Alineación

Lección 05.1

Dr. Pablo Alvarado Moya

MP6127 Visión por Computadora
Programa de Maestría en Electrónica
Énfasis en Procesamiento Digital de Señales
Escuela de Ingeniería Electrónica
Tecnológico de Costa Rica

I Cuatrimestre 2013

Contenido

- **Alineación** es el proceso de hacer calzar varias imágenes
- Las imagenes se distorsionan geométricamente para calzar en una referencia común.
- Tareas como:
  - Registro de imágenes en mosaicing (creación de panoramas)
  - Calibración

Ejemplo (1)

Dadas dos imágenes:

Imagen 1

Imagen 2

Puntos de interés + descriptores → correspondencias

Canal verde: Imagen 1 — Canal magenta: Imagen 2

Correspondencias falsas

¿Cómo verificar cuáles correspondencias son consistentes?

Transformaciones lineales

- Caso frecuente: suponer transformación lineal
- Repasar vídeo 4.1 de Procesamiento Digital de Imágenes

Transformaciones lineales elementales

Todas las transformaciones lineales tienen la forma

$$\underline{\hat{\mathbf{x}}}' = \mathbf{M}\underline{\mathbf{x}} = \underline{\mathbf{f}}(\underline{\mathbf{x}}; \underline{\mathbf{p}})$$

con

- el punto a transformar $\underline{\mathbf{x}} = [x, y, 1]^T$ en coordenadas homogéneas,
- el punto transformado $\underline{\hat{\mathbf{x}}}' = [x', y', 1]^T$ en coordenadas homogéneas, donde "ˆ" denota que la transformación es pura (sin ruido)
- el punto medido $\underline{\mathbf{x}}' = [\tilde{x}', \tilde{y}', 1]^T$ corresponde al punto transformado, más el ruido de medición y de imprecisión de la transformación. También se denota con $\underline{\tilde{\mathbf{x}}}'$
- $\underline{\mathbf{p}}$ los parámetros de la transformación.

Transformaciones lineales elementales

Tipo	**Matriz M**	**Parámetros p**	**Función f(x; p)**
Traslación	$\begin{bmatrix} 1 & 0 & t_x \\ 0 & 1 & t_y \end{bmatrix}$	$[t_x, t_y]$	$\begin{bmatrix} x + t_x \\ y + t_y \end{bmatrix}$
Euclideana	$\begin{bmatrix} c_\theta & -s_\theta & t_x \\ c_\theta & s_\theta & t_y \end{bmatrix}$	$[t_x, t_y, \theta]^T$	$\begin{bmatrix} xc_\theta - ys_\theta + t_x \\ xs_\theta + yc_\theta + t_y \end{bmatrix}$
Similitud	$\begin{bmatrix} 1+a & -b & t_x \\ b & 1+a & t_y \end{bmatrix}$	$[t_x, t_y, a, b]^T$	$\begin{bmatrix} (1+a)x - by + t_x \\ bx + (1+a)y + t_y \end{bmatrix}$
Afín	$\begin{bmatrix} 1+a_{00} & a_{01} & t_x \\ a_{10} & 1+a_{11} & t_y \end{bmatrix}$	$[t_x, t_y, a_{ij}]^T$	$\begin{bmatrix} (1+a_{00})x - a_{11}y + t_x \\ a_{10}x + (1+a_{11})y + t_y \end{bmatrix}$
Proyectiva	$\begin{bmatrix} 1+h_{00} & h_{01} & h_{02} \\ h_{10} & 1+h_{11} & h_{12} \\ h_{20} & h_{21} & 1 \end{bmatrix}$	$[h_{ij}]^T$	$\frac{1}{D}\begin{bmatrix} (1+h_{00})x - h_{01}y + h_{02} \\ h_{10}x + (1+h_{11})y + h_{12} \end{bmatrix}$ $D = h_{20}x + h_{21}y + 1$

