

Corisco: Robust edgel-based orientation estimation for generic camera models

Nicolau L. Werneck

<http://nic.hpavc.net>

Supervisor: Profa. Dra. Anna Helena Reali Costa
Intelligent Techniques Laboratory, LTI — PCS — Poli

Universidade de São Paulo

Augsburg, Deutschland — 8/10/2015

Meta-introduction

Nicolau Werneck, Sc.D.

- E.Eng. graduated from UFMG, Unicamp and USP.
- Specialized in signal processing and pattern recognition, especially in computer vision and parameter estimation problems.
- Ex-Google and ex-Geekie.
- Wants to help robots help us.

Presentation adapted from my doctorate defense.
Results were published in Werneck and Costa [2013].

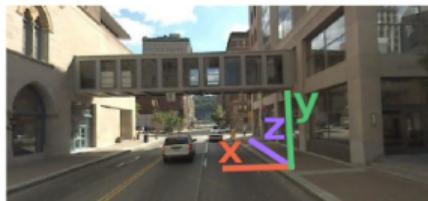
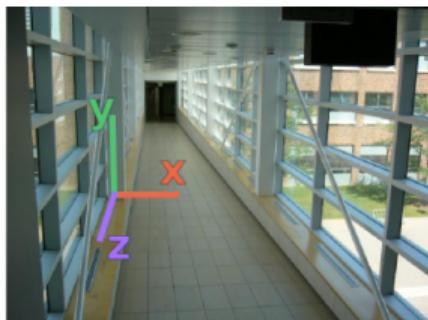
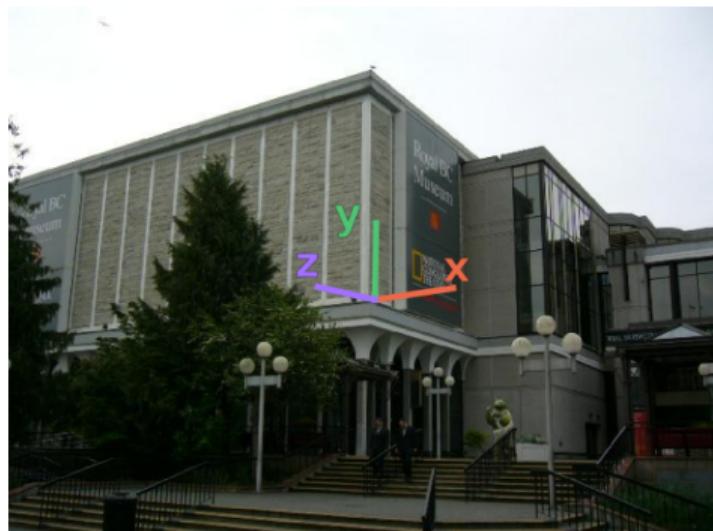
Presentation Schedule

- 1 Problem introduction
- 2 How *Corisco* works
- 3 Experiments
- 4 Conclusion

Anthropic Environments

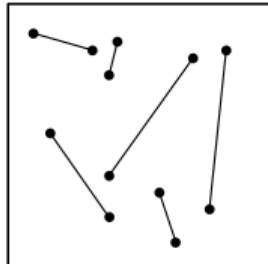
An *anthropic environment* is composed by straight lines, parallel to the directions of a *natural reference frame*.

The orientation we find is a three-dimensional rotation between the natural and the camera reference frame.

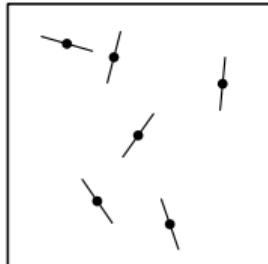


Edgels and straight lines

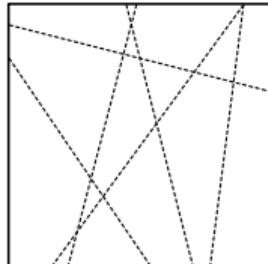
Edgels are points samples over curves or (straight) lines.



Segmentos

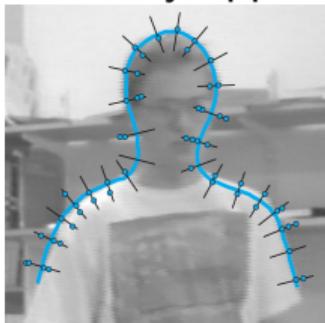


Edgels



Retas

Edgels have many applications.



Proposal

We proposed a monocular vision method, denominated *Corisco*, that can estimate the orientation of a camera relative to an anthropic environment.

Evolution of existing edgel based methods [Coughlan and Yuille, 2003].

Application examples:

- Guiding a mobile robot.
- Initial estimates for multi-view reconstruction.
- Object orientation estimation.

Proposal

We proposed a **monocular** vision method, denominated *Corisco*, that can estimate the orientation of a camera relative to an anthropic environment.

Evolution of existing edgel based methods [Coughlan and Yuille, 2003].

Application examples:

- Guiding a mobile robot.
- Initial estimates for multi-view reconstruction.
- Object orientation estimation.

Proposal

We proposed a monocular vision method, denominated *Corisco*, that can estimate the orientation of a camera relative to an anthropic environment.

Evolution of existing edgel based methods [Coughlan and Yuille, 2003].

Application examples:

- Guiding a mobile robot.
- Initial estimates for multi-view reconstruction.
- Object orientation estimation.

Contributions

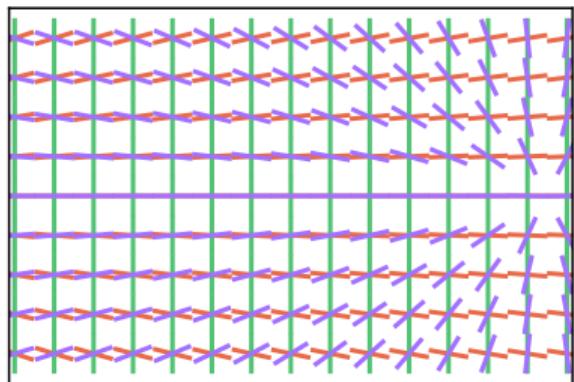
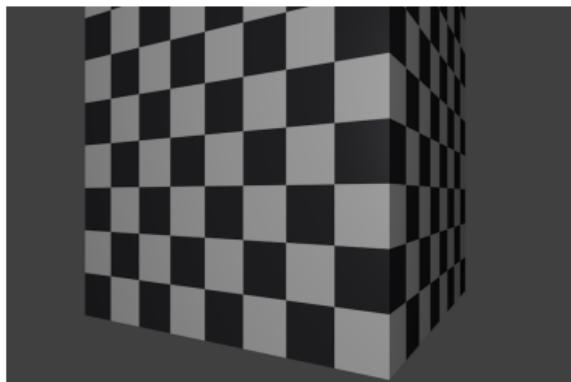
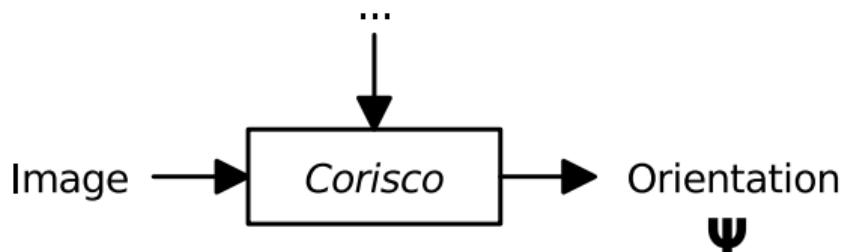
Corisco has the following peculiarities:

- Supports any possible camera model.
- Compromise between speed and precision.
- No assumptions about the solution.
- Dismisses the use of very costly operations (\sin , \tan , \exp , \log). Uses the function $x^{-1/2}$.

High-level view

Inputs: Image, control and intrinsic parameters.

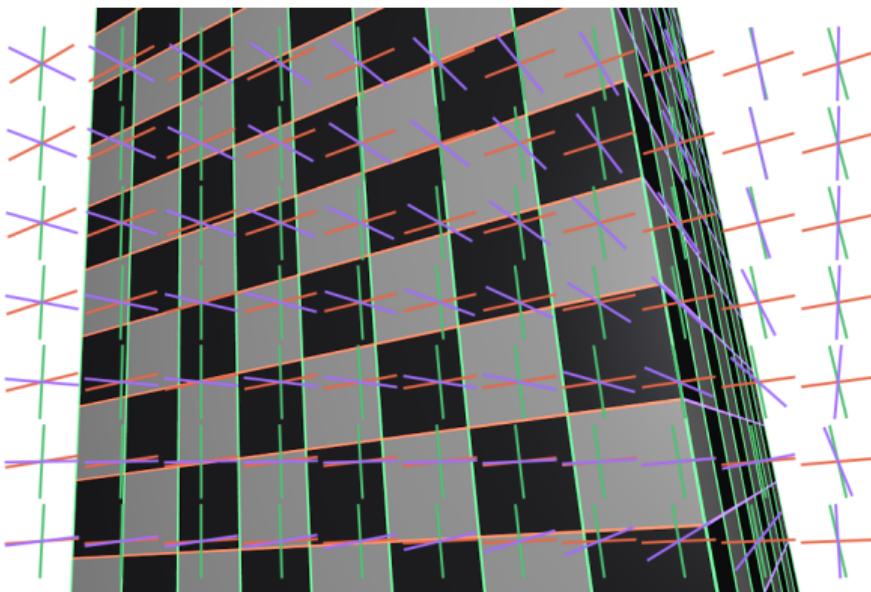
Output: Orientation Ψ (three-dimensional rotation).



Geometria

A direção \mathbf{v} de uma reta sobre o ponto \mathbf{p} depende de Ψ .

$$(\Psi, p) \rightarrow v$$



Objetivo

Nossa pergunta

Qual seria a melhor forma de estimar a orientação de câmera em tempo real em um ambiente antrópico a partir de uma única imagem distorcida?

Objetivo

Nossa pergunta

*Qual seria a melhor forma de estimar a orientação de câmera em **tempo real** em um ambiente antrópico a partir de uma única **imagem distorcida**?*

Edgels *versus* retas

A orientação pode ser encontrada por retas (Caprile and Torre [1990], Cipolla et al. [1999], Rother [2002]).

- Mais intuitivos.
- Alcançam boas precisões.

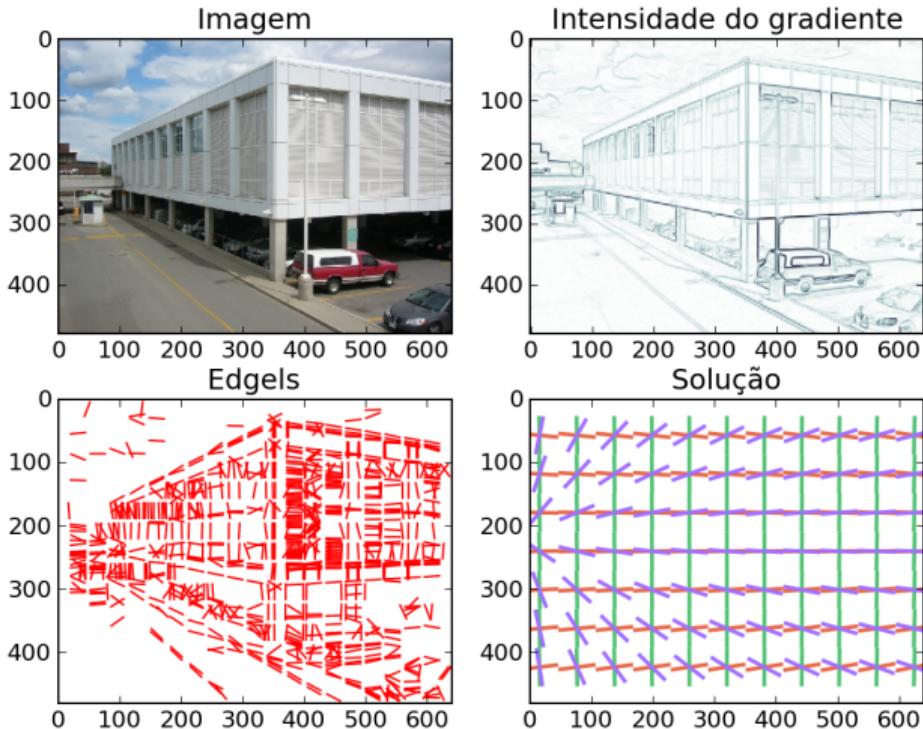
No entanto:

- Restrição à projeção perspectiva.
- Complexidade da extração de retas.

