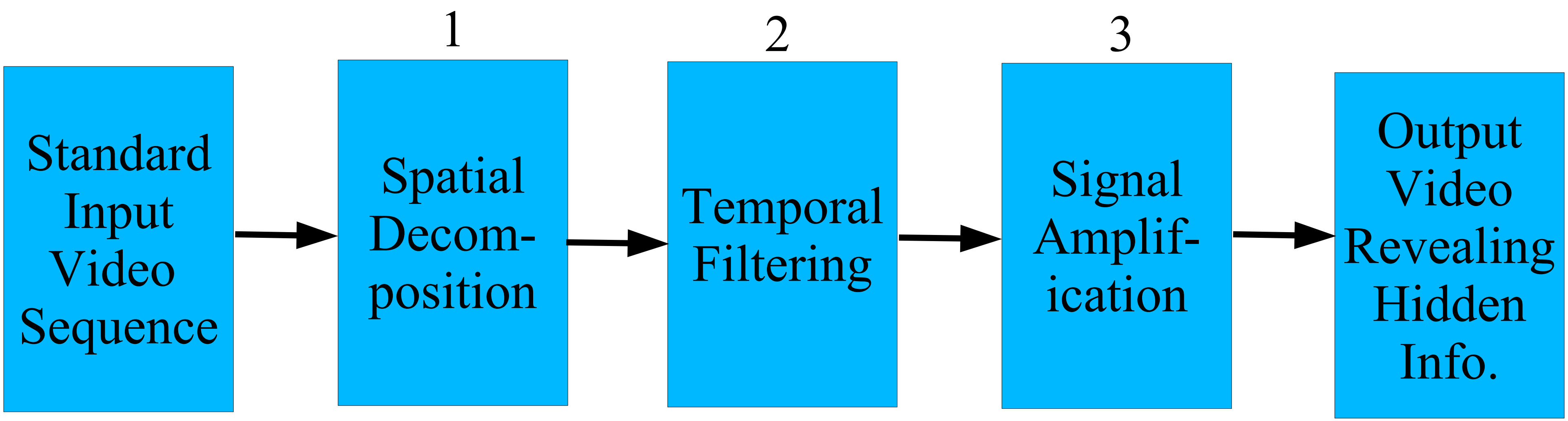
Previous Related Works

- Lagrangian approaches
- Accurate motion estimation
- Computationally expensive

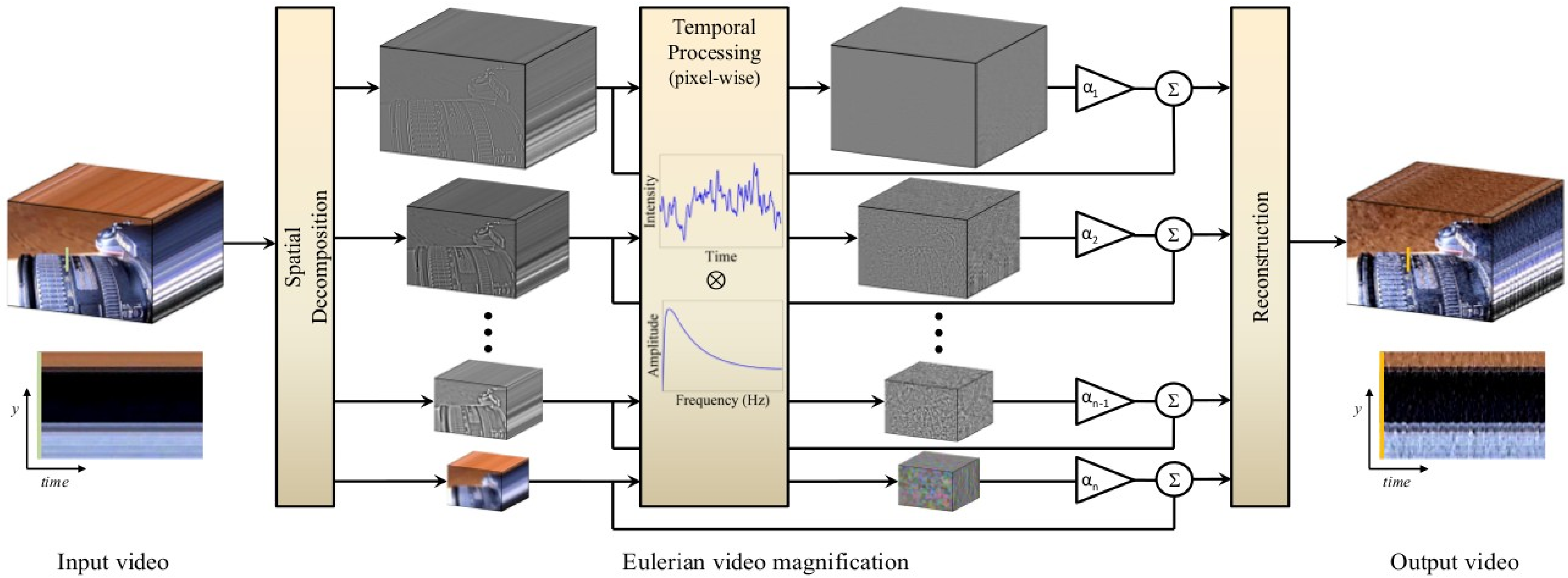
This Work

- Eulerian approach: Each pixel is processed independently
  - Treat each pixel as a time series and apply signal processing to it
- Do not explicitly estimate motion
- Exaggerate motion by amplifying temporal color changes at **fixed positions**
- Robust & real time

Eulerian Video Magnification



Eulerian Video Magnification



1. Spatial Decomposition

- Decompose the video sequence into different spatial frequency bands
- These bands might be magnified differently
  - Might have different signal-to-noise ratios
  - Might have spatial frequencies for which the linear approximation used in motion magnification does not hold

2. Temporal Filtering

- Increase the signal-to-noise ratio
- On each spatial band
- Extract the frequency band of interest(Fourier Theory)
  - E.g. if 24 - 240 beats per minute → only select frequencies of 0.4 - 4 Hz

3. Signal Amplification

- Multiply the signal by an amplification factor α
- Specified by the user
- Add the magnified signal to the original → final output obtained

Spatiotemporal Coherency

- Natural videos are spatially and temporally smooth
- Filtering performed uniformly
- Therefore, coherency implicitly maintained

How temporal processing produces motion magnification?