Parametrización hace que si $\underline{\mathbf{p}} = \underline{\mathbf{0}} \Rightarrow \underline{\hat{\mathbf{x}}}' = \underline{\mathbf{x}}$

Jacobiano

- El **jacobiano** de una función vectorial paramétrica $\underline{\mathbf{f}} : \mathbb{R}^d \times \mathbb{R}^n \rightarrow \mathbb{R}^m$ es la matriz de todas las derivadas de primer orden con respecto a los parámetros:

$$\mathbf{J}\underline{\mathbf{f}}(\underline{\mathbf{x}}) = \mathbf{J}(\underline{\mathbf{x}}) = \begin{bmatrix} \frac{\partial f_1(\underline{\mathbf{x}};\underline{\mathbf{p}})}{\partial p_1} & \frac{\partial f_1(\underline{\mathbf{x}};\underline{\mathbf{p}})}{\partial p_2} & \cdots & \frac{\partial f_1(\underline{\mathbf{x}};\underline{\mathbf{p}})}{\partial p_n} \\ \frac{\partial f_2(\underline{\mathbf{x}};\underline{\mathbf{p}})}{\partial p_1} & \frac{\partial f_2(\underline{\mathbf{x}};\underline{\mathbf{p}})}{\partial p_2} & \cdots & \frac{\partial f_2(\underline{\mathbf{x}};\underline{\mathbf{p}})}{\partial p_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial f_m(\underline{\mathbf{x}};\underline{\mathbf{p}})}{\partial p_1} & \frac{\partial f_m(\underline{\mathbf{x}};\underline{\mathbf{p}})}{\partial p_2} & \cdots & \frac{\partial f_m(\underline{\mathbf{x}};\underline{\mathbf{p}})}{\partial p_n} \end{bmatrix} = \begin{bmatrix} \nabla^T f_1 \\ \nabla^T f_2 \\ \vdots \\ \nabla^T f_m \end{bmatrix}$$

- Se denota también como $\partial\underline{\mathbf{f}}(\underline{\mathbf{x}};\underline{\mathbf{p}})/\partial\underline{\mathbf{p}}$

Tipo	**Matriz M**	**Parámetros $\underline{\mathbf{p}}$**	**Función $\underline{\mathbf{f}}(\underline{\mathbf{x}};\underline{\mathbf{p}})$**
Similitud	$\begin{bmatrix} 1+a & -b & t_x \\ b & 1+a & t_y \end{bmatrix}$	$[t_x, t_y, a, b]^T$	$\begin{bmatrix} (1+a)x-by+t_x \\ bx+(1+a)y+t_y \end{bmatrix}$

Con lo anterior:

$$\underline{\mathbf{f}}(x, y; t_x, t_y, a, b) = \begin{bmatrix} f_1(x, y; t_x, t_y, a, b) \\ f_2(x, y; t_x, t_y, a, b) \end{bmatrix} = \begin{bmatrix} (1+a)x - by + t_x \\ bx + (1+a)y + t_y \end{bmatrix}$$

Por lo tanto, la matriz jacobiana es

$$\mathbf{J}(\underline{\mathbf{x}}) = \begin{bmatrix} \frac{\partial f_1}{\partial t_x} & \frac{\partial f_1}{\partial t_y} & \frac{\partial f_1}{\partial a} & \frac{\partial f_1}{\partial b} \\ \frac{\partial f_2}{\partial t_x} & \frac{\partial f_2}{\partial t_y} & \frac{\partial f_2}{\partial a} & \frac{\partial f_2}{\partial b} \end{bmatrix} = \begin{bmatrix} 1 & 0 & x & -y \\ 0 & 1 & y & x \end{bmatrix}$$