Edgels permitem lidar com distorções e são mais fáceis de extrair e de sub-amostrar.

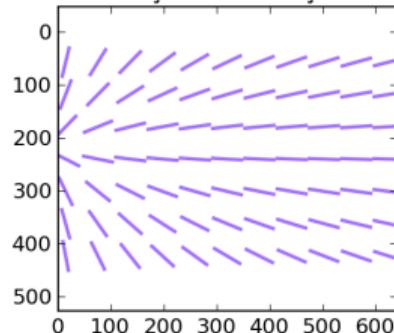
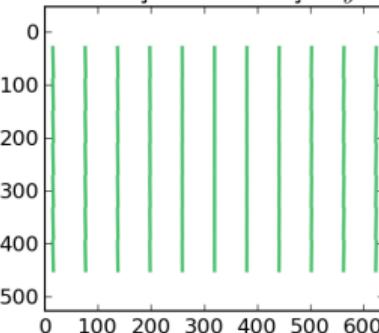
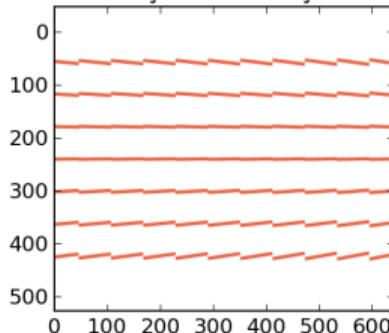
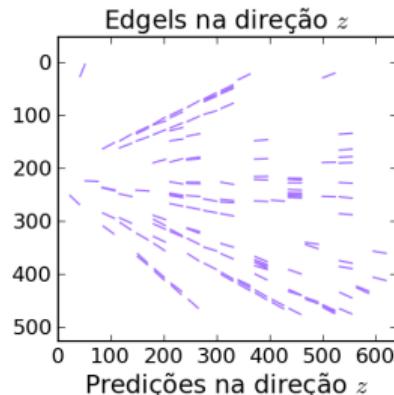
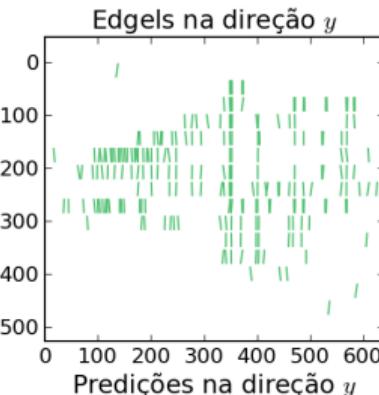
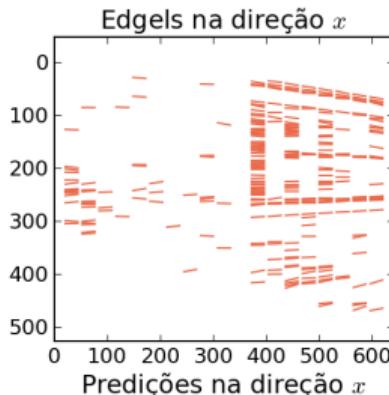
Detalhamento do Corisco

Exemplo de estimativa de orientação



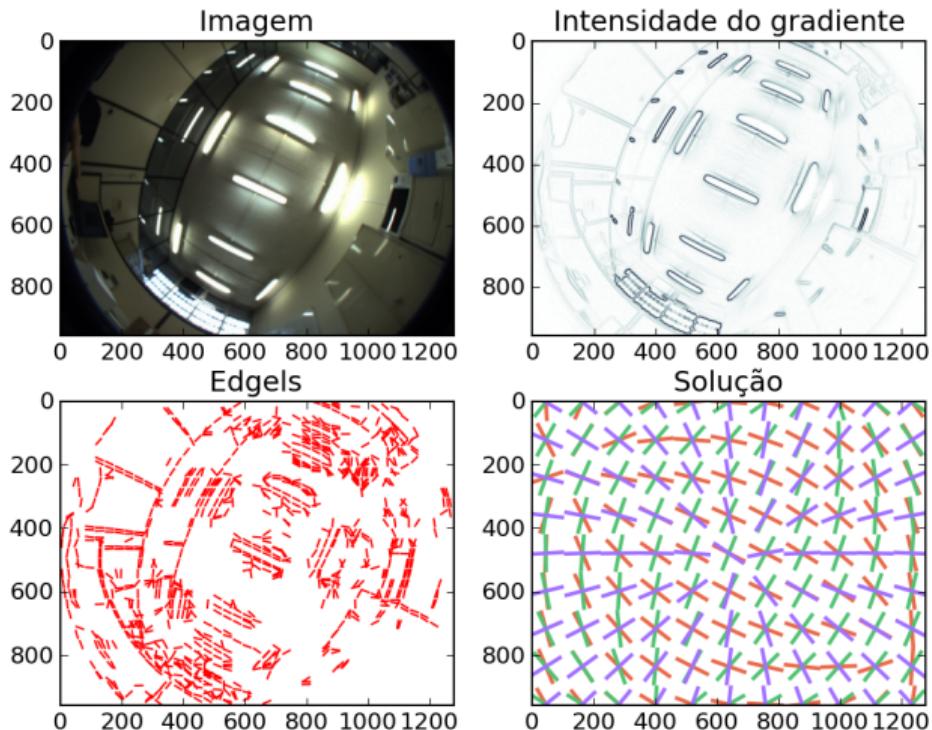
Detalhamento do Corisco

Edgels classificados e direções preditas



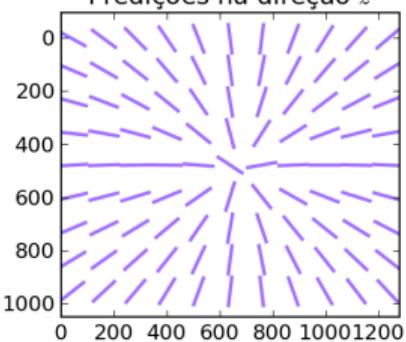
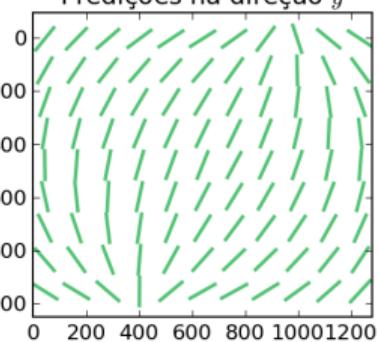
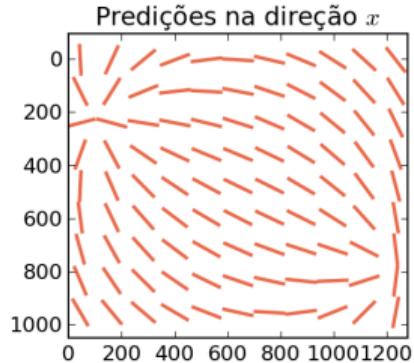
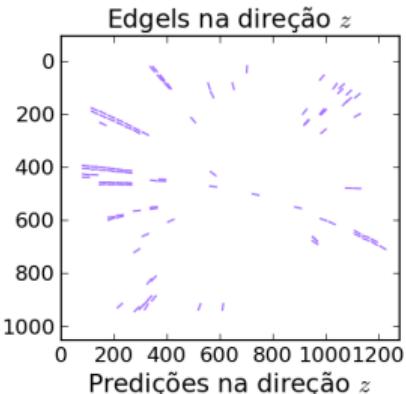
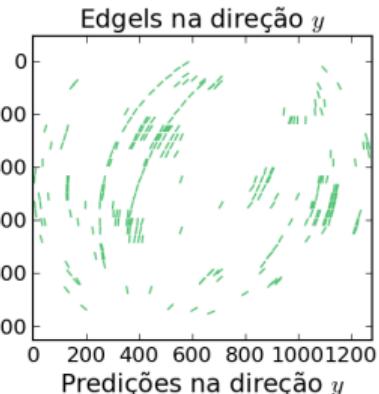
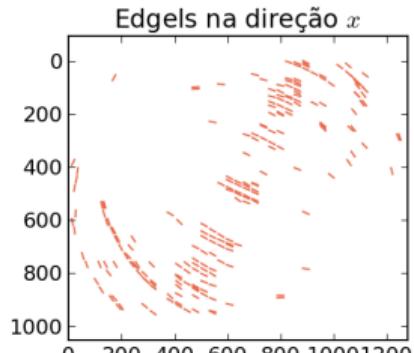
Detalhamento do Corisco

