Background Removal

CS4243 Computer Vision and Pattern Recognition

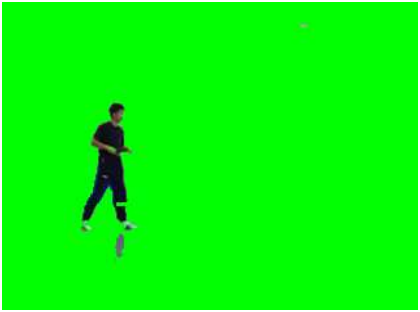
Leow Wee Kheng

Department of Computer Science
School of Computing
National University of Singapore

Introduction

Background removal removes background in video to expose the foreground moving objects.

(a) input



(b) result

Background removal is related to tracking and motion segmentation.

- Motion tracking tracks moving objects in video.
  It can be confused if background moves or changes color.
- Motion segmentation separates objects from background based on motion difference.
- Background removal removes background based on statistical difference without tracking.

Two variations of background removal:

- With known background, i.e., clean plate.
- Without known background.

Background Removal With Clean Plate

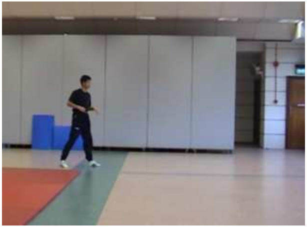
Background removal with clean plate is relatively easy.

Method:

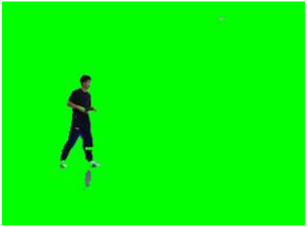
- Compute absolute color difference between image and clean plate.
- Pixel with small difference: background pixel.
- Pixel with large difference: foreground pixel.

Example:

(a) clean plate



(b) input



(c) result

- Background in result is set to green for visual clarify.
- Some foreground pixels are removed because they have similar colors as corresponding background pixels.

Notes:

- If camera is stationary and lighting is well controlled, needs only one clean plate.
  (example: man.mpg)
- If camera is moving and lighting is well controlled, needs motion-controlled camera to shoot the clean plate video.
  (example: chess.mpg)
- If lighting is not well controlled, then need more general methods.

Background Removal Without Clean Plate

Background removal without clean plate is more difficult.

Possible if moving object does not occupy the same position all the time.

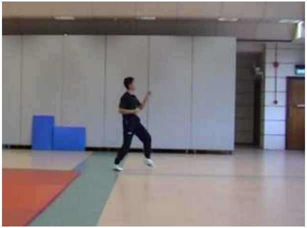
Cases:

1. Stationary camera, fixed lighting condition
2. Stationary camera, varying lighting condition
3. Moving camera

Stationary Camera, Fixed Lighting

Consider this example:

frame 2500

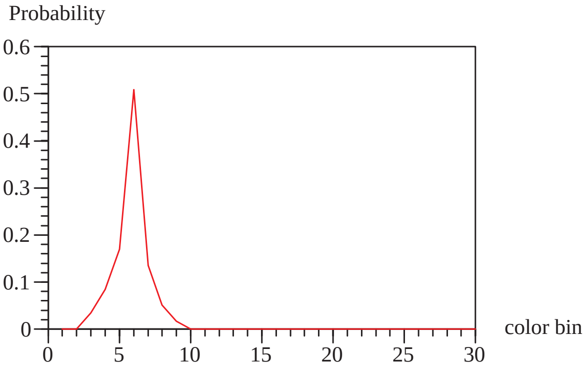


frame 2600

frame 2700

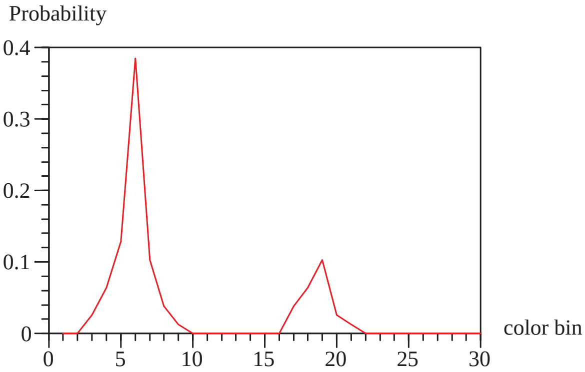
- Moving object occupies a small area of the image.
- Moving object does not stay at the same position for a long time.
- What is the color distribution of a particular pixel?

For a background pixel that is never covered by moving object:



- Peak color is most frequently occurring color of background pixel.

For a background pixel that is sometimes covered by moving object:



- Higher peak: most frequently occurring color of background pixel.
- Lower peak: most frequently occurring color of moving object.
- Relative height of two peaks depends on how long the moving object covers the background.

To separate the two peaks, apply clustering algorithm.