Jacobiano de transformaciones lineales elementales

Tipo	**Matriz M**	**Parámetros p**	**Jacobiano J**
Traslación	$\begin{bmatrix} 1 & 0 & t_x \\ 0 & 1 & t_y \end{bmatrix}$	$[t_x, t_y]$	$\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$
Euclideana	$\begin{bmatrix} c_\theta & -s_\theta & t_x \\ c_\theta & s_\theta & t_y \end{bmatrix}$	$[t_x, t_y, \theta]^T$	$\begin{bmatrix} 1 & 0 & -s_\theta - c_\theta y \\ 0 & 1 & c_\theta x - s_\theta y \end{bmatrix}$
Similitud	$\begin{bmatrix} 1+a & -b & t_x \\ b & 1+a & t_y \end{bmatrix}$	$[t_x, t_y, a, b]^T$	$\begin{bmatrix} 1 & 0 & x & -y \\ 0 & 1 & y & x \end{bmatrix}$
Afín	$\begin{bmatrix} 1+a_{00} & a_{01} & t_x \\ a_{10} & 1+a_{11} & t_y \end{bmatrix}$	$[t_x, t_y, a_{ij}]^T$	$\begin{bmatrix} 1 & 0 & x & y & 0 & 0 \\ 0 & 1 & 0 & 0 & x & y \end{bmatrix}$
Proyectiva	$\begin{bmatrix} 1+h_{00} & h_{01} & h_{02} \\ h_{10} & 1+h_{11} & h_{12} \\ h_{20} & h_{21} & 1 \end{bmatrix}$	$[h_{ij}]^T$	¿$\mathbf{J}$?

- Serie de Taylor de función escalar univariada:

$$
\begin{aligned}
f(x) &= \sum_{i=0}^{\infty} \frac{f^{(n)}(x_0)}{n!}(x - x_0)^n \\
&= f(x_0) + (x - x_0)f'(x_0) + \frac{(x - x_0)^2}{2!}f''(x_0) + \cdots
\end{aligned}
$$

- Serie de Taylor de función vectorial univariada:

$$\underline{\mathbf{f}}(x) = \begin{bmatrix} f_1(x) \\ \vdots \\ f_d(x) \end{bmatrix} = \begin{bmatrix} \sum_{i=0}^{\infty} \frac{f_1^{(n)}(x_0)}{n!}(x-x_0)^n \\ \vdots \\ \sum_{i=0}^{\infty} \frac{f_d^{(n)}(x_0)}{n!}(x-x_0)^n \end{bmatrix} = \sum_{i=0}^{\infty} \frac{\underline{\mathbf{f}}^{(n)}(x_0)}{n!}(x-x_0)^n$$

$$= \underline{\mathbf{f}}(x_0) + (x-x_0)\underline{\mathbf{f}}'(x_0) + \frac{(x-x_0)^2}{2!}\underline{\mathbf{f}}''(x_0) + \cdots$$

$$= \begin{bmatrix} f_1(x) \\ \vdots \\ f_d(x) \end{bmatrix} + (x-x_0) \begin{bmatrix} f_1'(x) \\ \vdots \\ f_d'(x) \end{bmatrix} + \frac{(x-x_0)^2}{2!} \begin{bmatrix} f_1''(x) \\ \vdots \\ f_d''(x) \end{bmatrix} + \cdots$$

- Serie de Taylor de función escalar multivariada

$$f(\underline{\mathbf{x}}) = f(\underline{\mathbf{x}}_0) + \nabla f(\underline{\mathbf{x}}_0)^T(\underline{\mathbf{x}} - \underline{\mathbf{x}}_0) + \frac{1}{2!}(\underline{\mathbf{x}} - \underline{\mathbf{x}}_0)^T \mathbf{H} f(\underline{\mathbf{x}}_0)(\underline{\mathbf{x}} - \underline{\mathbf{x}}_0) + \cdots$$

- Serie de Taylor de función vectorial multivariada

$$\underline{\mathbf{f}}(\underline{\mathbf{x}}) = \underline{\mathbf{f}}(\underline{\mathbf{x}}_0) + \mathbf{J}(\underline{\mathbf{x}}_0)(\underline{\mathbf{x}} - \underline{\mathbf{x}}_0) + \cdots$$