Exemplo de estimativa de orientação

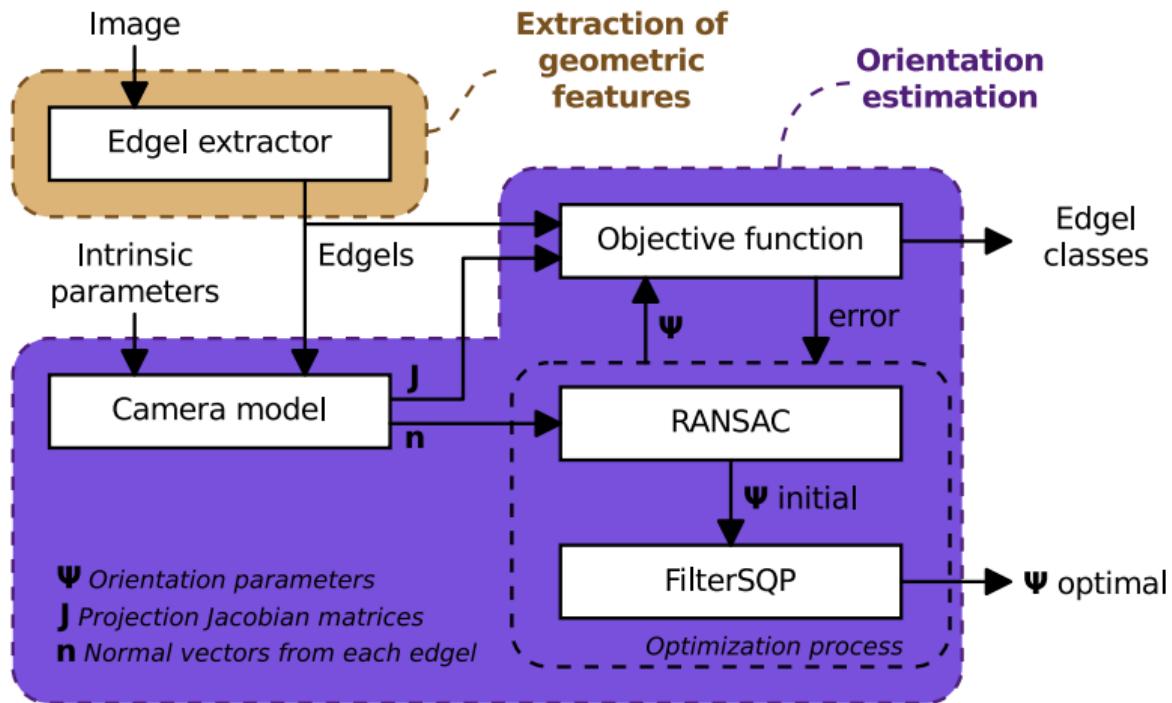


Detalhamento do Corisco

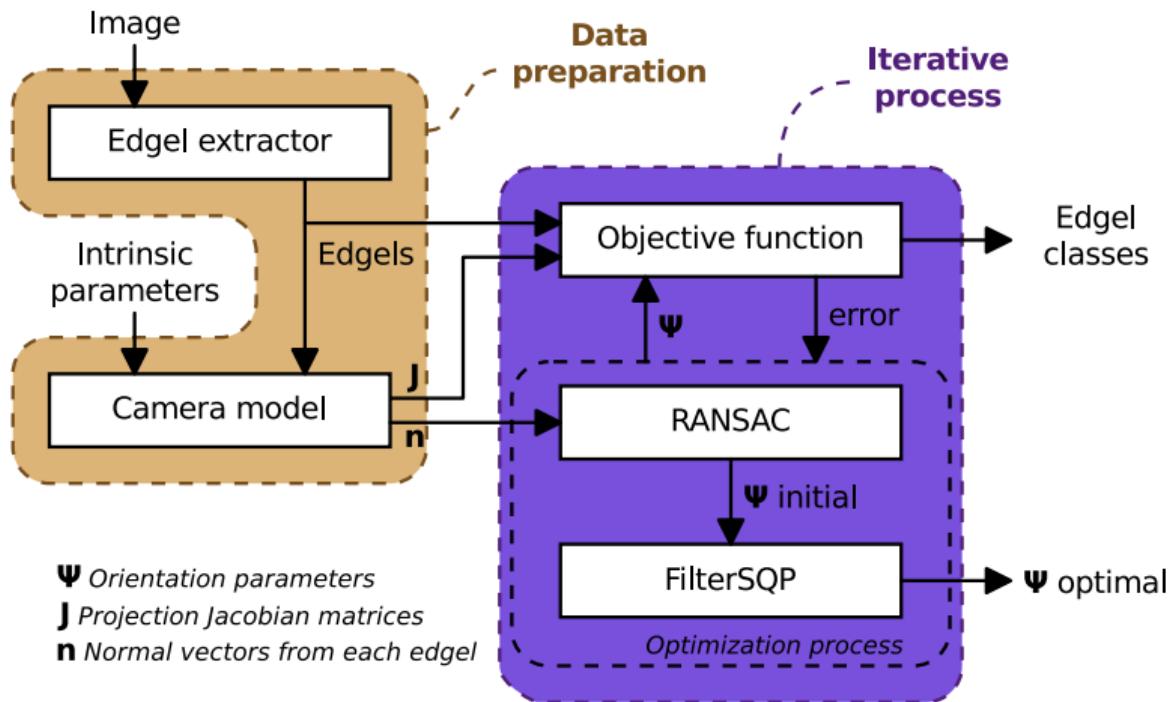
Edgels classificados e direções preditas



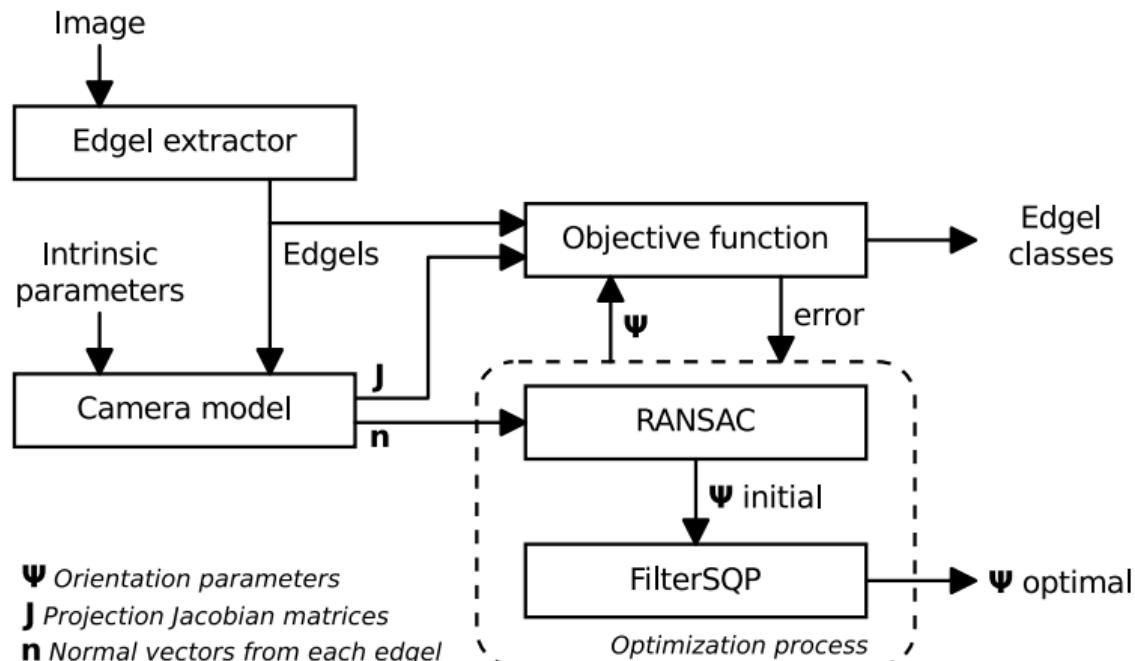
Block diagram



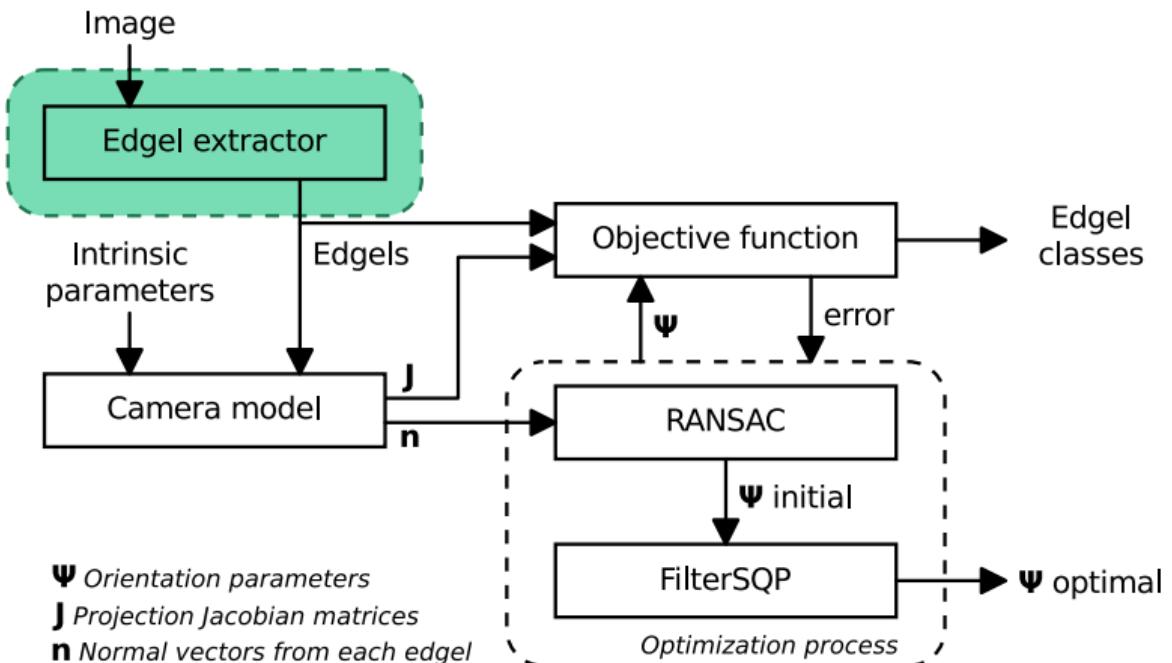
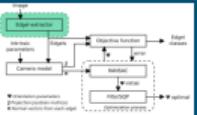
Block diagram

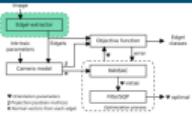


Block diagram



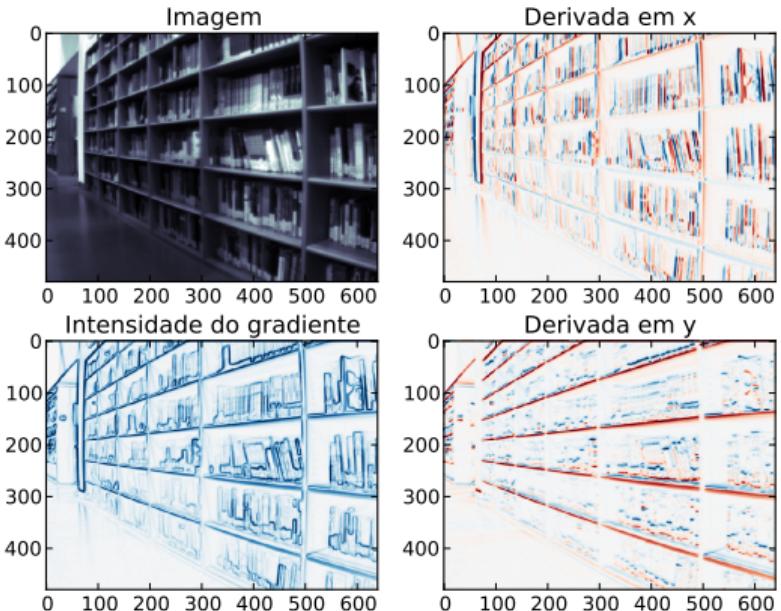
Extração de edgels

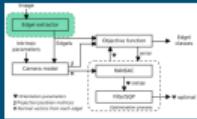




Extração de edgels

- Similar à detecção de bordas de Canny.
- Bordas são máximos locais na direção do gradiente.

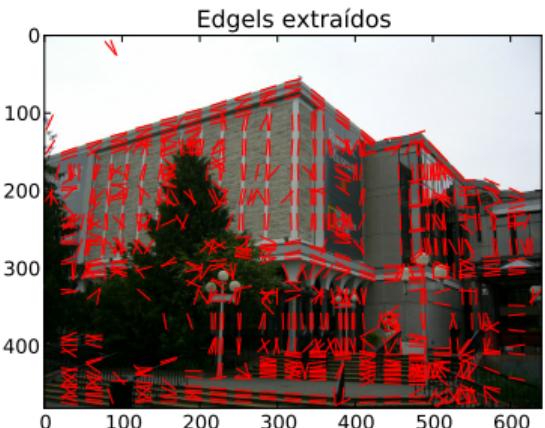
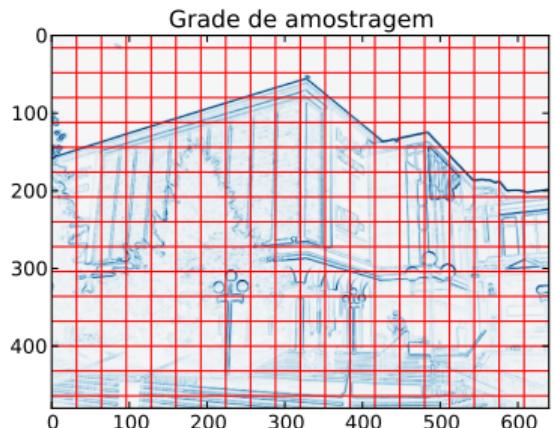




