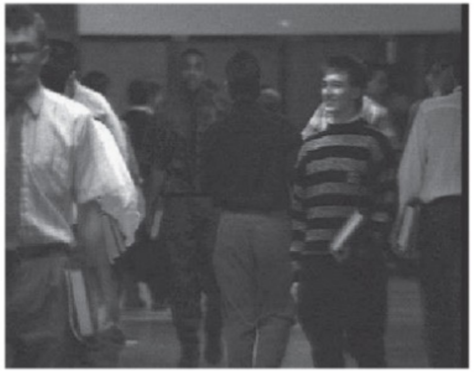
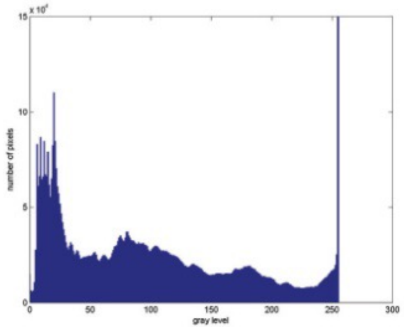
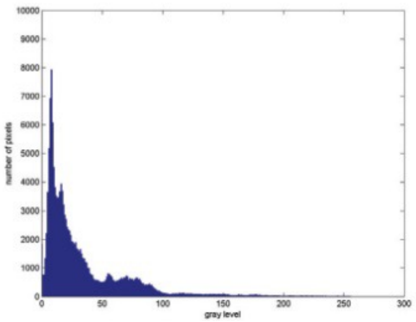
(3.3) Gray level histograms

Histograms give a simple but powerful technique for capturing the statistics of any type of data.

- The histogram records the number of occurrences (pixels) in each bin (with a specific value). For a [0,255] , 8bpp image we have 256 bins, one for each gray level. Can also look at gray level intervals (fewer bins).
- Gives no information on **where** in the image the gray levels occur







Normalized histogram: computed from the histogram by dividing each value of the histogram by the total number of pixels in the image.

This is an estimation of the **Probability density function (PDF)**

$H(l) = n_l$ → n_l histogram value of gray level *l*

w → *width of the image*
h → *height of the image*
n_pix → number of pixels in the image

$\sum n_l = n = w*h$

Normalize(H(l)) = H(l)/n

Cumulative distribution function (CDF):

$$\boxed{\bar{c}[\ell] = \sum_{k=0}^{\ell} \bar{h}[k], \quad \ell = 0, \ldots, 255}$$

Histogram Equalization:

A pixel with gray level l gets new gray level l' according to:

$$\boxed{l' = round(255 * \bar{c}[l])}$$

Cumulative distribution (CDF):function

$$\bar{c}[\ell] = \sum_{k=0}^{\ell} \bar{h}[k], \quad \ell = 0, \ldots, 255$$

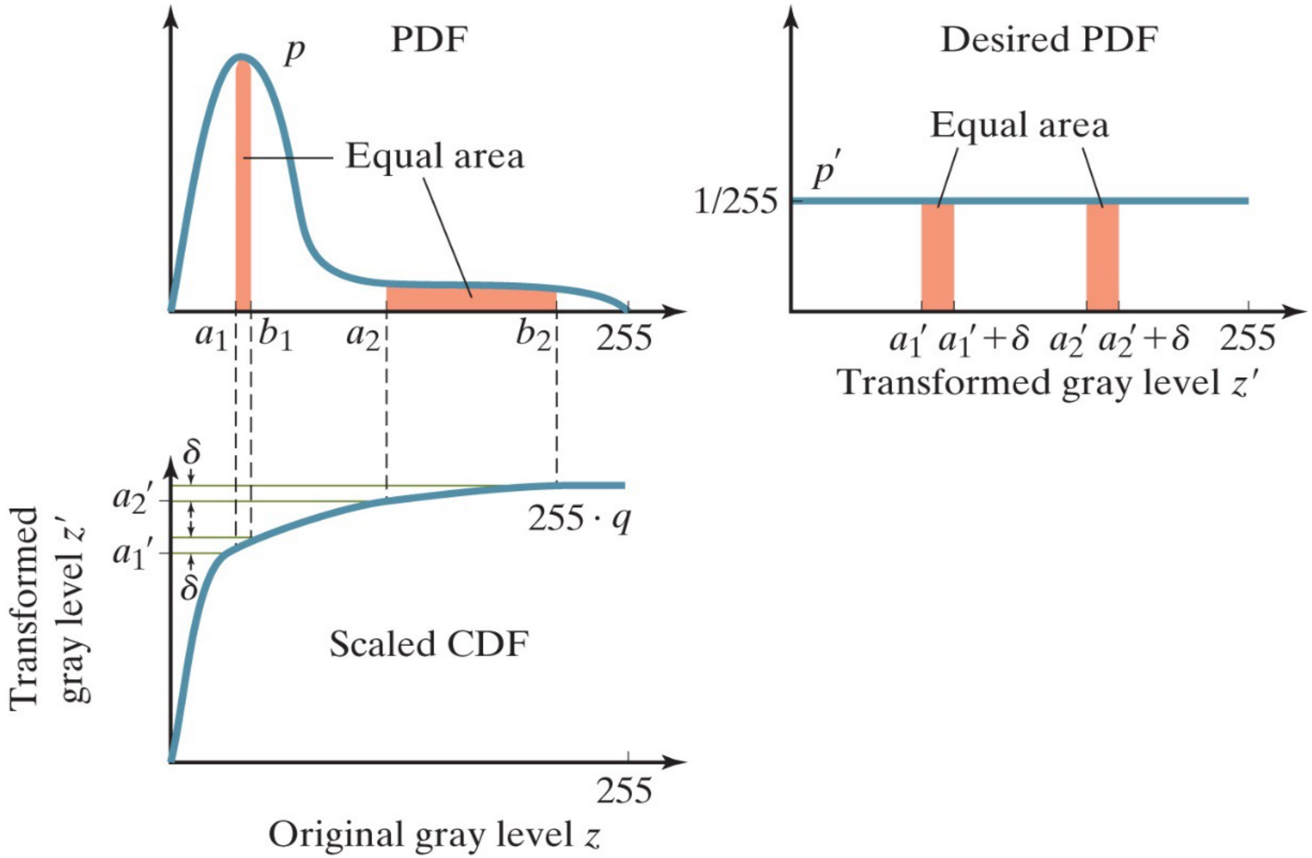
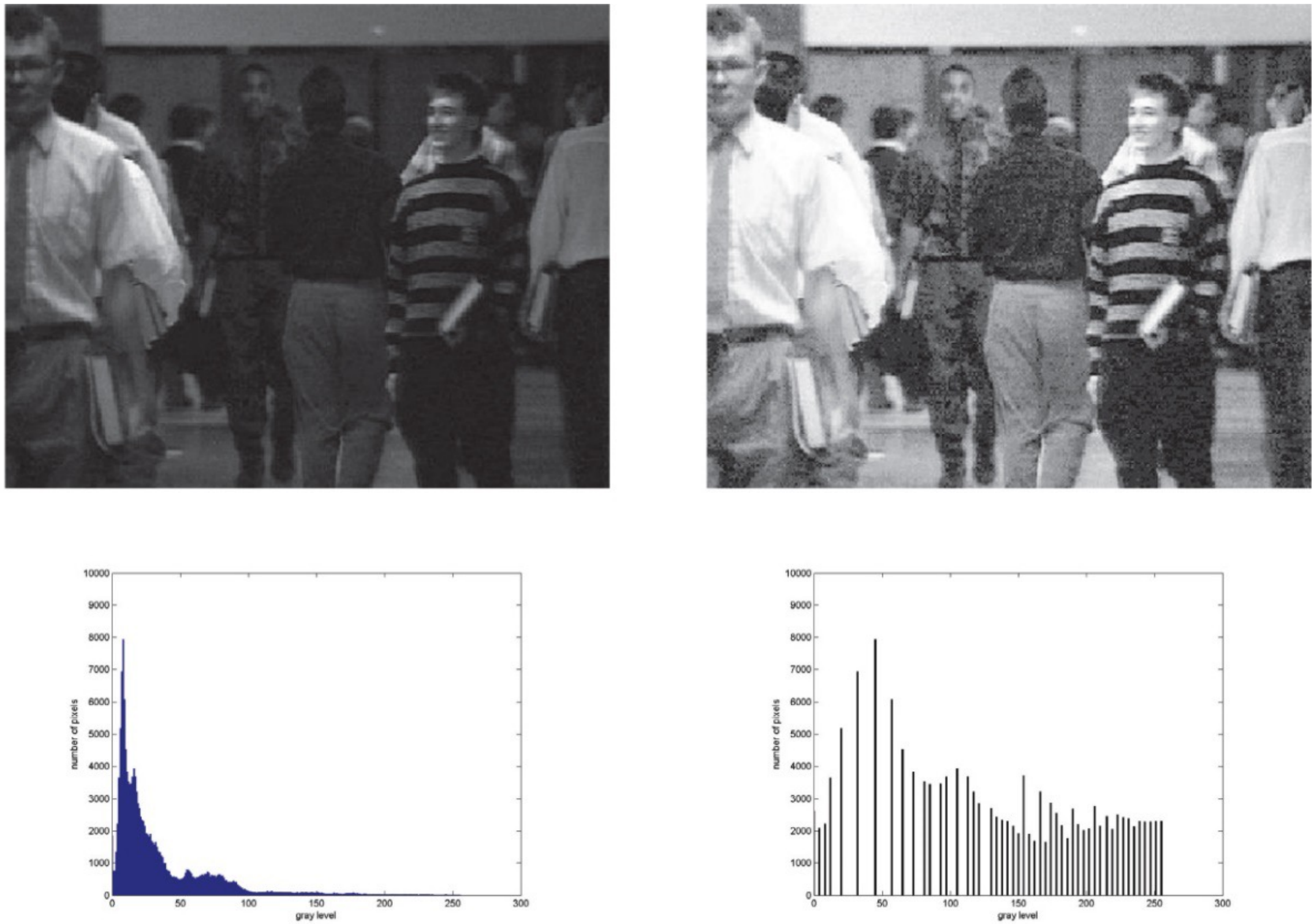


Figure 3.21 The result of histogram equalization applied to an image. The increase in contrast is noticeable. The normalized histogram of the result is much flatter than the original histogram, but it is not completely flat due to discretization effects. Source: Movie *Hoop Dreams*.



Try yourself

For the toy-image I given below :

1) Find histogram and normalized histogram
2) Perform histogram equalization (3 bpp -> [0, .. 7]

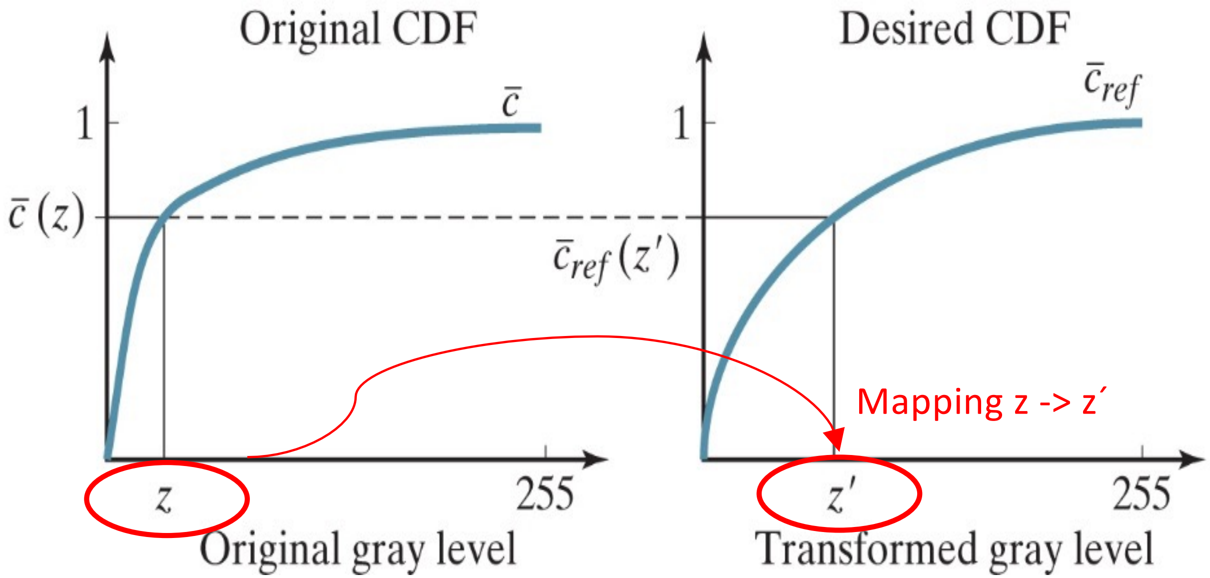
$$I = \begin{bmatrix} 7 & 4 & 2 & 0 \\ 4 & 2 & 4 & 5 \\ 3 & 3 & 5 & 6 \end{bmatrix}$$