Alineación usando mínimos cuadrados

- Asúmase que se tiene un conjunto de correspondencias:

$$\{(\underline{\mathbf{x}}_i, \underline{\mathbf{x}}'_i) \mid i = 1, 2 \ldots m\}$$

- La transformación $\underline{\mathbf{f}}$ predice dónde quedará $\underline{\mathbf{x}}_i$

$$\hat{\underline{\mathbf{x}}}'_i = \underline{\mathbf{f}}(\underline{\mathbf{x}}_i; \underline{\mathbf{p}})$$

- Problema de optimización: ¿cómo encontrar *mejor* $\underline{\mathbf{p}}$?
- Residuo: vector predicho menos vector medido:

$$\underline{\mathbf{r}}_i(\underline{\mathbf{p}}) = \underline{\mathbf{f}}(\underline{\mathbf{x}}_i; \underline{\mathbf{p}}) - \underline{\mathbf{x}}'_i = \hat{\underline{\mathbf{x}}}'_i - \underline{\mathbf{x}}'_i$$

- Error se puede definir como:

$$E_{LS}(\underline{\mathbf{p}}) = \sum_i^m \|\underline{\mathbf{r}}_i(\underline{\mathbf{p}})\|^2 \qquad m \geq \dim(\underline{\mathbf{p}})$$

Relaciones lineales

- Modelos de traslación, de similitud y afines (entre otros) tienen relación lineal entre parámetros y el vector de movimiento $\underline{\hat{\mathbf{x}}}' - \underline{\mathbf{x}}$
- Para estos modelos se cumple

$$\underline{\hat{\mathbf{x}}}' - \underline{\mathbf{x}} = \mathbf{J}(\underline{\mathbf{x}})\underline{\mathbf{p}}$$

Por ejemplo: transformación de similitud

$$\underline{\hat{\mathbf{x}}}' - \underline{\mathbf{x}} = \begin{bmatrix} (1+a-1)x - by + t_x \\ bx + (1+a-1)y + t_y \end{bmatrix} = \begin{bmatrix} ax - by + t_x \\ bx - ax + t_y \end{bmatrix}$$

$$= \begin{bmatrix} 1 & 0 & x & -y \\ 0 & 1 & y & x \end{bmatrix} \begin{bmatrix} t_x \\ t_y \\ a \\ b \end{bmatrix}$$

- De lo anterior

$$\begin{aligned}
\underline{\hat{\mathbf{x}}}'_i - \underline{\mathbf{x}}_i &= \mathbf{J}(\underline{\mathbf{x}}_i)\underline{\mathbf{p}} \qquad \Rightarrow \qquad \underline{\hat{\mathbf{x}}}'_i = \mathbf{J}(\underline{\mathbf{x}}_i)\underline{\mathbf{p}} + \underline{\mathbf{x}}_i \\
\underline{\mathbf{r}}_i(\underline{\mathbf{p}}) &= \underline{\hat{\mathbf{x}}}'_i - \underline{\mathbf{x}}'_i \\
&= \mathbf{J}(\underline{\mathbf{x}}_i)\underline{\mathbf{p}} - (\underline{\mathbf{x}}'_i - \underline{\mathbf{x}}_i) \\
&= \mathbf{J}(\underline{\mathbf{x}}_i)\underline{\mathbf{p}} - \Delta\underline{\mathbf{x}}_i
\end{aligned}$$

- El problema a resolver es encontrar parámetros $\underline{\mathbf{p}}$ que minimizan el error:

$$E_{LLS} = \sum_i \|\underline{\mathbf{r}}_i\|^2 = \sum_i \left\|\mathbf{J}(\underline{\mathbf{x}}_i)\underline{\mathbf{p}} - \Delta\underline{\mathbf{x}}_i\right\|^2$$

