Image Mosaicing

• **Introduction** [1], [2]
  - Definition
  - Motivation

• **Image Mosaicing Methods** [1], [2]
  - Registration and Warping
  - Image Compositing

• **Examples**

**References**

**Image Mosaicing Methods**

An image mosaicing process

- **Image Registration**: given a set of $N$ images $\{I_1, I_2, ..., I_N\}$ with a partial overlap between at least two images, compute an image-to-image transformation that will map each image $I_2, ..., I_N$ into the coordinate system of $I_1$.
- **Image Warping**: warp each image $I_2, ..., I_N$ using the computed transformation.
- **Image Interpolation**: resample the warped image.
- **Image Compositing**: blend images together to create a single image on the reference coordinate system.
Warping in the Discrete Domain

Forward and reverse mapping

In the forward mapping the source image is scanned pixel by pixel, and copies them to the appropriate location in the destination image.

The reverse mapping goes through the destination image, pixel by pixel, and samples the corresponding pixel from the source image. The main advantage of the reverse mapping is that every pixel in the destination image will have assigned an intensity value. In the forward mapping case, some of the pixels in the destination images may not be coloured, and would have to be interpolated.

1. Apply forward mapping at \([x, y]^T\), to obtain real valued point \([x', y']^T\).
2. Assign \([x, y]^T\) intensity to closest \([x', y']^T\)

1. For each destination pixel, at \([x', y']^T\) apply reverse mapping to obtain \([x, y]^T\) real-valued coordinates.
2. Interpolate intensities at \([x, y]^T\) from neighbourhood pixels and copy intensity to \([x', y']^T\)
Warping in the Discrete Domain

Expansion and contraction problems

When working with digital images, we deal with a discrete space and quantized intensities. Warping an image in the discrete space has as a consequence dilations and contractions of the rectangular pixels, originating, in general, quadrilaterals. This expansion/contractions demands the use of convenient methods for estimating pixel intensities on the image result. Two different problems may arise. In the case of expansion some pixels have no intensity assigned. In the case of contraction, several original pixel may converge to a single one. In both cases we need to estimate the new pixel intensities. These two problems are two typical instances of image resampling. In the first case, expansion, we have to use interpolation techniques to estimate the intermediate pixel intensities. The contraction may originate aliasing problems. To limit its effect we can use anti-aliasing filters.
Image Compositing

A particular case of image combination is the function *dissolve* characterised by

\[ I = \text{Dissolve}_a(I_1, I_2) = (1 - a)I_1 + a I_2 \]

where \( a \) is in the interval \([0, 1]\). We may notice that for \( a = 0 \), we have

\[ I = \text{Dissolve}_0(I_1, I_2) = I_1 \]

and for \( a = 1 \)

\[ I = \text{Dissolve}_1(I_1, I_2) = I_2 \]

For other values of \( a \), the image result is the weighted average of the two images, as illustrated in Figure below. The value of \( a \) is constant, being independent of the position of the pixel to be combined.

*Image dissolving: a) Image I₁; b) Dissolve_{0.25}(I₁, I₂); c) Dissolve_{0.25}(I₁, I₂); d) Dissolve_{0.75}(I₁, I₂); e) Image I₂.*
**Image Compositing**

Combining images by domain decomposing

In the case illustrated in the Figures below, the idea is to combine the images from a) and b), by decomposing the domain into two regions: region A, corresponding to the background (black region in c); region B, corresponding to the object on the foreground (the runner, on white region in c). The final image e) is given by the expression:

\[ I(x,y) = I_1(x,y) \quad \text{if} \quad (x,y) \in A \quad \text{and} \quad I(x,y) = I_2(x,y) \quad \text{if} \quad (x,y) \in B \]

The operations needed to obtain this result are straightforward, corresponding to a sequence of the type:

Step 1: \[ I_T = \text{Thresh}(I_1) \]
Step 2: \[ I_M = \text{Or}(I_1, I_2) \quad \text{(or Max}(I_1, I_2) \text{)} \]
Step 3: \[ I_F = \text{And}(I_M, I_1) \quad \text{(or Min}(I_M, I_1) \text{)} \]

---

a) Image \( I_1 \); \quad b) Image \( I_2 \); \quad c) \( I_T = \text{Thresh}(I_1) \); \quad d) \( I_M = \text{Or}(I_1, I_2) \); \quad e) \( I_F = \text{And}(I_M, I_1) \)


**Image Compositing**

**Blending Methods**

- *Image averaging*: average overlapping pixels.
- *Median*: compute the median of overlapping pixels.
- *Newest intensity*: take the most recent pixel intensity.
- *Bilinear blending*: weighted averaged, according to the distance to image center.
Examples (1)

(1) Examples by Nuno Ferreira at Faculty of Engineering of University of Porto
Examples

Frame 1

Frame 2

Perspective Transformation $\rightarrow M^{-1}$

Blending Function
Examples (illustration of perspective effects)

Frame 1

Frame 2