Autostiching on A9.com images,

Spruce street, Philadelphia
Image Warping

image filtering: change range of image
\[ g(x) = T(f(x)) \]

image warping: change domain of image
\[ g(x) = f(T(x)) \]

Parametric (global) warping

Examples of parametric warps:
- translation
- rotation
- aspect
- affine
- perspective
- cylindrical
Parametric (global) warping

Transformation $T$ is a coordinate-changing machine:

$$p' = T(p)$$

What does it mean that $T$ is global?

- Is the same for any point $p$
- Can be described by just a few numbers (parameters)

Let's represent $T$ as a matrix:

$$p' = M \cdot p$$

Scaling

Scaling a coordinate means multiplying each of its components by a scalar.

Uniform scaling means this scalar is the same for all components:

Scaling operation:

$$x' = ax$$
$$y' = by$$

Or, in matrix form:

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} a & 0 \\ 0 & b \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

Scaling

Non-uniform scaling: different scalars per component:

Scaling operation:

$$x' = ax$$
$$y' = by$$

Or, in matrix form:

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} a & 0 \\ 0 & b \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

What's inverse of $S$?
2-D Rotation

This is easy to capture in matrix form:

\[
\begin{bmatrix}
  x' \\
  y'
\end{bmatrix} =
\begin{bmatrix}
  \cos(\theta) & -\sin(\theta) \\
  \sin(\theta) & \cos(\theta)
\end{bmatrix}
\begin{bmatrix}
  x \\
  y
\end{bmatrix}
\]

Even though \(\sin(\theta)\) and \(\cos(\theta)\) are nonlinear functions of \(\theta\),

- \(x'\) is a linear combination of \(x\) and \(y\)
- \(y'\) is a linear combination of \(x\) and \(y\)

What is the inverse transformation?
- Rotation by \(-\theta\)
- For rotation matrices, \(\det(R) = 1\) so \(R^{-1} = R^T\)

2x2 Matrices

What types of transformations can be represented with a 2x2 matrix?

2D Identity?

\[
\begin{bmatrix}
  x' \\
  y'
\end{bmatrix} =
\begin{bmatrix}
  1 & 0 \\
  0 & 1
\end{bmatrix}
\begin{bmatrix}
  x \\
  y
\end{bmatrix}
\]

2D Scale around (0,0)?

\[
\begin{bmatrix}
  x' \\
  y'
\end{bmatrix} =
\begin{bmatrix}
  s_x & 0 \\
  0 & s_y
\end{bmatrix}
\begin{bmatrix}
  x \\
  y
\end{bmatrix}
\]
**2x2 Matrices**

What types of transformations can be represented with a 2x2 matrix?

2D Rotate around (0,0)?

\[
\begin{align*}
x' &= \cos \Theta \cdot x - \sin \Theta \cdot y \\
y' &= \sin \Theta \cdot x + \cos \Theta \cdot y
\end{align*}
\]

\[
\begin{bmatrix}
x' \\
y'
\end{bmatrix} = 
\begin{bmatrix}
\cos \Theta & -\sin \Theta \\
\sin \Theta & \cos \Theta
\end{bmatrix} 
\begin{bmatrix}
x \\
y
\end{bmatrix}
\]

2D Shear?

\[
\begin{align*}
x' &= x + \text{sh}_x \cdot y \\
y' &= \text{sh}_y \cdot x + y
\end{align*}
\]

\[
\begin{bmatrix}
x' \\
y'
\end{bmatrix} = 
\begin{bmatrix}
1 & \text{sh}_x \\
\text{sh}_y & 1
\end{bmatrix} 
\begin{bmatrix}
x \\
y
\end{bmatrix}
\]

2D Mirror about Y axis?

\[
\begin{align*}
x' &= -x \\
y' &= y
\end{align*}
\]

\[
\begin{bmatrix}
x' \\
y'
\end{bmatrix} = 
\begin{bmatrix}
-1 & 0 \\
0 & 1
\end{bmatrix} 
\begin{bmatrix}
x \\
y
\end{bmatrix}
\]

2D Mirror over (0,0)?

\[
\begin{align*}
x' &= -x \\
y' &= -y
\end{align*}
\]

\[
\begin{bmatrix}
x' \\
y'
\end{bmatrix} = 
\begin{bmatrix}
-1 & 0 \\
0 & -1
\end{bmatrix} 
\begin{bmatrix}
x \\
y
\end{bmatrix}
\]

**2x2 Matrices**

What types of transformations can be represented with a 2x2 matrix?

**All 2D Linear Transformations**

Linear transformations are combinations of ...
- Scale,
- Rotation,
- Shear, and
- Mirror

Properties of linear transformations:
- Origin maps to origin
- Lines map to lines
- Parallel lines remain parallel
- Ratios are preserved
- Closed under composition

\[
\begin{bmatrix}
x' \\
y'
\end{bmatrix} = 
\begin{bmatrix}
a & b \\
c & d
\end{bmatrix} 
\begin{bmatrix}
x \\
y
\end{bmatrix}
\]

Only linear 2D transformations can be represented with a 2x2 matrix
Linear Transformations as Change of Basis

Any linear transformation is a basis!!!