- Puesto que $\|\underline{\mathbf{r}}_i\|^2 = \underline{\mathbf{r}}_i^T \underline{\mathbf{r}}_i$ entonces

$$\begin{aligned}
E &= \sum_i \left(\mathbf{J}(\underline{\mathbf{x}}_i)\underline{\mathbf{p}} - \Delta\underline{\mathbf{x}}_i\right)^T \left(\mathbf{J}(\underline{\mathbf{x}}_i)\underline{\mathbf{p}} - \Delta\underline{\mathbf{x}}_i\right) \\
&= \sum_i \left(\underline{\mathbf{p}}^T \mathbf{J}^T(\underline{\mathbf{x}}_i) - \Delta\underline{\mathbf{x}}_i^T\right) \left(\mathbf{J}(\underline{\mathbf{x}}_i)\underline{\mathbf{p}} - \Delta\underline{\mathbf{x}}_i\right) \\
&= \underline{\mathbf{p}}^T \left[\sum_i \mathbf{J}^T(\underline{\mathbf{x}}_i)\mathbf{J}(\underline{\mathbf{x}}_i)\right] \underline{\mathbf{p}} - 2\underline{\mathbf{p}}^T \left[\sum_i \mathbf{J}^T(\underline{\mathbf{x}}_i)\Delta\underline{\mathbf{x}}_i\right] + \sum_i \|\Delta\underline{\mathbf{x}}_i\|^2 \\
&= \underline{\mathbf{p}}^T \mathbf{A}\underline{\mathbf{p}} - 2\underline{\mathbf{p}}^T \underline{\mathbf{b}} + c
\end{aligned}$$

- El mínimo de $E = \underline{\mathbf{p}}^T \mathbf{A}\underline{\mathbf{p}} - 2\underline{\mathbf{p}}^T \underline{\mathbf{b}} + c$ resuelve el sistema simétrico positivo definido de ecuaciones normales:

$$\mathbf{A}\underline{\mathbf{p}} = \underline{\mathbf{b}}$$

- matriz hessiana $\mathbf{A} = \sum_i \mathbf{J}^T(\underline{\mathbf{x}}_i)\mathbf{J}(\underline{\mathbf{x}}_i)$
  (**CUIDADO**: no confundir con matriz hessiana $\mathbf{H}$ de segundas derivadas)
- Cuando la matriz $\mathbf{A}$ es mal condicionada, mejor usar solución de sistema lineal sobre-especificado (muchas más correspondencias que parámetros) empleando SVD.
- $\underline{\mathbf{b}} = \sum_i \mathbf{J}^T(\underline{\mathbf{x}}_i)\Delta\underline{\mathbf{x}}_i$

- No siempre es válida suposición de linealidad

$$\underline{\hat{\mathbf{x}}}' - \underline{\mathbf{x}} = \mathbf{J}(\underline{\mathbf{x}})\underline{\mathbf{p}}$$

- Es necesario hacer regresión no lineal (problema no lineal de mínimos cuadrados)
- Solución es iterativa:

$$\underline{\mathbf{p}}^{(k+1)} = \underline{\mathbf{p}}^{(k)} + \Delta\underline{\mathbf{p}}^{(k)}$$

- Se busca minimizar el error

$$E_{NLS}(\Delta\underline{\mathbf{p}}^{(k)}) = \sum_i \left\| \underline{\mathbf{r}}_i^{k+1} \right\|^2 = \sum_i \left\| \underline{\mathbf{f}}(\underline{\mathbf{x}}_i; \underline{\mathbf{p}} + \Delta\underline{\mathbf{p}}) - \underline{\mathbf{x}}'_i \right\|^2$$