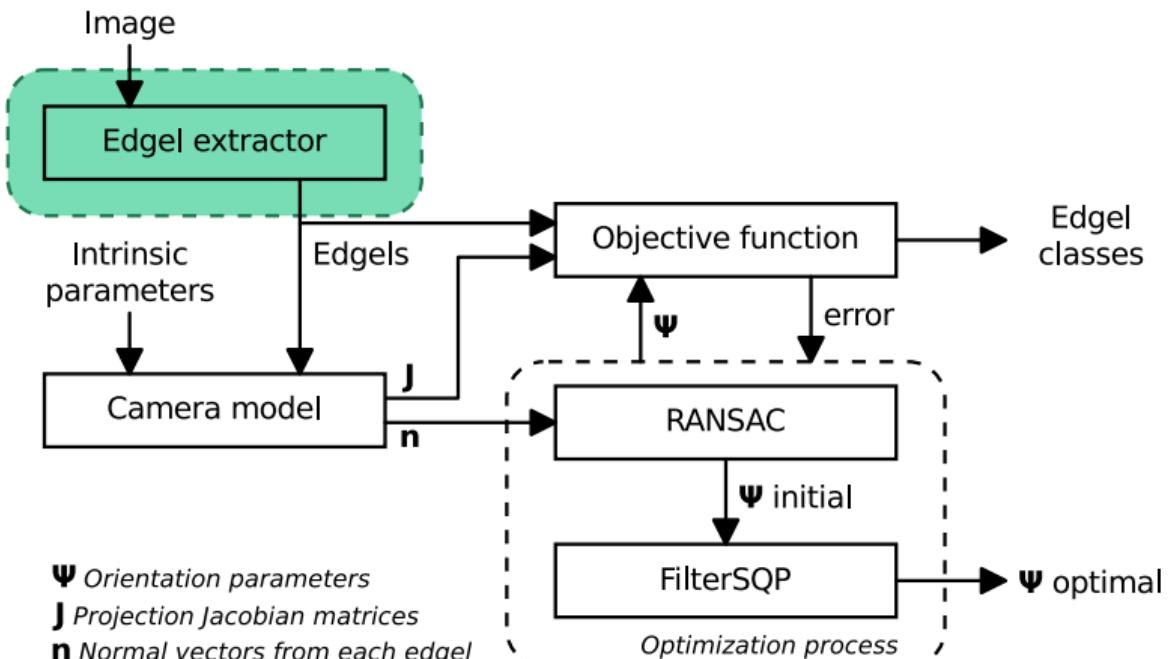
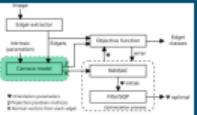
Extração de edgels

Varredura da imagem sobre um conjunto de linhas e colunas que formam uma *máscara em forma de grade*.

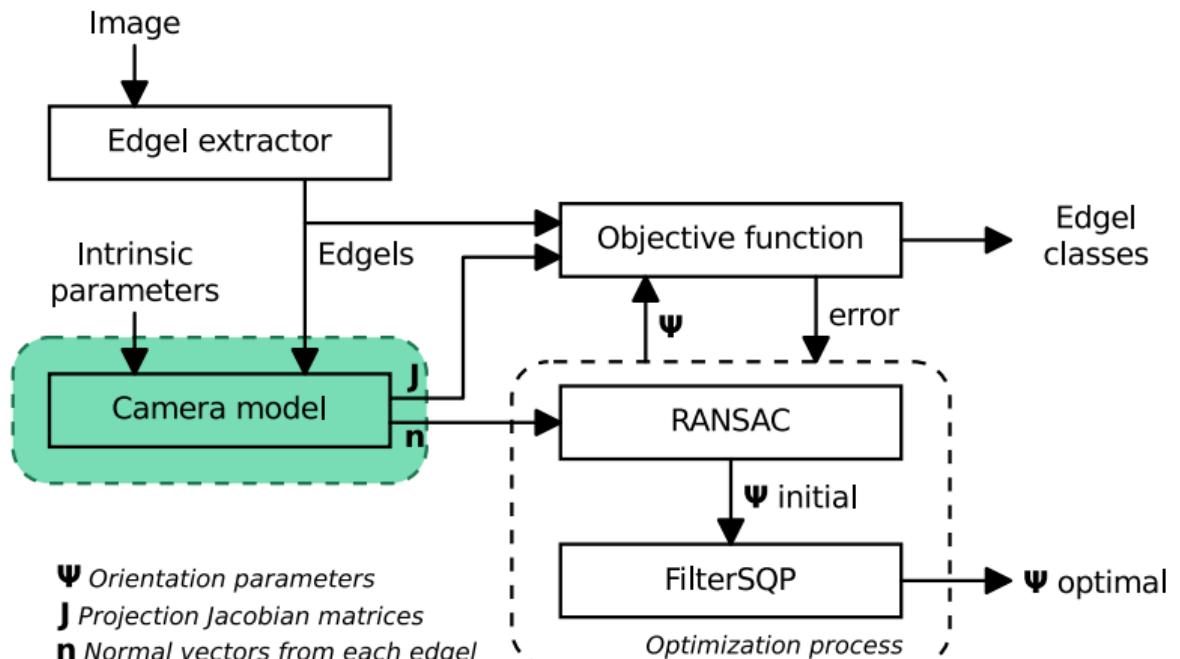
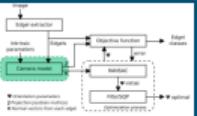
Cada borda encontrada produz um edgel. Sua direção deve ser aproximadamente ortogonal à da linha varrida.



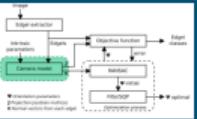
Modelos de câmera



Modelos de câmera

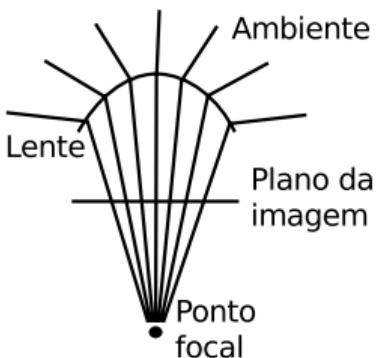
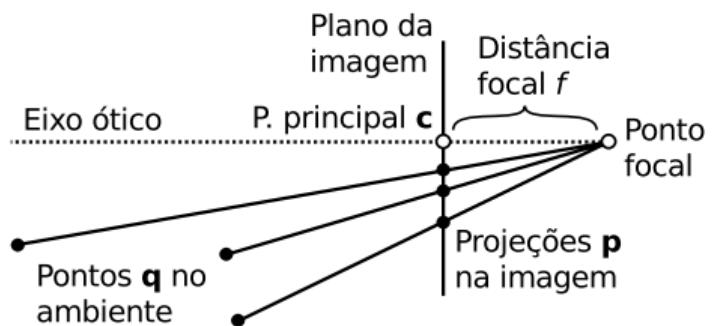


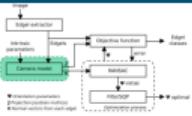
Modelos de câmera



Mapeamento bijetivo entre os pontos da imagem e direções ao redor do ponto focal da câmera.

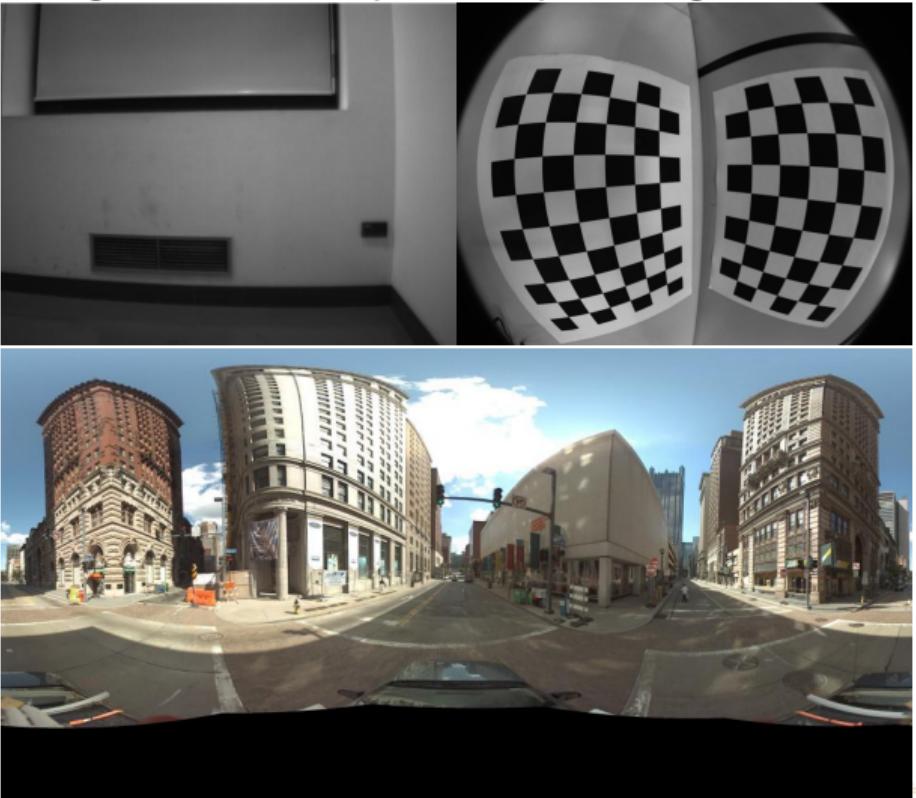
$$\mathbf{q} \rightleftharpoons \mathbf{p}$$



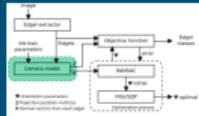


Exemplos de modelos de câmera

Grande angular, olho-de-peixe, equiretangular.



Equações de modelos de câmera



Perspectiva

$$\mathbf{p}^x = (\mathbf{q}^x / \mathbf{q}^z) f + \mathbf{c}^x$$

$$\mathbf{p}^y = (\mathbf{q}^y / \mathbf{q}^z) f + \mathbf{c}^y$$

Equiretangular (lat-lon)

$$\mathbf{p}^x = f \tan^{-1}(\mathbf{q}^z, \mathbf{q}^x)$$

$$\mathbf{p}^y = f \sin^{-1}(\mathbf{q}^y / |\mathbf{q}|)$$

Harris (distorção radial)

$$g(x) = \frac{1}{\sqrt{1-2\kappa x^2}}$$

$$\mathbf{p}^x = \mathbf{p}'^x g(|\mathbf{p}'|) + \mathbf{c}^x$$

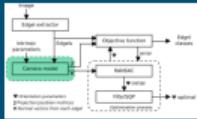
$$\mathbf{p}^y = \mathbf{p}'^y g(|\mathbf{p}'|) + \mathbf{c}^y$$

Polar Equidistante (olho-de-peixe)

$$\varphi = \cos^{-1}(\mathbf{q}^z / |\mathbf{q}|)$$

$$\mathbf{p}^x = \varphi \frac{\mathbf{q}^x}{\sqrt{\mathbf{q}^{x^2} + \mathbf{q}^{y^2}}} f + \mathbf{c}^x$$

$$\mathbf{p}^y = \varphi \frac{\mathbf{q}^y}{\sqrt{\mathbf{q}^{x^2} + \mathbf{q}^{y^2}}} f + \mathbf{c}^y$$

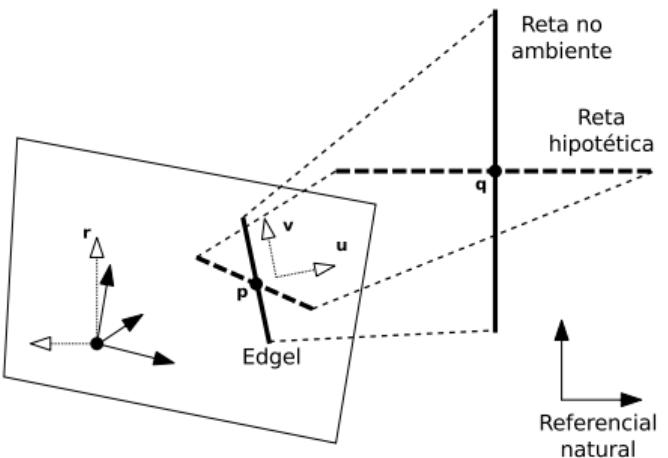


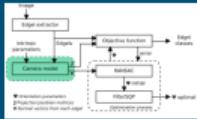
Projeção de um edgel

Um ponto \mathbf{q}_n de uma reta na direção \mathbf{r}_k é projetado em \mathbf{p}_n , produzindo um edgel com direção

$$\mathbf{v}_{nk} \propto \mathbf{J}_n \mathbf{r}_k \quad (\mathbf{v} \leftarrow (\Psi, \mathbf{p}))$$

A matriz Jacobiana da projeção \mathbf{J}_n depende de \mathbf{p}_n . A direção ortogonal a \mathbf{v}_n é denominada \mathbf{u}_n .