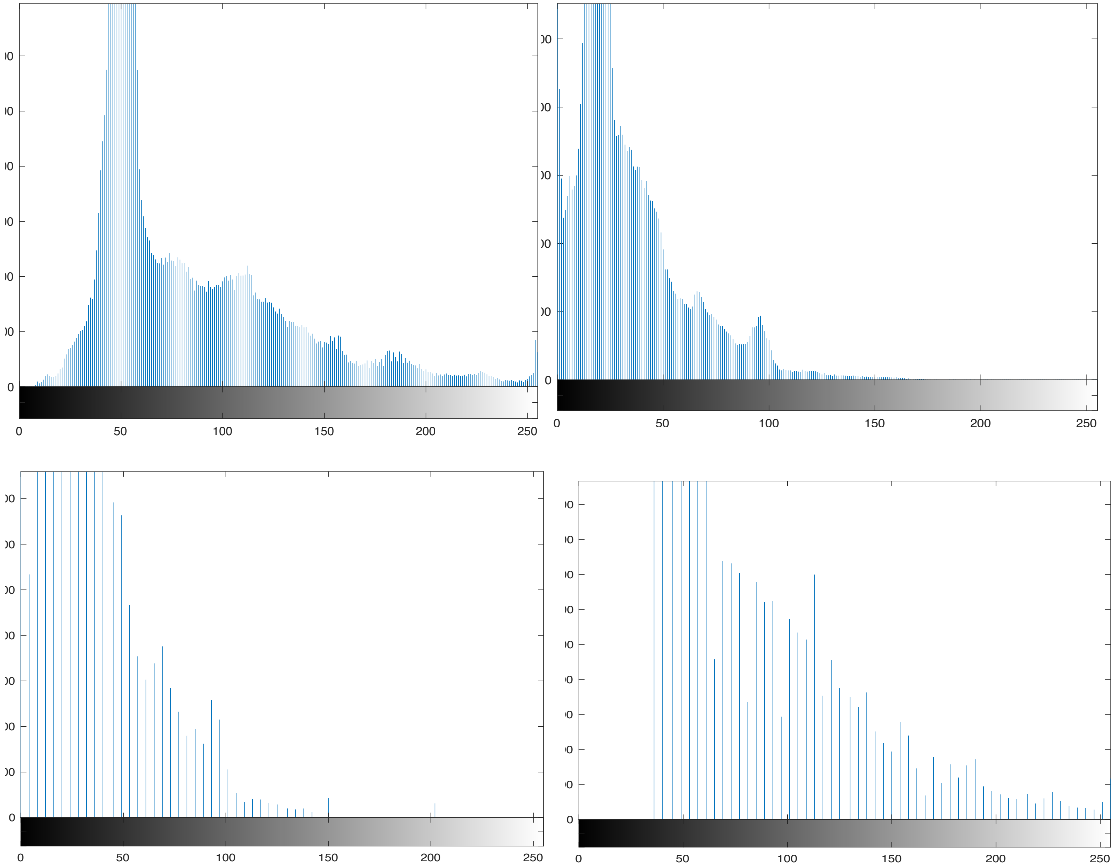
Histogram Matching

- Histogram equalization flattens the histogram as much as possible. Sometimes we rather match it to another (reference) histogram to make images comparable. (to compensate for changing light conditions etc.) -> histogram matching.
- Again, the cumulative distribution function is used (CDF)

Histogram Matching

Figure 3.23 Histogram matching. Given the CDF $\bar{c}$ of the original image and the desired CDF $\bar{c}_{ref}$, histogram matching transforms an original gray level z to a new gray level z' by finding the value of z' such that $\bar{c}(z) = \bar{c}_{ref}(z')$. As before, discretization effects are ignored in this illustration.




0 50 100 150 200 250
0 50 100 150 200 250
0 50 100 150 200 250
0 50 100 150 200 250

(3.4) Multispectral transformations

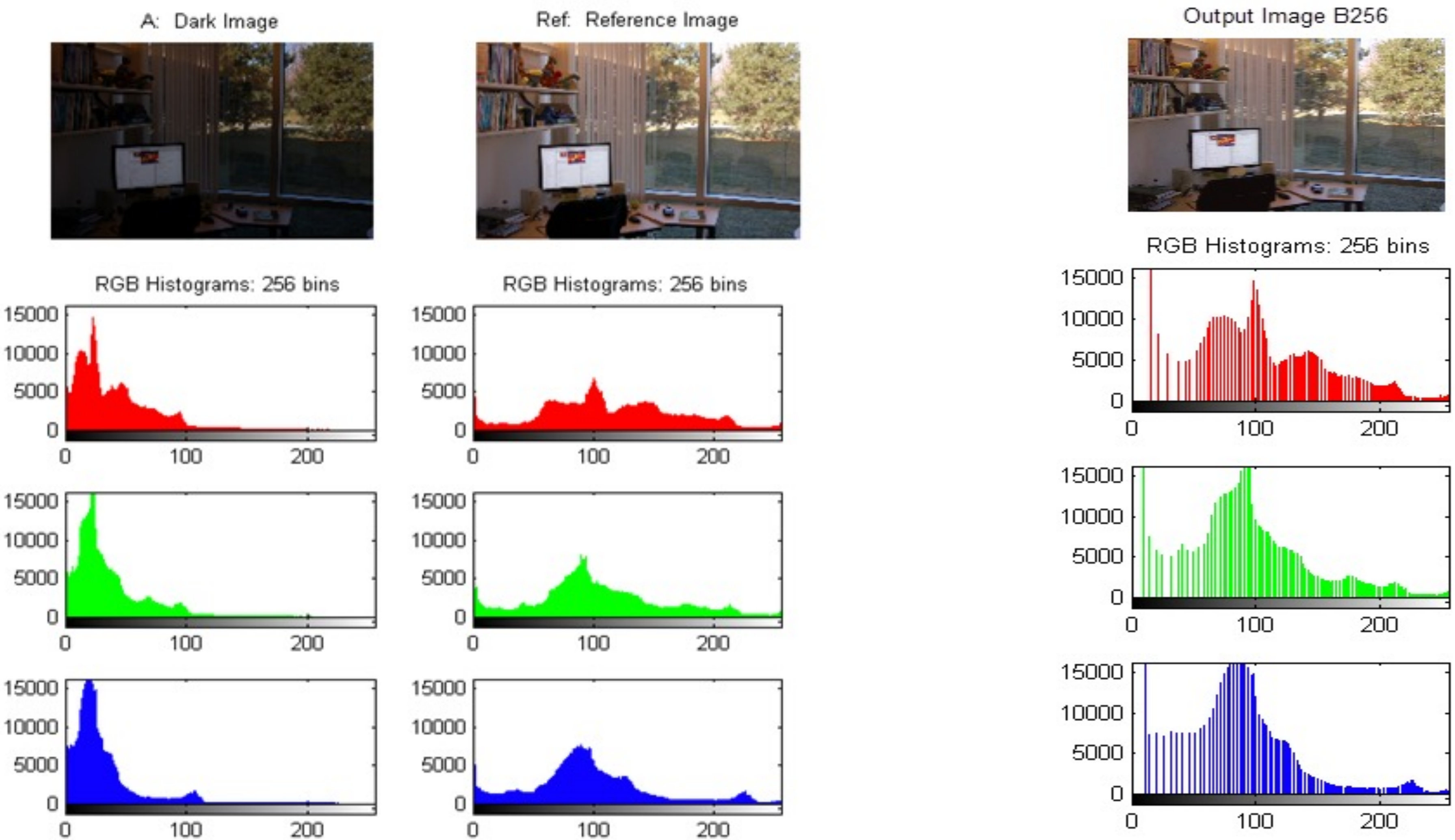
- Multiple values for each pixel (ex. color RGB, or other)
- A mapping where either the input, the output or both images are multispectral is called a multispectral transformatin.
- Simple RGB to gray:

$$I'(x,y) = \frac{1}{4}[I_R(x,y) + 2I_G(x,y) + I_B(x,y)]$$

- Simple gray to RGB:

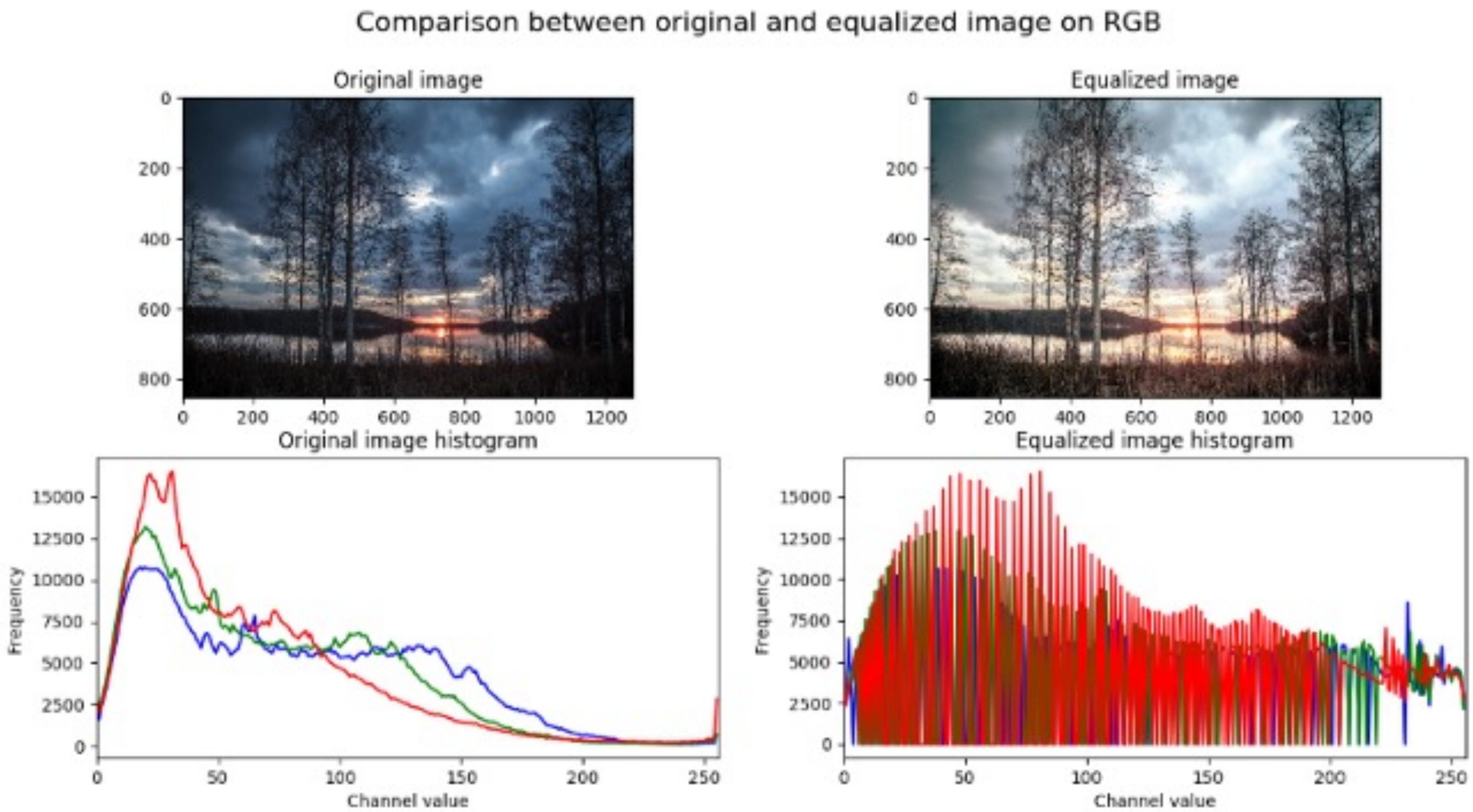
$$I'_R(x,y) = I(x,y), \quad I'_G(x,y) = I(x,y), \quad I'_B(x,y) = I(x,y)$$