- Con la aproximación de primer orden con Serie de Taylor

$$\underline{\mathbf{f}}(\underline{\mathbf{x}};\underline{\mathbf{p}}^{(k)}+\Delta\underline{\mathbf{p}}) = \underbrace{\underline{\mathbf{f}}(\underline{\mathbf{x}};\underline{\mathbf{p}}^{(k+1)})}_{\hat{\underline{\mathbf{x}}}'^{(k+1)}} \approx \underbrace{\underline{\mathbf{f}}(\underline{\mathbf{x}};\underline{\mathbf{p}}^{(k)})}_{\hat{\underline{\mathbf{x}}}'^{(k)}} + \mathbf{J}(\underline{\mathbf{x}};\underline{\mathbf{p}}^{(k)})\Delta\underline{\mathbf{p}}$$

por tanto, para el residuo se cumple

$$\begin{aligned}
\underline{\mathbf{r}}_i^{(k+1)} &= \underline{\mathbf{f}}(\underline{\mathbf{x}}_i;\underline{\mathbf{p}}^{(k)}+\Delta\underline{\mathbf{p}}) - \underline{\mathbf{x}}'_i \\
&\approx \hat{\underline{\mathbf{x}}}'^{(k)} + \mathbf{J}(\underline{\mathbf{x}}_i;\underline{\mathbf{p}}^{(k)})\Delta\underline{\mathbf{p}} - \underline{\mathbf{x}}'_i \\
&\approx \mathbf{J}(\underline{\mathbf{x}}_i;\underline{\mathbf{p}}^{(k)})\Delta\underline{\mathbf{p}} + \underbrace{\hat{\underline{\mathbf{x}}}_i'^{(k)} - \underline{\mathbf{x}}'_i}_{\underline{\mathbf{r}}_i^{(k)}} \\
&\approx \mathbf{J}(\underline{\mathbf{x}}_i;\underline{\mathbf{p}}^{(k)})\Delta\underline{\mathbf{p}} + \underline{\mathbf{r}}_i^{(k)}
\end{aligned}$$

- Considerando lo anterior:

$$
\begin{aligned}
&E_{NLS}(\Delta\underline{\mathbf{p}}^{(k)}) \\
&= \sum_i \left\| \underline{\mathbf{r}}_i^{(k+1)} \right\|^2 = \sum_i \left\| \underline{\mathbf{f}}(\underline{\mathbf{x}}_i; \underline{\mathbf{p}} + \Delta\underline{\mathbf{p}}) - \underline{\mathbf{x}}'_i \right\|^2 \\
&\approx \sum_i \left\| \mathbf{J}(\underline{\mathbf{x}}_i; \underline{\mathbf{p}}^{(k)})\Delta\underline{\mathbf{p}} + \underline{\mathbf{r}}_i^{(k)} \right\|^2 \\
&= \Delta\underline{\mathbf{p}}^T \left[\sum_i \mathbf{J}^T\mathbf{J} \right] \Delta\underline{\mathbf{p}} + 2\Delta\underline{\mathbf{p}}^T \left[\sum_i \mathbf{J}^T \underline{\mathbf{r}}_i^{(k)} \right] + \sum_i \|\underline{\mathbf{r}}_i^{(k)}\|^2 \\
&= \Delta\underline{\mathbf{p}}^T \mathbf{A} \Delta\underline{\mathbf{p}} + 2\Delta\underline{\mathbf{p}}^T \underline{\mathbf{b}} + c
\end{aligned}
$$

- Para encontrar el mínimo, se resuelve

$$
\begin{aligned}
2\mathbf{A}\Delta\underline{\mathbf{p}} + 2\underline{\mathbf{b}} &= 0 \\
\mathbf{A}\Delta\underline{\mathbf{p}} &= -\underline{\mathbf{b}}
\end{aligned}
$$