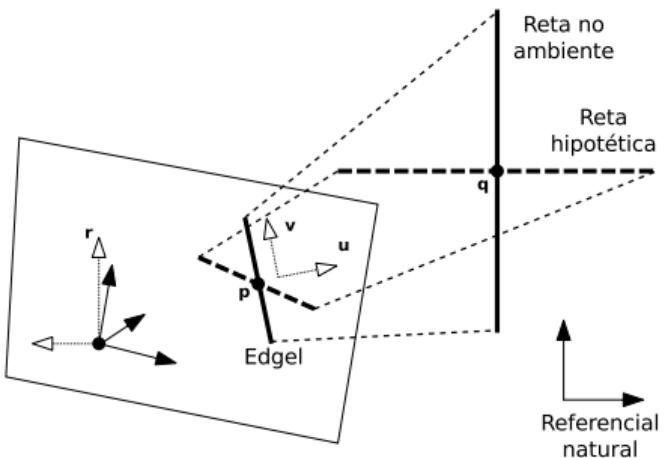


Projeção de um edgel

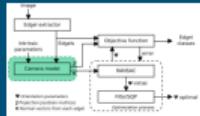
Um ponto \mathbf{q}_n de uma reta na direção \mathbf{r}_k é projetado em \mathbf{p}_n , produzindo um edgel com direção

$$\mathbf{v}_{nk} \propto \mathbf{J}_n \mathbf{r}_k \quad (\mathbf{v} \leftarrow (\Psi, \mathbf{p}))$$

A matriz Jacobiana da projeção \mathbf{J}_n depende de \mathbf{p}_n . A direção ortogonal a \mathbf{v}_n é denominada \mathbf{u}_n .



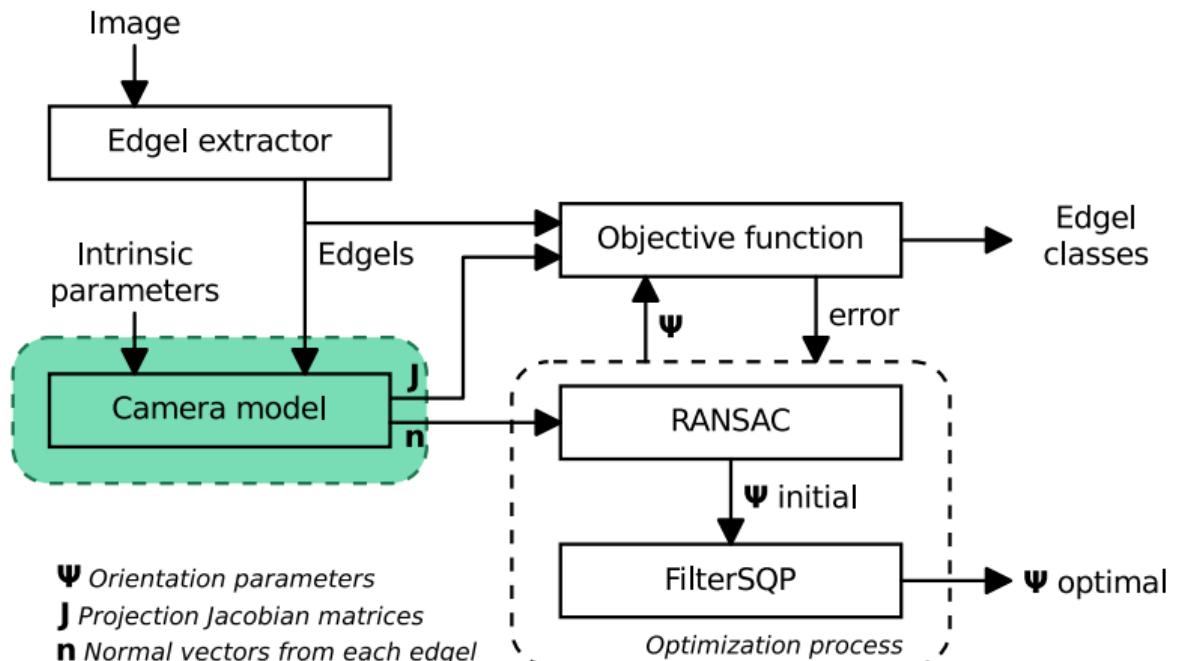
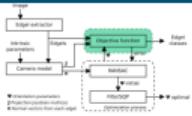
Normal de um edgel



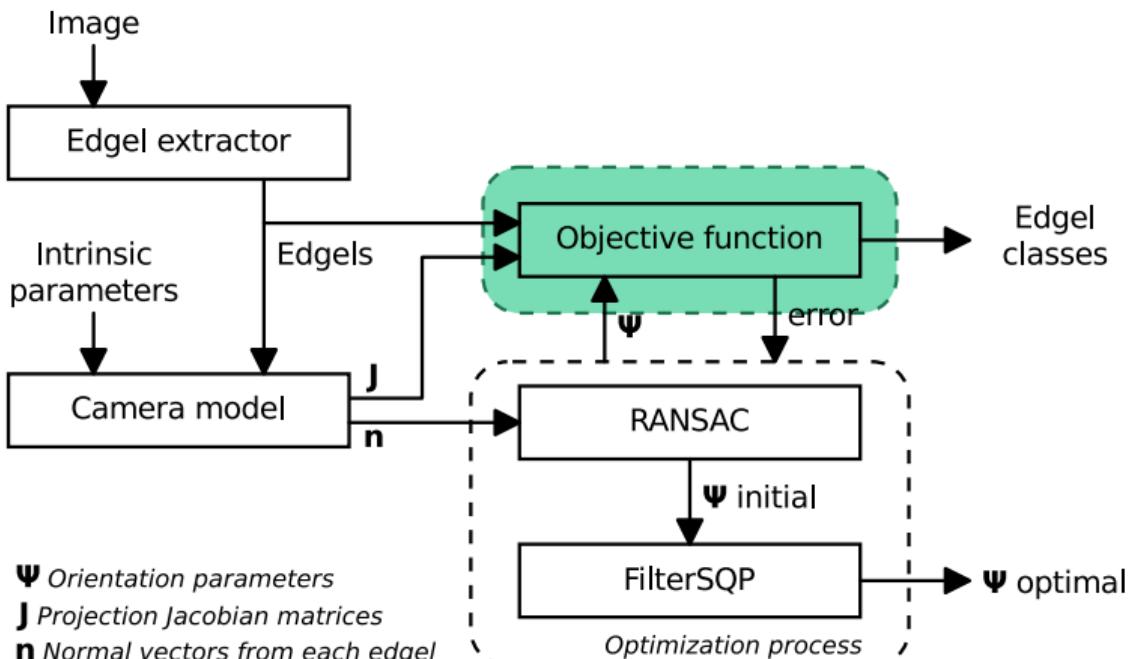
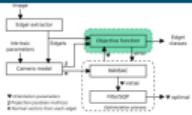
Direção de um plano definido pelo ponto focal e por um edgel, ou uma reta correspondente no ambiente.

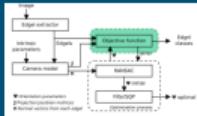
$$\mathbf{n}_k = \mathbf{u}_k^x \mathbf{J}_k^x + \mathbf{u}_k^y \mathbf{J}_k^y$$

Função objetivo



Função objetivo





Função objetivo

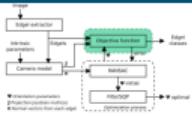
A função objetivo é o cerne do método.

- Independe da técnica de extração.
- Determina o resultado.
- Conduz a escolha do algoritmo de otimização.

A expressão tem a forma de uma soma de erros obtidos de cada observação.

$$y_n = x_n - \hat{x}_n(\Psi)$$

$$F(\Psi) = \sum_n (y_n)^2$$



Matriz de rotação

O cálculo das predições começa com o cálculo das direções do referencial natural \mathbf{r}_k em função de Ψ .

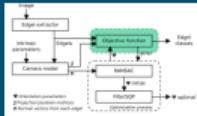
O Corisco trabalha com quaternions.

$$\Psi = (\Psi^a, \Psi^b, \Psi^c, \Psi^d) \quad |\Psi| = 1$$

Para uma orientação Ψ qualquer temos

$$\mathbf{R}(\Psi) = [\mathbf{r}_x \mathbf{r}_y \mathbf{r}_z]^T =$$

$$\begin{bmatrix} \Psi^{a2} + \Psi^{b2} - \Psi^{c2} - \Psi^{d2} & 2\Psi^b\Psi^c + 2\Psi^a\Psi^d & 2\Psi^b\Psi^d - 2\Psi^a\Psi^c \\ 2\Psi^b\Psi^c - 2\Psi^a\Psi^d & \Psi^{a2} - \Psi^{b2} + \Psi^{c2} - \Psi^{d2} & 2\Psi^c\Psi^d + 2\Psi^a\Psi^b \\ 2\Psi^b\Psi^d + 2\Psi^a\Psi^c & 2\Psi^c\Psi^d - 2\Psi^a\Psi^b & \Psi^{a2} - \Psi^{b2} - \Psi^{c2} + \Psi^{d2} \end{bmatrix}$$



Resíduo de um edgel

Dados os \mathbf{r}_k , calculam-se as direções preditas \mathbf{v}_{nk} utilizando \mathbf{J}_k .

$$\Psi \rightarrow \mathbf{r}_k \rightarrow \mathbf{v}_{nk}$$

$$\mathbf{v}_{nk} \propto \mathbf{J}_n \mathbf{r}_k$$

O resíduo é obtido entre:

- \mathbf{v}_n (ou \mathbf{u}_n) medidos da imagem.
- \mathbf{v}_{nk} preditas a partir de Ψ e \mathbf{p}_n .