- What's the inverse transform?
- How can we change from any basis to any basis?
- What if the basis are orthogonal?

\[ p = 4i + 3j = (4, 3) \]
\[ i = (1, 0) \]
\[ j = (0, 1) \]
\[ p = 4i + 3j = (4, 3) \]
\[ i = (1, 0) \]
\[ j = (0, 1) \]
\[ p' = 4u + 3v \]
\[ p_x' = 4u_x + 3v_x \]
\[ p_y' = 4u_y + 3v_y \]

Homogeneous Coordinates

Homogeneous coordinates
- represent coordinates in 2 dimensions with a 3-vector

\[ \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \rightarrow \text{homogeneous coords} \rightarrow \begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} \]

Q: How can we represent translation as a 3x3 matrix?

\[ x' = x + t_x \]
\[ y' = y + t_y \]

A: Using the rightmost column:

\[
\begin{bmatrix}
1 & 0 & t_x \\
0 & 1 & t_y \\
0 & 0 & 1
\end{bmatrix}
\]

Homogeneous Coordinates
**Translation**

Example of translation

Homogeneous Coordinates

Add a 3rd coordinate to every 2D point
- \((x, y, w)\) represents a point at location \((x/w, y/w)\)
- \((x, y, 0)\) represents a point at infinity
- \((0, 0, 0)\) is not allowed

Convenient coordinate system to represent many useful transformations

**Homogeneous Coordinates**

**Basic 2D Transformations**

Basic 2D transformations as 3x3 matrices

**Affine Transformations**

Affine transformations are combinations of ...
- Linear transformations, and
- Translations

Properties of affine transformations:
- Origin does not necessarily map to origin
- Lines map to lines
- Parallel lines remain parallel
- Ratios are preserved
- Closed under composition
- Models change of basis
Projective Transformations

Projective transformations …

• Affine transformations, and
• Projective warps

Properties of projective transformations:

• Origin does not necessarily map to origin
• Lines map to lines
• Parallel lines do not necessarily remain parallel
• Ratios are not preserved
• Closed under composition
• Models change of basis

Matrix Composition

Transformations can be combined by matrix multiplication

\[
\begin{bmatrix}
\begin{vmatrix}
1 & 0 & tx \\
0 & 1 & ty \\
0 & 0 & 1 \\
\end{vmatrix}
& \begin{vmatrix}
\cos \Theta & -\sin \Theta & 0 \\
\sin \Theta & \cos \Theta & 0 \\
0 & 0 & 1 \\
\end{vmatrix}
& \begin{vmatrix}
sx & 0 & 0 \\
0 & sy & 0 \\
0 & 0 & 1 \\
\end{vmatrix}
\end{bmatrix}
\begin{bmatrix}
x \\
y \\
w \\
\end{bmatrix} = \begin{bmatrix}
x' \\
y' \\
w' \\
\end{bmatrix}
\]

\[
p' = T(t_x,t_y) \quad R(\Theta) \quad S(s_x,s_y) \quad p
\]

2D image transformations

<table>
<thead>
<tr>
<th>Name</th>
<th>Matrix</th>
<th># D.O.F</th>
<th>Preserves:</th>
<th>Icon</th>
</tr>
</thead>
<tbody>
<tr>
<td>translation</td>
<td>[I \ t]_{2x3}</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>rigid (Euclidean)</td>
<td>[R \ t]_{2x3}</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>similarity</td>
<td>[sR \ t]_{2x3}</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>affine</td>
<td>[A]_{3x3}</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>projective</td>
<td>[P]_{3x3}</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

These transformations are a nested set of groups

• Closed under composition and inverse is a member

Image warping

Given a coordinate transform \((x',y') = h(x,y)\) and a source image \(f(x,y)\), how do we compute a transformed image \(g(x',y') = f(T(x,y))\)?
**Forward warping**

Send each pixel \( f(x,y) \) to its corresponding location \((x',y') = T(x,y)\) in the second image.

Q: what if pixel lands “between” two pixels?

A: distribute color among neighboring pixels \((x',y')\) – Known as “splatting”

---

**Inverse warping**

Get each pixel \( g(x',y') \) from its corresponding location \((x,y) = T^{-1}(x',y')\) in the first image.

Q: what if pixel comes from “between” two pixels?

A: Interpolate color value from neighbors – nearest neighbor, bilinear, Gaussian, bicubic
Bilinear interpolation

Sampling at $f(x,y)$:

$$f(x, y) = (1 - a)(1 - b) f[i, j] + a(1 - b) f[i + 1, j] + ab f[i + 1, j + 1] + (1 - a)b f[i, j + 1]$$

Forward vs. inverse warping

Q: which is better?

A: usually inverse—eliminates holes
   • however, it requires an invertible warp function—not always possible...