Multispectral histogram matching

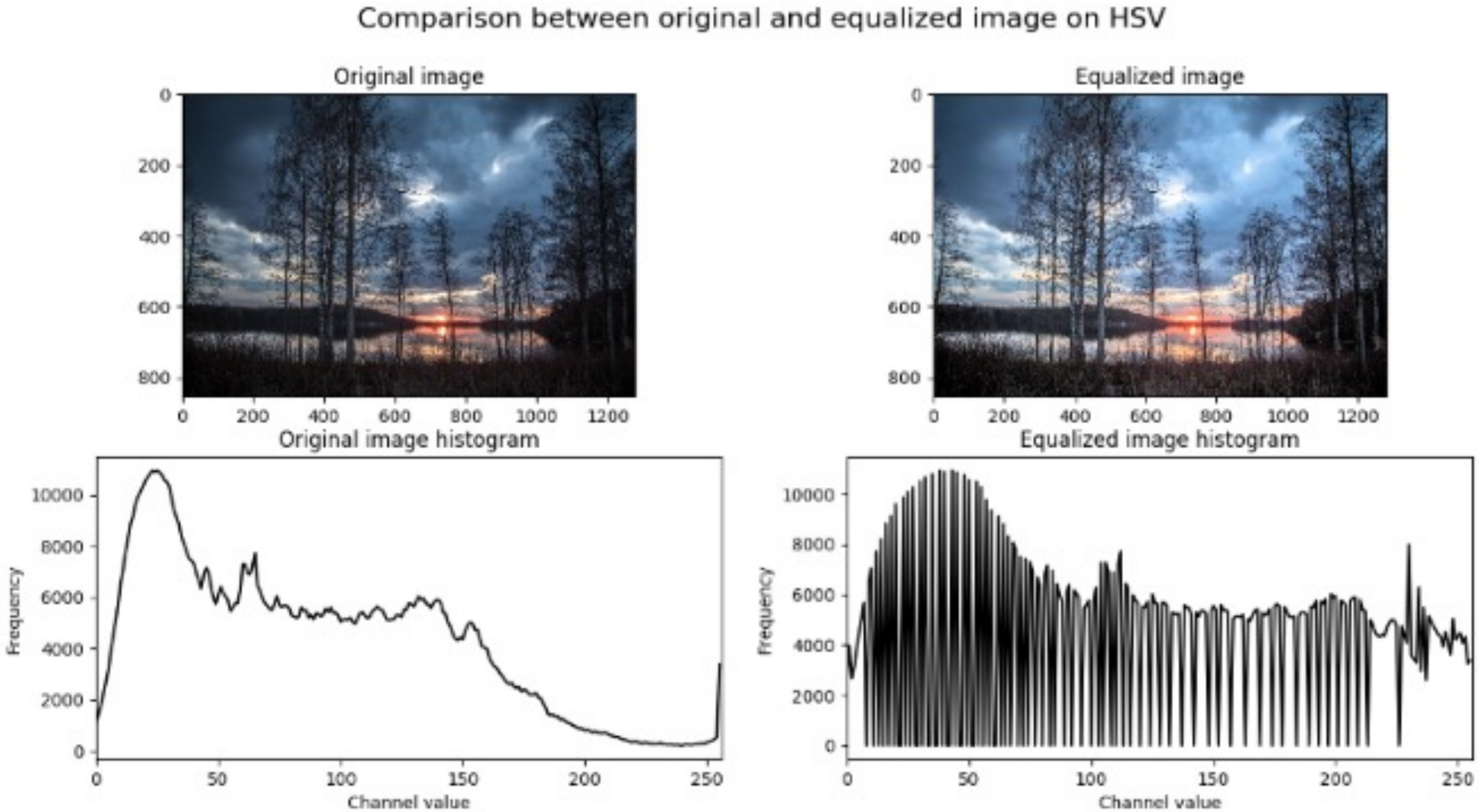


Mulitspectral histogram operations

- If done directly on RGB each channel – might produces some strange colors and effects.
- Can be better to do RGB -> HSV
- Do histogram operations on the Value (Intensity) channel , keep Hue (color) and saturation constant.



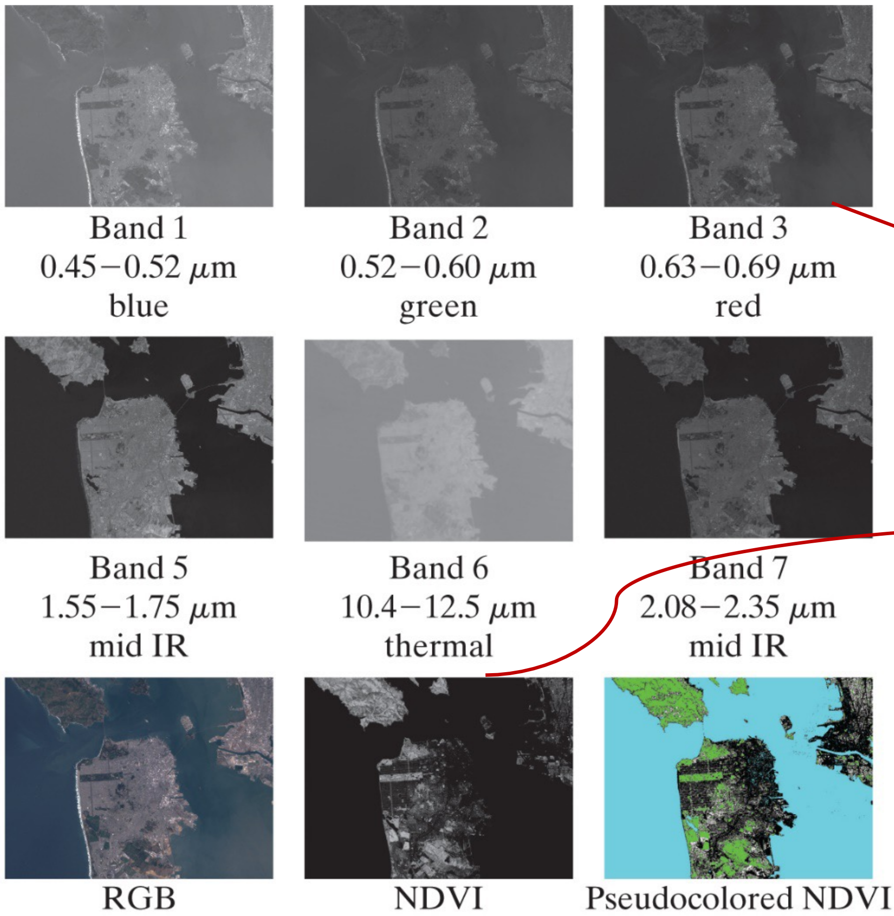
https://medium.com/@rndayala/image-histograms-in-opencv-40ee5969a3b7