Métodos anteriores:

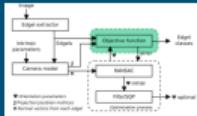
$$\angle \mathbf{v} = \arctan(\mathbf{v}^x, \mathbf{v}^y)$$

$$\angle \mathbf{v}_n - \angle \mathbf{v}_{nk}$$

Corisco:

$$\mathbf{u}_n \mathbf{v}_{nk}$$

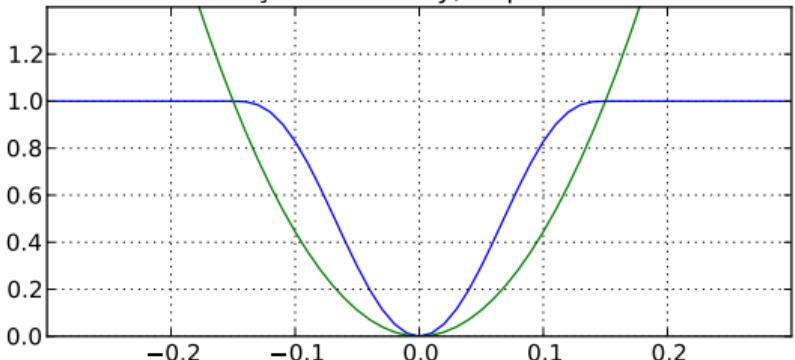
Técnicas de estimação

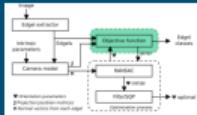


Comparação de técnicas já utilizadas com edgels:

	MAP	EM	M-estimação
Modelos	Probabilístico	Probabilístico	$\rho(x)$
Classes	4	4	3
Classificação	×	Iterativa	Direta

Funções de Tukey, e quadrática





Expressão da função objetivo

Estimador MAP (Coughlan and Yuille [2003]):

$$F(\Psi) = - \sum_n \log \left(\sum_k p(c_n = k) p(\angle \mathbf{v}_n | \Psi, \mathbf{p}_n, c_n = k) \right)$$

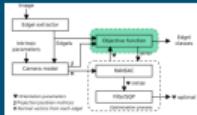
Estimador EM (Schindler and Dellaert [2004]):

$$F(\Psi) = - \sum_n \sum_k p(c_n = k) \log(p(\angle \mathbf{v}_n | \Psi, \mathbf{p}_n, c_n = k))$$

$$F(\Psi) = \sum_n \sum_k p(c_n = k) (\angle \mathbf{v}_n - \angle \mathbf{v}_{nk})^2$$

Corisco:

$$F(\Psi) = \sum_n \min_k \rho(\mathbf{u}_n \mathbf{v}_{nk})$$



Expressão da função objetivo

Estimador MAP (Coughlan and Yuille [2003]):

$$F(\Psi) = - \sum_n \log \left(\sum_k p(c_n = k) p(\angle \mathbf{v}_n | \Psi, \mathbf{p}_n, c_n = k) \right)$$

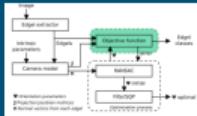
Estimador EM (Schindler and Dellaert [2004]):

$$F(\Psi) = - \sum_n \sum_k p(c_n = k) \log(p(\angle \mathbf{v}_n | \Psi, \mathbf{p}_n, c_n = k))$$

$$F(\Psi) = \sum_n \sum_k p(c_n = k) (\angle \mathbf{v}_n - \angle \mathbf{v}_{nk})^2$$

Corisco:

$$F(\Psi) = \sum_n \min_k \rho(\mathbf{u}_n \mathbf{v}_{nk})$$



Expressão da função objetivo

Estimador MAP (Coughlan and Yuille [2003]):

$$F(\Psi) = - \sum_n \log \left(\sum_k p(c_n = k) p(\angle \mathbf{v}_n | \Psi, \mathbf{p}_n, c_n = k) \right)$$

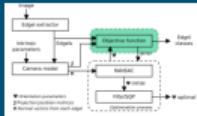
Estimador EM (Schindler and Dellaert [2004]):

$$F(\Psi) = - \sum_n \sum_k p(c_n = k) \log(p(\angle \mathbf{v}_n | \Psi, \mathbf{p}_n, c_n = k))$$

$$F(\Psi) = \sum_n \sum_k p(c_n = k) (\angle \mathbf{v}_n - \angle \mathbf{v}_{nk})^2$$

Corisco:

$$F(\Psi) = \sum_n \min_k \rho(\mathbf{u}_n \mathbf{v}_{nk})$$



Expressão da função objetivo

Estimador MAP (Coughlan and Yuille [2003]):

$$F(\Psi) = - \sum_n \log \left(\sum_k p(c_n = k) p(\angle \mathbf{v}_n | \Psi, \mathbf{p}_n, c_n = k) \right)$$

Estimador EM (Schindler and Dellaert [2004]):

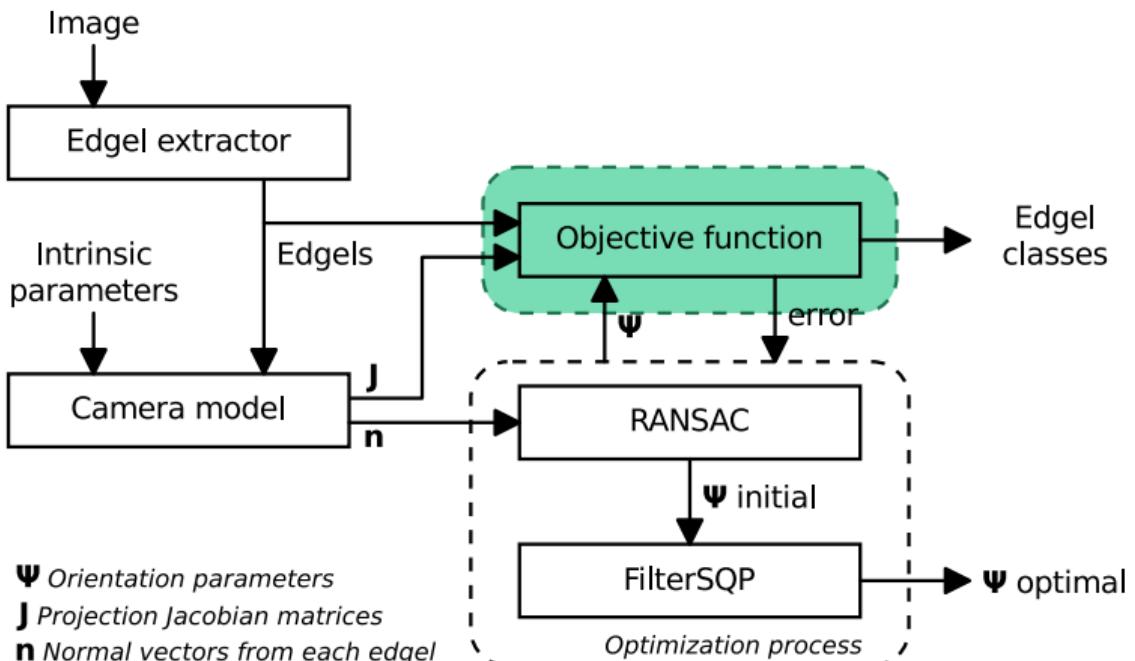
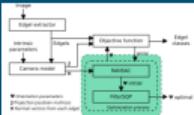
$$F(\Psi) = - \sum_n \sum_k p(c_n = k) \log(p(\angle \mathbf{v}_n | \Psi, \mathbf{p}_n, c_n = k))$$

$$F(\Psi) = \sum_n \sum_k p(c_n = k) (\angle \mathbf{v}_n - \angle \mathbf{v}_{nk})^2$$

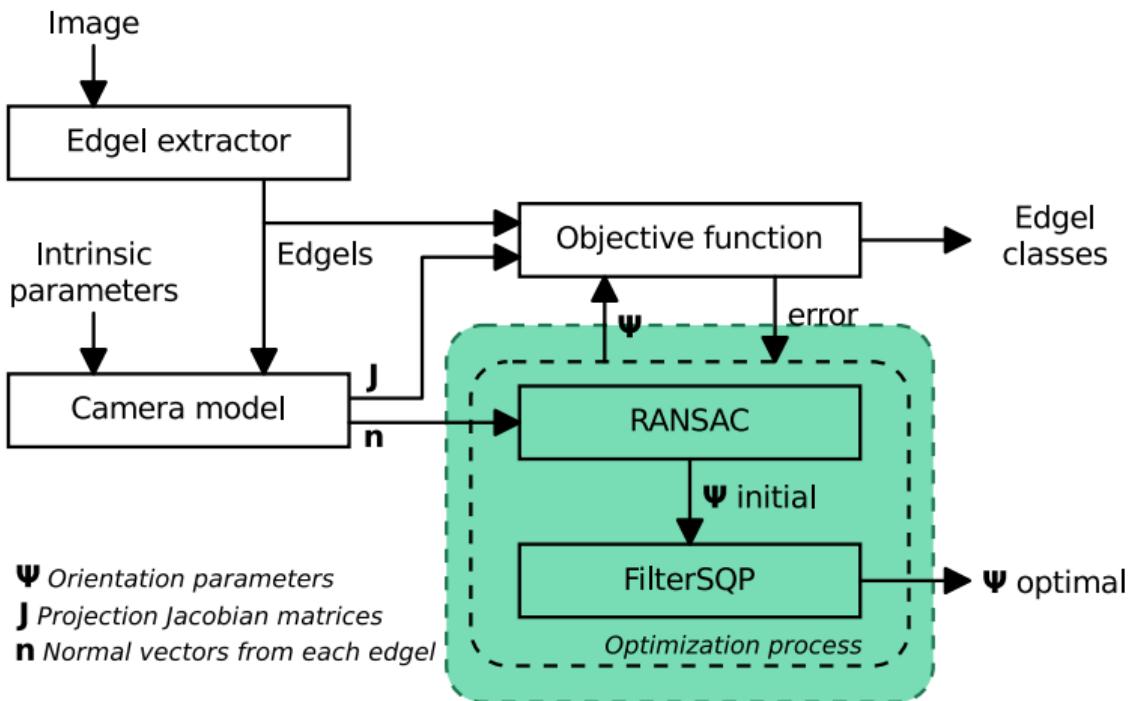
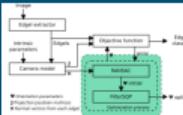
Corisco:

$$F(\Psi) = \sum_n \min_k \rho(\mathbf{u}_n \mathbf{v}_{nk})$$

Otimização



Otimização



Otimização

Primeiro passo: RANSAC, busca estocástica guiada pelos dados. Inerentemente ineficiente e impreciso.

→ Ψ inicial

Segundo passo: FilterSQP, otimização contínua, mais eficiente e precisa do que o RANSAC.

→ Ψ final

Otimização

Primeiro passo: RANSAC, busca estocástica guiada pelos dados. Inerentemente ineficiente e impreciso.

→ Ψ inicial

Segundo passo: FilterSQP, otimização contínua, mais eficiente e precisa do que o RANSAC.

→ Ψ final

Otimização

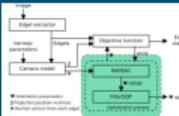
Primeiro passo: RANSAC, busca estocástica guiada pelos dados. Inerentemente ineficiente e impreciso.

→ Ψ inicial

Segundo passo: FilterSQP, otimização contínua, mais eficiente e precisa do que o RANSAC.

→ Ψ final

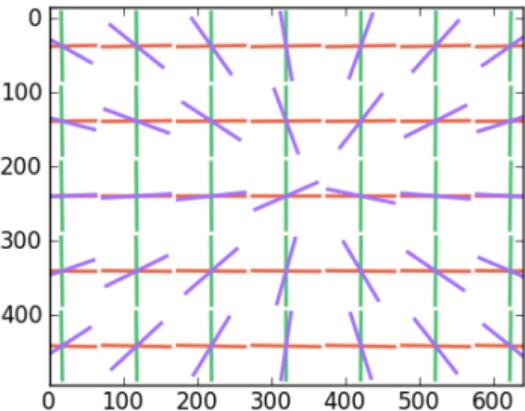
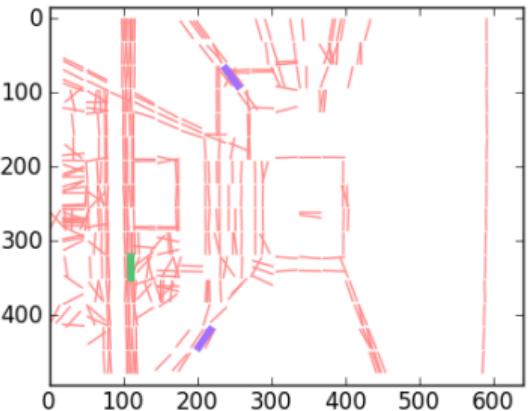
RANSAC

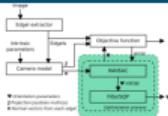


Triplas de observações são selecionadas ao acaso. De cada tripla calcula-se um Ψ hipotético, utilizando n_k .

O Ψ de menor $F(\Psi)$ é retido como estimativa inicial.