- Por la aproximación de primer orden, debe asegurarse que el tamaño de paso sea lo suficientemente pequeño, para mantener validez
- El algoritmo de Levenberg-Marquardt usa el error

$$\tilde{E} = \sum_i \left(\frac{1}{2} \left\| \mathbf{J}\Delta\underline{\mathbf{p}} + \underline{\mathbf{r}}_i^{(k)} \right\|^2 + \lambda \|\Delta\underline{\mathbf{p}}\| \right)$$

  con el factor de Marquardt λ que controla el tamaño de paso
- Si aumenta λ, la minimización de $\tilde{E}$ obliga a un paso pequeño.
- En la práctica se implementan controles para λ:
  - si el error se estanca: $\lambda \leftarrow \lambda/10$.
  - si error aumenta: $\lambda \leftarrow \lambda \times 10$.
  - siempre se mantiene el mejor vector $\underline{\mathbf{p}}$
- Solución resulta en

$$(\mathbf{A} + \lambda\,\mathrm{diag}(\mathbf{A}))\,\Delta\underline{\mathbf{p}} = -\underline{\mathbf{b}}$$

Inicialización

- Inicializacion es relevante para asegurar convergencia a mínimo global
- Problemas no lineales pueden usar inicialización con problema lineal
- Ejemplo: transformación euclideana (t_x, t_y, θ) se inicializa con transformación de similitud (t_x, t_y, a, b) y luego $\theta = \tan^{-1}\left(\frac{b}{1+a}\right)$

Transformación proyectiva (homografía)

Tipo	**Matriz M**	**Parámetros $\underline{\mathbf{p}}$**	**Función $\underline{\mathbf{f}}(\underline{\mathbf{x}};\underline{\mathbf{p}})$**
Proyectiva	$\begin{bmatrix} 1+h_{00} & h_{01} & h_{02} \\ h_{10} & 1+h_{11} & h_{12} \\ h_{20} & h_{21} & 1 \end{bmatrix}$	$[h_{ij}]^T$	$\frac{1}{D}\begin{bmatrix}(1+h_{00})x-h_{01}y+h_{02} \\ h_{10}x+(1+h_{11})y+h_{12}\end{bmatrix}$ $D = h_{20}x + h_{21}y + 1$

Jacobiano:

$$\mathbf{J} = \frac{1}{D}\begin{bmatrix} x & y & 1 & 0 & 0 & 0 & -x'x & -x'y \\ 0 & 0 & 0 & x & y & 1 & -y'x & -y'y \end{bmatrix}$$

- Métodos anteriores no permiten valores atípicos (*outliers*)
- RANSAC busca detectar y eliminar estos valores antes de calcular parámetros
- Primero se selecciona aleatoriamente un subconjunto de k correspondencias, y se estima $\underline{\mathbf{p}}$
- Se calculan los residuos

$$\underline{\mathbf{r}}_i = \underline{\mathbf{f}}(\underline{\mathbf{x}}_i; \underline{\mathbf{p}}) - \underline{\mathbf{x}}'_i$$

- Cuenta cuántos valores típicos (*inliers*) cumplen $\|\underline{\mathbf{r}}_i\| \leq \epsilon$
- Umbral ϵ depende de aplicación pero usualmente se elige entre 1–3 píxeles.
- El proceso se repite S veces, y se elijen los parámetros que producen el mayor número de valores típicos.

Otros métodos robustos

- El método de mínima mediana de cuadrados es similar al RANSAC, pero calcula la mediana de todos los residuos
- Este método elije los parámetros que producen la menor mediana
- El método Preemptive RANSAC realiza una preselecciones para disminuir el número de correspondencias cuando es muy alto

Número de iteraciones

- Sea p la probabilidad de que una correspondencia es válida
- Sea P la probabilidad de éxito luego de S intentos
- La probabilidad en un intento de que todas las k muestras sean valores típicos es p^k.
- La probabilidad de que todos los S intentos fallen es

$$1 - P = (1 - p^k)^S$$

- El menor número de intentos es entonces

$$S = \frac{\log(1 - P)}{\log(1 - p^k)}$$

- Debe usarse el **menor** número k posible

Resumen

Este documento ha sido elaborado con software libre incluyendo LaTeX, Beamer, GNUPlot, GNU/Octave, XFig, Inkscape, LTI-Lib-2, GNU-Make, Kazam, Xournal y Subversion en GNU/Linux