- $I(x,t)$ = the image intensity at position x at time t

How temporal processing produces motion magnification?

- $I(x,t)$ = the image intensity at position x at time t
- $I(x,0) = f(x)$
- $I(x,t) = f(x+\delta(t))$

Displacement function

How temporal processing produces motion magnification?

- $I(x,t)$ = the image intensity at position x at time t
- $I(x,0) = f(x)$
- $I(x,t) = f(x+\delta(t))$
- The goal of motion magnification is to synthesize the signal:

$$I(x,t) = f(x+\delta(t)) = f(x + (1+\alpha)\delta(t))$$

- for some amplification factor α.

Lagrangian vs. Eulerian Methods

- Lagrangian methods: support larger amplification factors
- Eulerian method: smoother structures & small amplifications

Outline

- Introduction to illusory motion
- Exploring Photobios (Shlizerman et al. 2011)
- Video Magnification (Wu et al. 2012)
- Summary & Conclusion
- Questions & Discussion

Summary: Exploring Photobios

- Generating motion from well-aligned static images using cross dissolving.
- Not only illusion of motion, but real motion!

Summary: Video Magnification

- Revealing and magnifying very small motions & variations using temporal signal processing
- Eulerian (in contrast to Lagrangian) approach → robust and real-time

Conclusion: Similarity

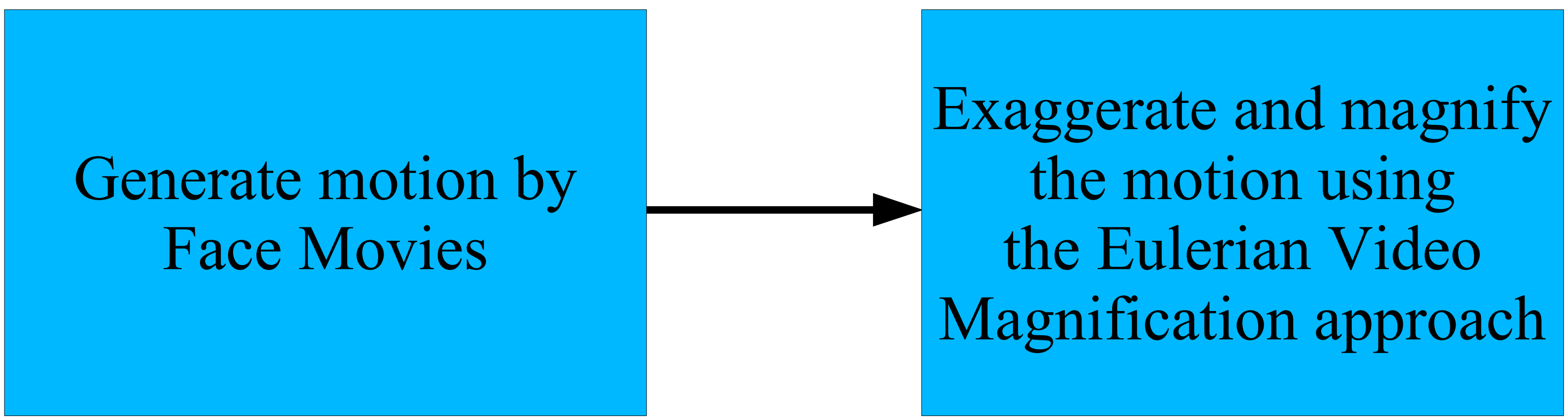
- Both somehow produce kind of *motion*
  - The former: motion from **static** images
  - The latter: Make very small motions visible