Iteração 1759, $F=146.5$





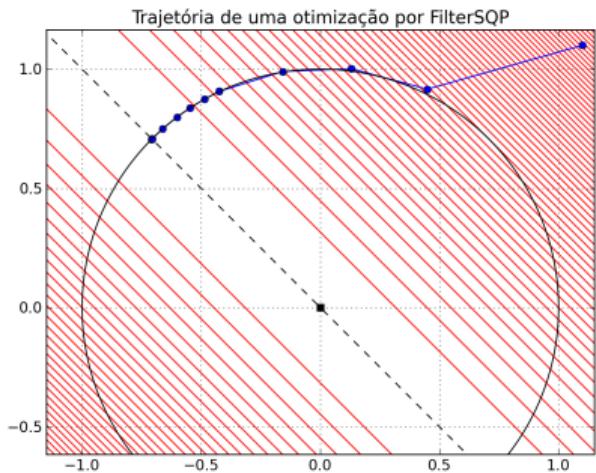
FilterSQP

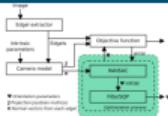
Minimizar $F(\Psi)$ sobre as 4D e restrito a $|\Psi| = 1$.

Programa não-linear →

SQP: Sequential Quadratic Programming

O FilterSQP (Fletcher and Leyffer [2002]) dispensa funções de penalidade.





Derivada

No Corisco as derivadas em Ψ são calculadas através de fórmulas fechadas.

$$\frac{\partial F}{\partial \Psi^a}(\Psi) = \sum_{nk} K_{nk} \rho'(\mathbf{u}_n \mathbf{v}_{nk}) \left(u_n^x \frac{\partial \mathbf{v}_{nk}^x}{\partial \Psi^a} + u_n^y \frac{\partial \mathbf{v}_{nk}^y}{\partial \Psi^a} \right)$$

As derivadas das direções \mathbf{r}_k são triviais:

$$\frac{\partial \mathbf{r}^x}{\partial \Psi^a} = 2(\Psi^a, \Psi^d, -\Psi^c)$$

$$\frac{\partial \mathbf{r}^x}{\partial \Psi^b} = 2(\Psi^b, \Psi^c, \Psi^d)$$

...

Experimentos

Foram realizados 3 experimentos que permitem avaliar o desempenho do *Corisco*.

Cada experimento utilizou um conjunto de imagens, e um método diferente para obter orientações de referência.

O erro observado é o deslocamento em graus da “rotação residual” entre cada estimativa e referência.

O *Corisco* foi executado variando-se

- o tamanho da grade,
- e o número de iterações do RANSAC.

Imagens: 101 imagens de ambientes antrópicos.

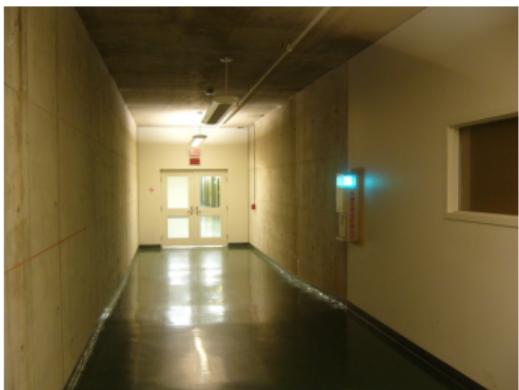
Modelo: Perspectiva.

Referência: Método semi-automático baseado em retas.

Comparaçao: Métodos testados por Denis et al. [2008].

Parâmetros do modelo, orientações de referência e estatísticas de desempenho fornecidos pelos autores.

YorkUrbanDB



YorkUrbanDB

Method	Time [s]	Error			
		Mean	σ	1/4	Median
EM Newton	27+?	4.00°	1.00°	1.15°	2.61°
MAP Quasi-Newton	6+?	4.00°	1.00°	1.32°	2.39°
EM Quasi-Newton	1+?	9.00°	1.00°	4.04°	6.21°
J-linkage	1.13	8.23°	13.76°	1.14°	2.36°
<i>Corisco</i> $C_r = 10^4$ $C_g = 1$	47.20	1.51°	3.26°	0.69°	1.09°
<i>Corisco</i> $C_r = 10^4$ $C_g = 4$	16.68	1.71°	3.35°	0.72°	1.14°
<i>Corisco</i> $C_r = 10^4$ $C_g = 32$	7.57	2.43°	4.03°	0.97°	1.54°
<i>Corisco</i> $C_r = 10^3$ $C_g = 1$	8.12	1.70°	3.22°	0.70°	1.11°
<i>Corisco</i> $C_r = 10^3$ $C_g = 4$	2.50	2.02°	3.86°	0.81°	1.24°
<i>Corisco</i> $C_r = 10^3$ $C_g = 32$	0.99	2.44°	3.54°	1.00°	1.68°
<i>Corisco</i> $C_r = 200$ $C_g = 1$	5.34	2.08°	3.38°	0.72°	1.22°
<i>Corisco</i> $C_r = 200$ $C_g = 4$	1.89	3.27°	6.38°	0.85°	1.34°
<i>Corisco</i> $C_r = 200$ $C_g = 32$	0.45	3.29°	4.99°	0.99°	1.72°
					3.46°

YorkUrbanDB

Method	Time [s]	Error				
		Mean	σ	1/4	Median	3/4
EM Newton	27+?	4.00°	1.00°	1.15°	2.61°	4.10°
MAP Quasi-Newton	6+?	4.00°	1.00°	1.32°	2.39°	4.07°
EM Quasi-Newton	1+?	9.00°	1.00°	4.04°	6.21°	10.33°
J-linkage	1.13	8.23°	13.76°	1.14°	2.36°	4.44°
<i>Corisco</i> $C_r = 10^4$ $C_g = 1$	47.20	1.51°	3.26°	0.69°	1.09°	1.51°
<i>Corisco</i> $C_r = 10^4$ $C_g = 4$	16.68	1.71°	3.35°	0.72°	1.14°	1.64°
<i>Corisco</i> $C_r = 10^4$ $C_g = 32$	7.57	2.43°	4.03°	0.97°	1.54°	2.42°
<i>Corisco</i> $C_r = 10^3$ $C_g = 1$	8.12	1.70°	3.22°	0.70°	1.11°	1.77°
<i>Corisco</i> $C_r = 10^3$ $C_g = 4$	2.50	2.02°	3.86°	0.81°	1.24°	1.80°
<i>Corisco</i> $C_r = 10^3$ $C_g = 32$	0.99	2.44°	3.54°	1.00°	1.68°	2.64°
<i>Corisco</i> $C_r = 200$ $C_g = 1$	5.34	2.08°	3.38°	0.72°	1.22°	1.78°
<i>Corisco</i> $C_r = 200$ $C_g = 4$	1.89	3.27°	6.38°	0.85°	1.34°	2.35°
<i>Corisco</i> $C_r = 200$ $C_g = 32$	0.45	3.29°	4.99°	0.99°	1.72°	3.46°

YorkUrbanDB

Method	Time [s]	Error				
		Mean	σ	1/4	Median	3/4
EM Newton	27+?	4.00°	1.00°	1.15°	2.61°	4.10°
MAP Quasi-Newton	6+?	4.00°	1.00°	1.32°	2.39°	4.07°
EM Quasi-Newton	1+?	9.00°	1.00°	4.04°	6.21°	10.33°
J-linkage	1.13	8.23°	13.76°	1.14°	2.36°	4.44°
<i>Corisco</i> $C_r = 10^4$ $C_g = 1$	47.20	1.51°	3.26°	0.69°	1.09°	1.51°
<i>Corisco</i> $C_r = 10^4$ $C_g = 4$	16.68	1.71°	3.35°	0.72°	1.14°	1.64°
<i>Corisco</i> $C_r = 10^4$ $C_g = 32$	7.57	2.43°	4.03°	0.97°	1.54°	2.42°
<i>Corisco</i> $C_r = 10^3$ $C_g = 1$	8.12	1.70°	3.22°	0.70°	1.11°	1.77°
<i>Corisco</i> $C_r = 10^3$ $C_g = 4$	2.50	2.02°	3.86°	0.81°	1.24°	1.80°
<i>Corisco</i> $C_r = 10^3$ $C_g = 32$	0.99	2.44°	3.54°	1.00°	1.68°	2.64°
<i>Corisco</i> $C_r = 200$ $C_g = 1$	5.34	2.08°	3.38°	0.72°	1.22°	1.78°
<i>Corisco</i> $C_r = 200$ $C_g = 4$	1.89	3.27°	6.38°	0.85°	1.34°	2.35°
<i>Corisco</i> $C_r = 200$ $C_g = 32$	0.45	3.29°	4.99°	0.99°	1.72°	3.46°

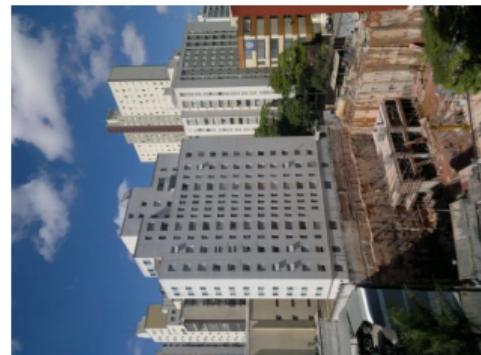
YorkUrbanDB

Method	Time [s]	Error				
		Mean	σ	1/4	Median	3/4
EM Newton	27+?	4.00°	1.00°	1.15°	2.61°	4.10°
MAP Quasi-Newton	6+?	4.00°	1.00°	1.32°	2.39°	4.07°
EM Quasi-Newton	1+?	9.00°	1.00°	4.04°	6.21°	10.33°
J-linkage	1.13	8.23°	13.76°	1.14°	2.36°	4.44°
<i>Corisco</i> $C_r = 10^4$ $C_g = 1$	47.20	1.51°	3.26°	0.69°	1.09°	1.51°
<i>Corisco</i> $C_r = 10^4$ $C_g = 4$	16.68	1.71°	3.35°	0.72°	1.14°	1.64°
<i>Corisco</i> $C_r = 10^4$ $C_g = 32$	7.57	2.43°	4.03°	0.97°	1.54°	2.42°
<i>Corisco</i> $C_r = 10^3$ $C_g = 1$	8.12	1.70°	3.22°	0.70°	1.11°	1.77°
<i>Corisco</i> $C_r = 10^3$ $C_g = 4$	2.50	2.02°	3.86°	0.81°	1.24°	1.80°
<i>Corisco</i> $C_r = 10^3$ $C_g = 32$	0.99	2.44°	3.54°	1.00°	1.68°	2.64°
<i>Corisco</i> $C_r = 200$ $C_g = 1$	5.34	2.08°	3.38°	0.72°	1.22°	1.78°
<i>Corisco</i> $C_r = 200$ $C_g = 4$	1.89	3.27°	6.38°	0.85°	1.34°	2.35°
<i>Corisco</i> $C_r = 200$ $C_g = 32$	0.45	3.29°	4.99°	0.99°	1.72°	3.46°

Imagens: 24+24 imagens de edifícios.

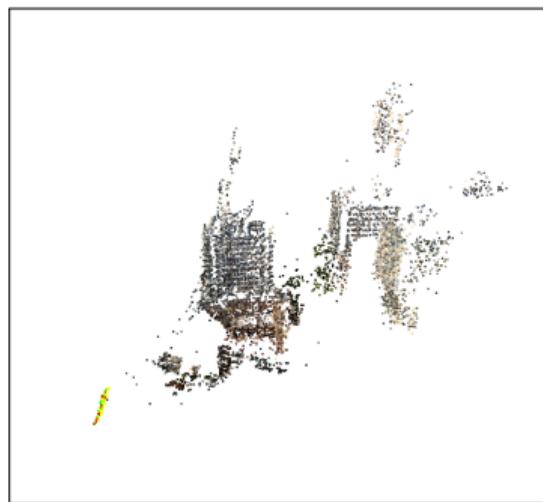
Modelo: Perspectiva com distorção radial (Harris).