https://medium.com/@rndayala/image-histograms-in-opencv-40ee5969a3b7

Multispectral images – LANDSAT

Figure 3.26 7 bands of a Landsat image of the San Francisco Bay Area. The bottom row shows the RGB image obtained by combining bands 1, 2, and 3; the NDVI calculated using bands 3 and 4; and the pseudocolored image obtained by density slicing on the NDVI (blue indicates water, green indicates vegetation, and tan indicates soil). Notice the vegetation occurs outside the city itself in Marin County (upper peninsula in the image) and several parks (namely, the Presidio, Golden Gate Park, and San Bruno Mountain State Park on the lower peninsula). Source: http://glcf.umd.edu, http://glcf.umd.edu/data/landsat/



$$\boxed{\mathrm{NDVI} = \frac{I_{IR}(x,y) - I_R(x,y)}{I_{IR}(x,y) + I_R(x,y)}}$$

Detect vegetation, ratio of red to the near-infraread

(3.5) Multi-Image (point) Transformations

- **Multi-image transformation:** includes images taken of the same scene at different times, or of different scenes entirely.
- logical operations (and, or, not, xor)
- arithmetic operations; examples:
  - **Absolute difference:** computes for each pixel location the absolute value of the difference between the pixels:

$$I'(x, y) = |I_1(x, y) - I_2(x, y)|$$

  - **Linear interpolation:** produces a convex combination of the two inputs:

$$I'(x, y) = \eta I_1(x, y) + (1 - \eta)I_2(x, y)$$

(3.6) Change detection – frame differencing

Difference image: to compare successive image frames in the video sequence.

$$I'(x,y) = |I_t(x,y) - I_{t-1}(x,y)| > \tau$$

Will this work?

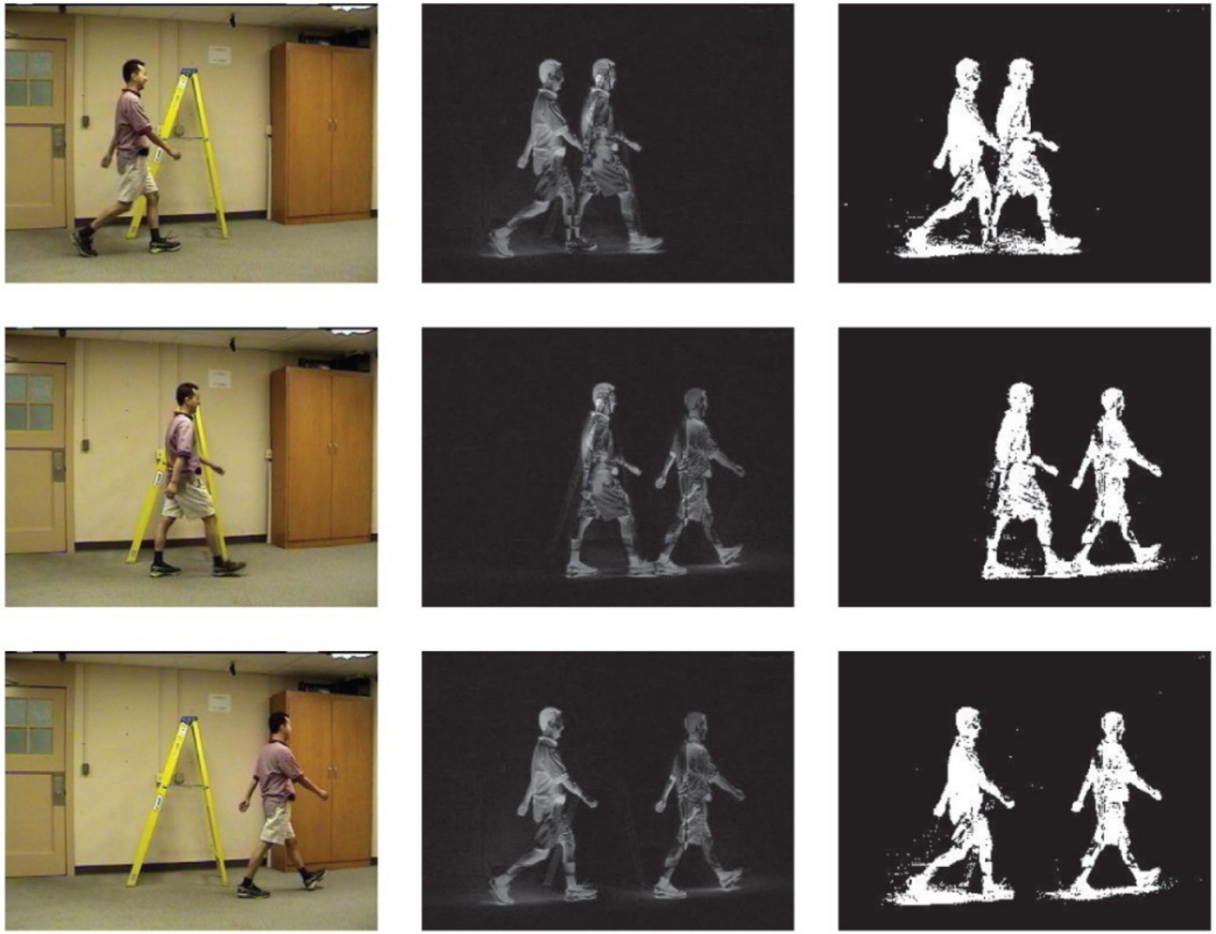
Input images

(3.6) Change detection – frame differencing

Difference image: to compare successive image frames in the video sequence.

$$I'(x, y) = |I_t(x, y) - I_{t-1}(x, y)| > \tau$$

double-image problem: the difference image will contain foreground pixels not only where the foreground object is located in the current frame but also where it was in the previous frame



Input images Absolute difference Thresholded

Solution:

- **Double-difference image:** used to solve the double-image problem, use three input images:

$$I' = |I_t - I_{t-1}| > \tau \text{ AND } |I_{t+1} - I_t| > \tau$$

- **Triple-difference image:** combines the absolute differences from all three image pairs using addition and subtraction prior to thresholding:

$$I'(x, y) = (|I_{t-1} - I_t| + |I_{t+1} - I_t| - |I_{t-1} - I_{t+1}|) > \tau$$

Double difference

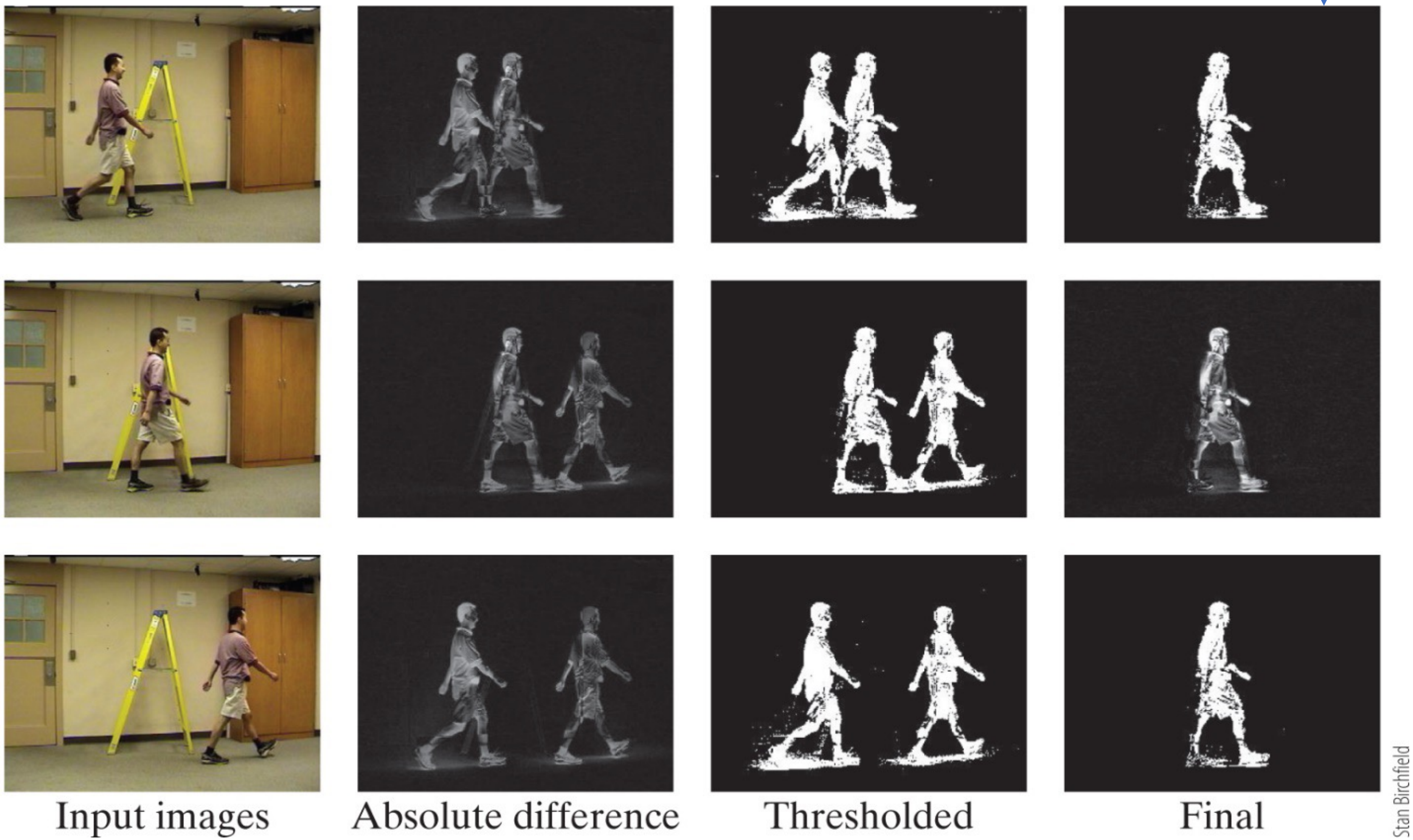


Figure 3.27 Detecting a moving object by frame differencing. LEFT COLUMN: Three image frames from a video sequence. SECOND COLUMN: The absolute difference between pairs of frames. THIRD COLUMN: Thresholded absolute difference. RIGHT COLUMN: Final result using double difference (top), triple difference (middle), and thresholded triple difference (bottom) methods.

Pairs of frames

Pairs of frames

Triplets of frames

Background subtraction

Background image: a reference image that does not contain any foreground objects. If available;