Conclusion: Difference

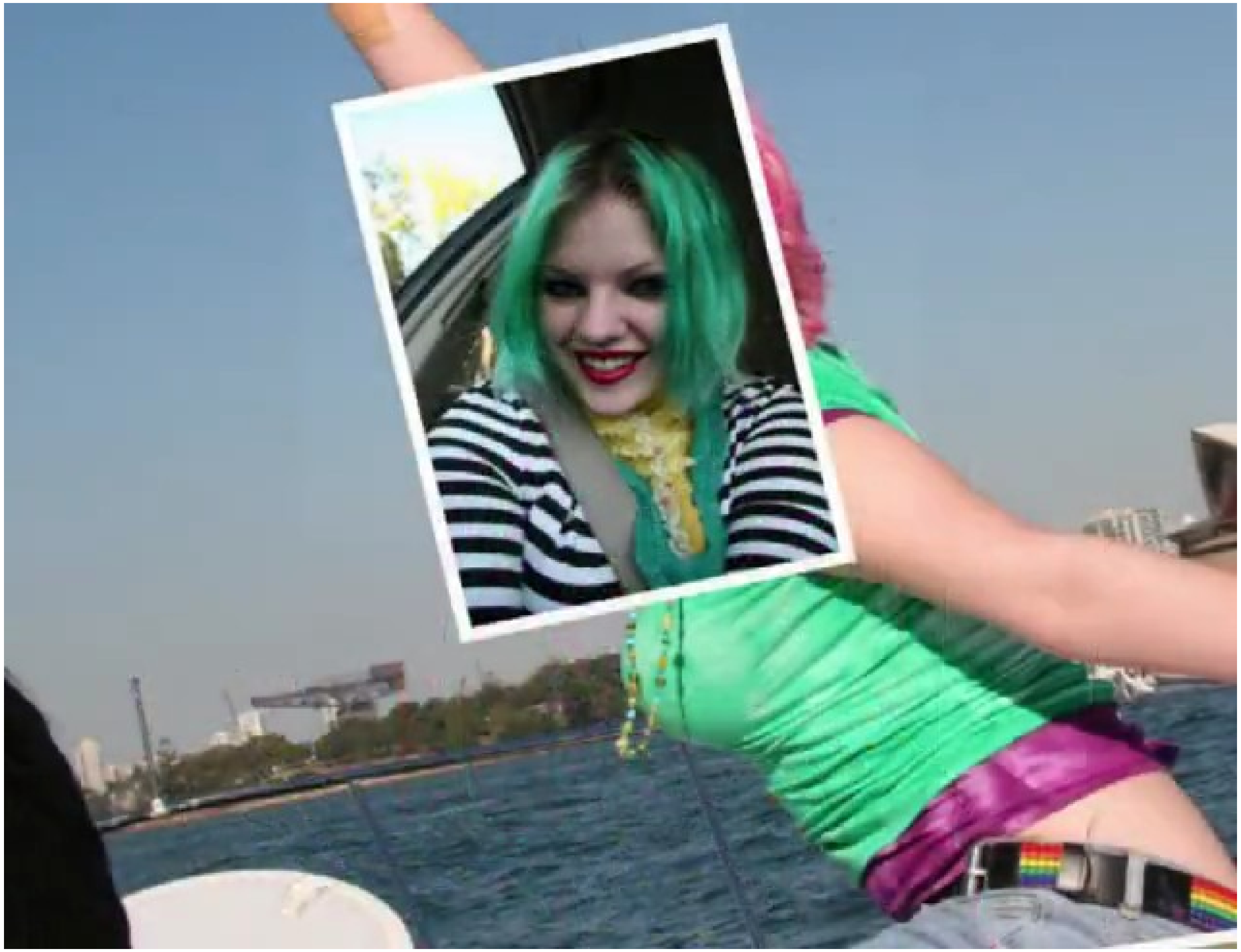
- The former: no motion has been occurred
- In the latter: motion has been occurred, invisible to the naked eye

Conclusion: Idea for Future Work

Idea: Using a combination, e.g.:



Conclusion: Idea for Future Work



Questions & Discussion

- Acknowledgements: Thanks James Tompkins
- Thanks for your patience!

- References:

  - Exploring Photobios, Kemelmacher-Shlizerman et al., SIGGRAPH 2011
  - Eulerian Video Magnification, Wu et al., SIGGRAPH 2012