- Each cluster C_i is represented by a cluster prototype or cluster center $\mathbf{w}_i$.
- Need an appropriate difference measure between pixel color $\mathbf{x}$ and cluster center $\mathbf{w}_i$, e.g., Euclidean distance:

$$d(\mathbf{x}, \mathbf{w}_i) = \|\mathbf{x} - \mathbf{w}_i\| \tag{1}$$

- The most popular clustering algorithm is k-means clustering.

k-Means Clustering

Choose k initial cluster centers $\mathbf{w}_1(0), \ldots, \mathbf{w}_k(0)$.
Repeat until converge

Distribute each color $\mathbf{x}$ to the nearest cluster C_i:

$$\mathbf{x} \in C_i(t) \quad \text{if } \|\mathbf{x} - \mathbf{w}_i\| < \|\mathbf{x} - \mathbf{w}_j\| \ \forall j \neq i$$

Compute new cluster centers:

$$\mathbf{w}_i(t+1) = \frac{1}{|C_i(t)|} \sum_{\mathbf{x} \in C_i(t)} \mathbf{x}$$

- In theory, k-means clustering can always converge.
- In practice, it may take many iterations to converge.
- Possible practical termination conditions:
  - Terminate after very few inputs change clusters.
  - Terminate after a fixed number of iterations.

For background removal

- Can choose $k = 2$.
- Choose initial cluster centers using heuristics, e.g., extreme colors.

After clustering,

- If a pixel falls in a foreground cluster, then it's a foreground pixel.
- If a pixel falls in a background cluster, then it's a background pixel.

Questions:

1. What if there is only one cluster, i.e., no foreground?
2. How to determine which cluster a pixel falls in?

Probability Density Estimation

- Build a color model for each pixel.
- Simplest model: Gaussian distribution.

1-dimensional Gaussian:

$$G(x) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right) \tag{2}$$

- μ: mean

$$\mu = \frac{1}{N}\sum_{i=1}^{N} x_i \tag{3}$$

- σ: standard deviation

$$\sigma^2 = \frac{1}{N}\sum_{i=1}^{N} (x_i - \mu)^2 \tag{4}$$

n-dimensional Gaussian:

$$G(\mathbf{x}) = \frac{1}{\sqrt{(2\pi)^n|\boldsymbol{\Sigma}|}} \exp\left(-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^T\boldsymbol{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu})\right) \tag{5}$$

- $\boldsymbol{\mu}$: mean vector

$$\boldsymbol{\mu} = \frac{1}{N}\sum_{i=1}^{N}\mathbf{x}_i \tag{6}$$

- $\boldsymbol{\Sigma}$: covariance matrix

$$\boldsymbol{\Sigma} = \frac{1}{N}\sum_{i=1}^{N}(\mathbf{x}_i-\boldsymbol{\mu})(\mathbf{x}_i-\boldsymbol{\mu})^T \tag{7}$$

- normalization:

$$\int_{-\infty}^{+\infty} G(\mathbf{x})d\mathbf{x} = 1 \tag{8}$$

Notes:

- In practice, $\mathbf{x}$ is discretized.
- A convenient method to ensure normalization is

$$g(\mathbf{x}) = \exp\left(-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^T\boldsymbol{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu})\right) \quad (9)$$

$$G(\mathbf{x}) = \frac{g(\mathbf{x})}{\sum_{\mathbf{x}} g(\mathbf{x})} \quad (10)$$

Advantage of probability over Euclidean distance:

- Euclidean distance $\|\mathbf{x}-\mathbf{w}_i\|$ is based only on mean.
- Probability $G(\mathbf{x})$ is based on both mean and covariance.

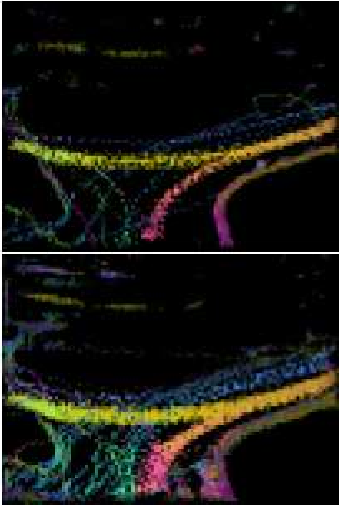
Is there a distance measure that is based on both mean and covariance?

Stationary Camera, Varying Lighting

Basic idea:

- There are multiple peaks in the color distribution.
- If moving object covers background pixel for only a very short time, then moving object's colors have very small peaks.
- So, can do clustering to identify large clusters.
  These are the background clusters.
- Model each cluster using a Gaussian, or
- Model the background colors using a mixture of Gaussians [SG98].
  Take CS6240 to learn about Gaussian mixture.

Example from [SG98]: Lighting condition changes over time.



Background Removal With Moving Camera

General idea:

- Perform motion tracking to recover camera motion.
- Stabilize video by removing camera motion [BAHH92, MOTS05].
- Perform background removal as for the case of stationary camera.
- Destabilize video, i.e., put back camera motion.

Reference

- J. R. Bergen, P. Anandan, K. J. Hanna, and R. Hingorani.
  Hierarchical model-based motion estimation.
  In *Proc. ECCV*, pages 237–252, 1992.
- Y. Matsushita, E. Ofek, X. Tang, and H.Y. Shum.
  Fullframe video stabilization.
  In *Proc. CVPR*, volume 1, pages 50–57, 2005.
- C. Stauffer and W. E. L. Grimson.
  Adaptive background mixture models for real-time tracking.
  In *Proc. IEEE Conf. on CVPR*, 1998.