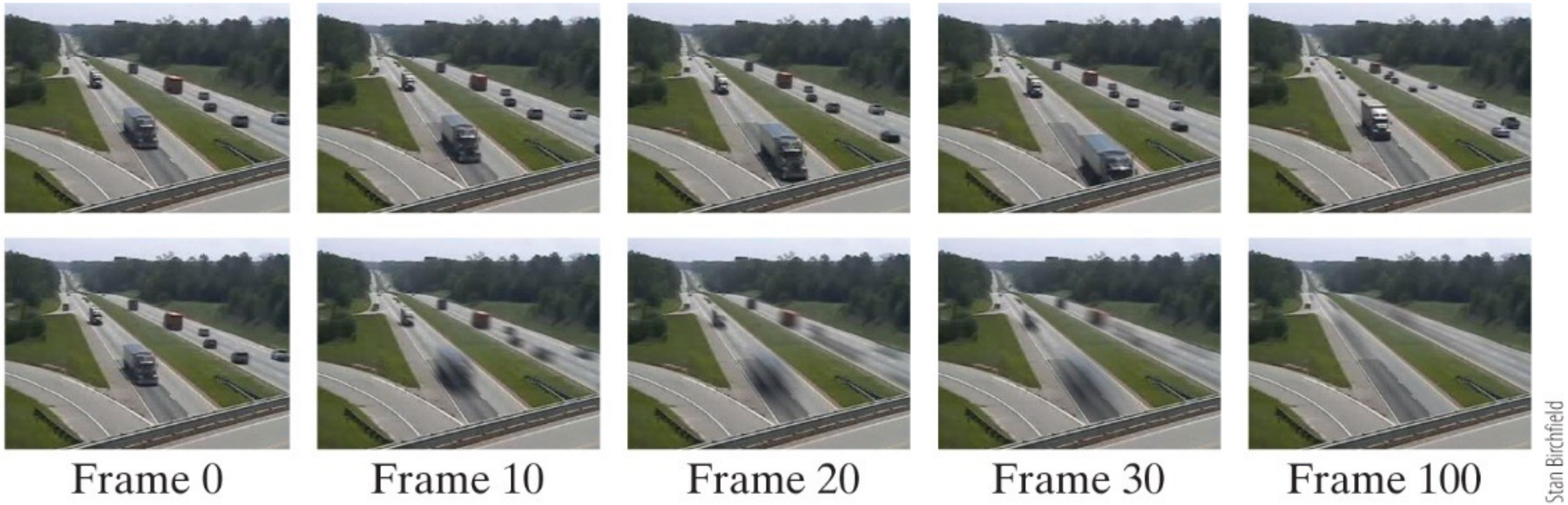
$$I'(x, y) = |I(x, y) - B(x, y)|$$

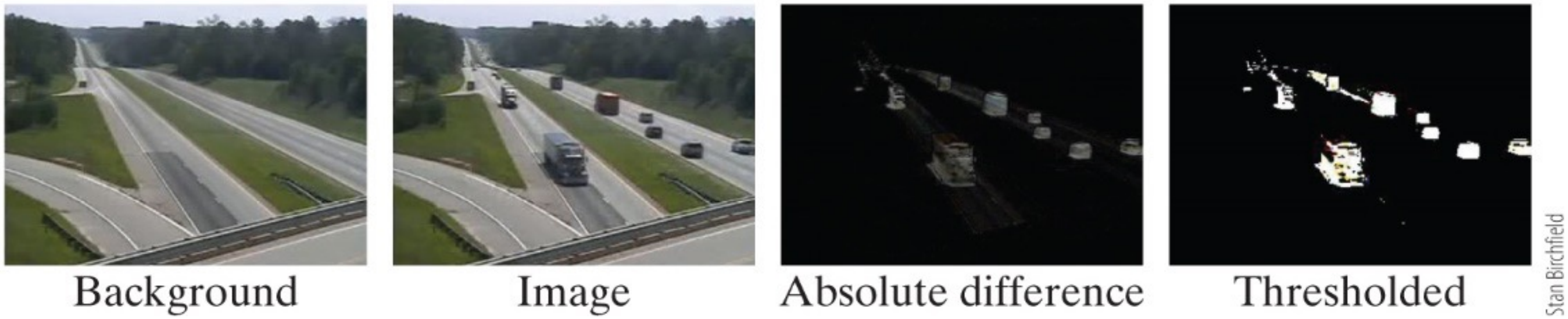
Can capture foreground objects even if they do not move for a while (not possible with frame difference)

Ex. From medical imaging: Digital subtraction angiography (DSA): a reference image is captured of a blood vessel before injecting it with dye to increase contrast.

Alternative; find **average image** over successive video frames

Figure 3.28 Top: Five images from a video sequence. Bottom: Each column shows the mean image obtained using all the images up to and including the one above it. As time progresses the moving objects disappear, leaving only the background.





(3.7) Compositing

Digital compositing: Used to blend live actions with computer graphics etc.

Example, dissolving: blend to images, moving from one to the other.

Figure 3.30 Two examples of dissolving one image into another. Source: Screenshots by WETA Digital Ltd. – © 2011 Paramount Pictures. 'The Adventures of Tintin'

Compositing with binary masks

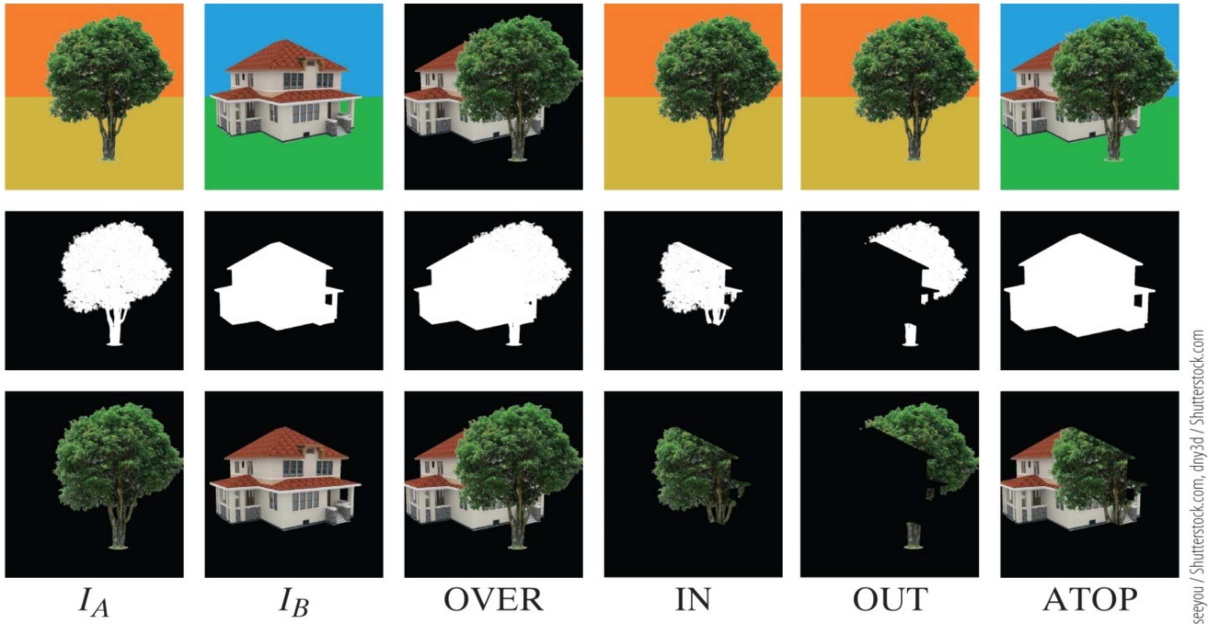


Figure 3.32 Common binary compositing operations applied to a pair of masked images. The top two rows show, from left to right: Original image I_A and mask M_A, original image I_B and mask M_B, and image I' and mask M' resulting from the four operations OVER, IN, OUT, and ATOP, respectively. The bottom row shows the result of ANDing each image with each mask.

More on Point transformations - Histogram, change detection etc. (3.3-3.7)

Three points from the topic:

1. What is a histogram? (Used in many fields)
    1. Count the number of pixels with specific values (bins of values)

2. how can it be useful studying/improving an image?
    1. Can see if the use of the possible grey levels is skewed. Can enhance by trying to flatten the histogram

3. How can we detect change between to images, for example same position different time points?
    1. Difference imaging (double) and background subtractions