Referência: *Bundler*.



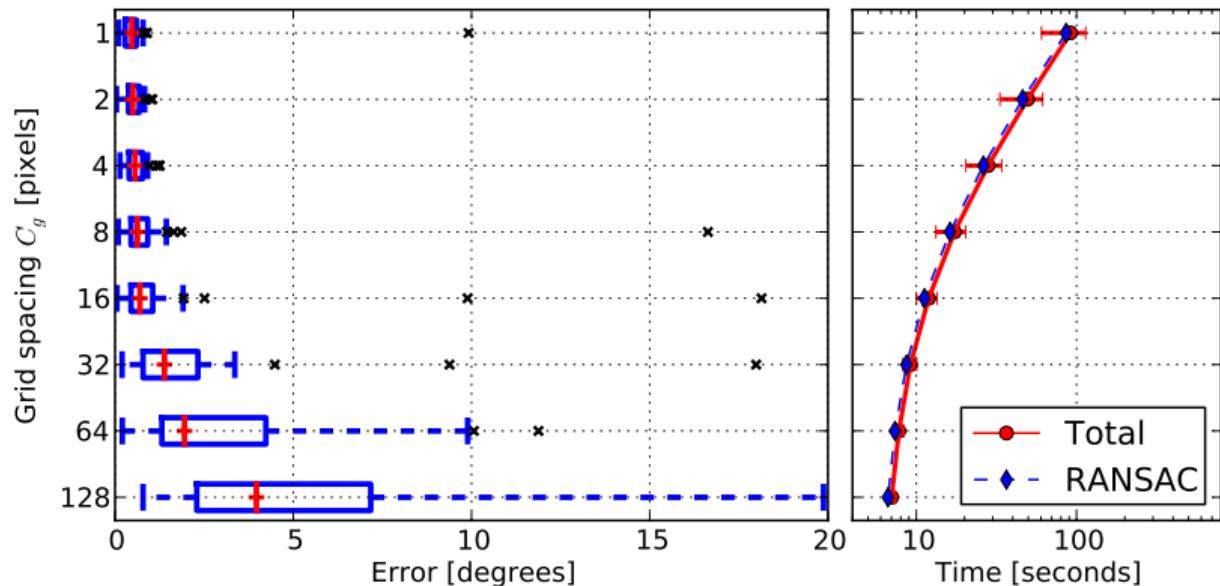
O *Bundler* (Snavely et al. [2006]) foi utilizado para obter as orientações de referência e os parâmetros intrínsecos.

Método multi-ocular baseado em pontos.



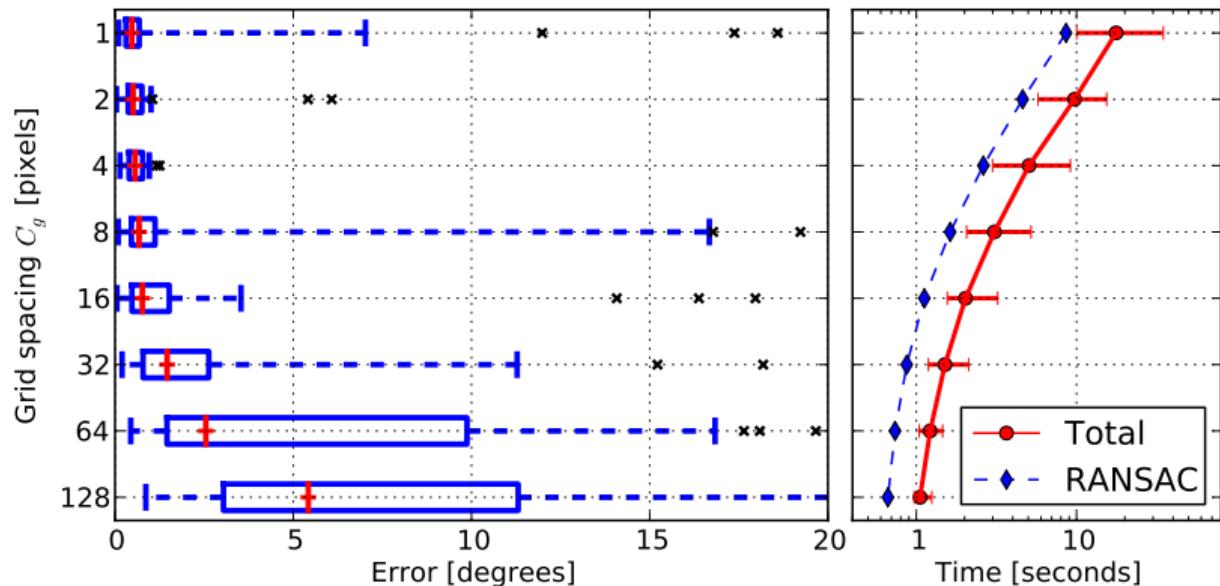
ApaSt

Corisco performance on ApaSt, $C_r = 10000$ RANSAC iterations
 Error distribution Process duration



ApaSt

Corisco performance on ApaSt, $C_r = 1000$ RANSAC iterations
Error distribution Process duration

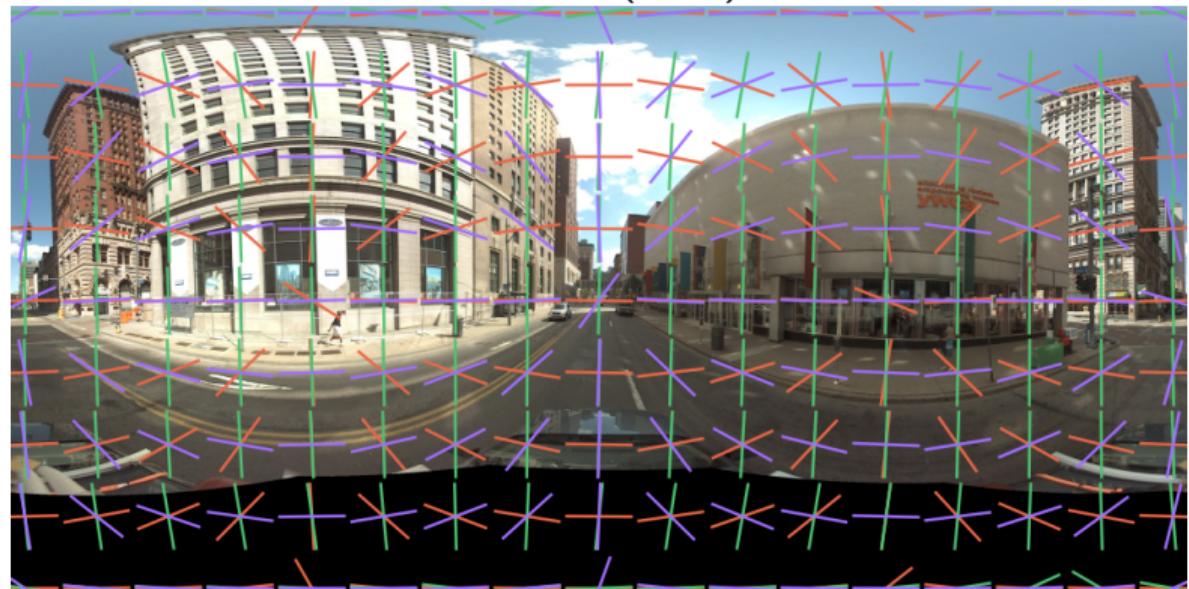


StreetView

Imagens: 250 imagens de um ambiente urbano.

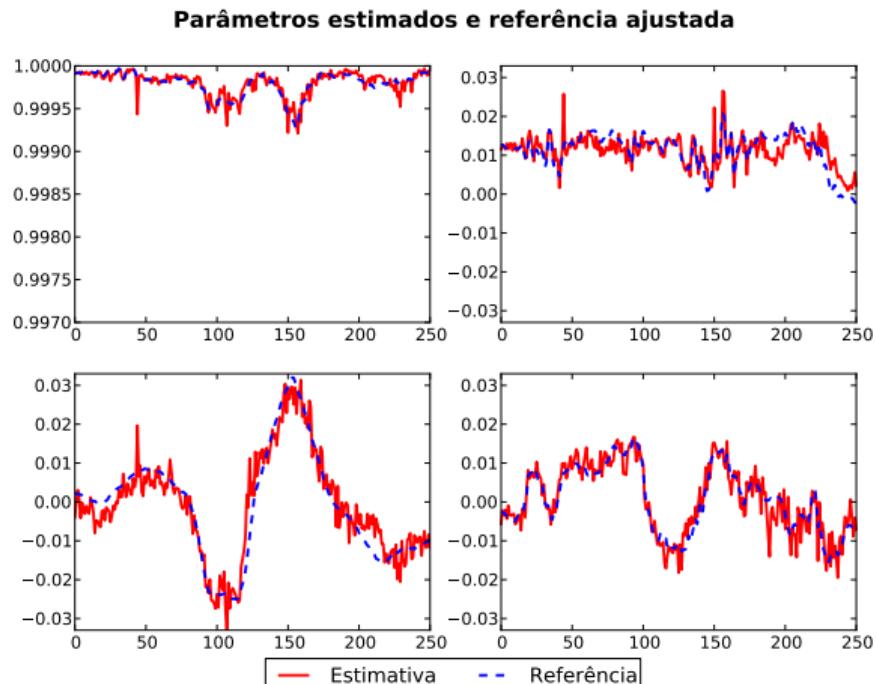
Modelo: Projeção equiretangular.

Referência: Sensores físicos (IMU).



StreetView

Erro $\simeq 1^\circ$, tempo $\simeq 17\text{s}$.



Conclusão

O *Corisco* constitui um método com grande potencial para aplicação imediata.

- Implementação relativamente simples.
- Grande robustez (distorções, RANSAC).
- Bom desempenho.

Demonstrou-se as vantagens do uso da máscara em forma de grade, e da M-estimação.

Também foi demonstrado como utilizar o FilterSQP para trabalhar naturalmente com quaternions.

Trabalhos futuros

- Controle automático dos parâmetros.
- Reconstrução multi-ocular e monocular.
- Utilizar filtros direcionais para medir o erro angular.
- Utilizar a máscara de grade em outros problemas.
- Estimar orientação e calibração, e extrair curvas em um processo unificado. (MRF?)

Fim

Obrigado!



Financiamento

Obrigado à Capes, CNPq e USP pelo suporte financeiro.

Publicações

- *Speeding up probabilistic inference of camera orientation by function approximation and grid masking*, Werneck and Costa [2011]
- *Mapping with monocular vision in two dimensions*, Werneck and Costa [2010]
- *Medição de distância e altura de bordas horizontais com visão monocular linear para robôs móveis*, Werneck et al. [2009]

Referências Bibliográficas

- B. Caprile and V. Torre. Using vanishing points for camera calibration. *International Journal of Computer Vision*, 4(2):127–139, March 1990. ISSN 0920-5691. doi: 10.1007/BF00127813. URL <http://www.springerlink.com/content/k75077108473tm15/>.
- R Cipolla, T Drummond, and D Robertson. Camera calibration from vanishing points in images of architectural scenes. In *British Machine Vision Conference*, volume 2, pages 382–391, Nottingham, England, 1999. BMVA.
- James M. Coughlan and A. L. Yuille. Manhattan World: Orientation and outlier detection by Bayesian inference. *Neural Computation*, 15(5): 1063—1088, March 2003. URL <http://www.mitpressjournals.org/doi/abs/10.1162/089976603765202668>.
- Patrick Denis, James H Elder, and Francisco J Estrada. Efficient edge-based methods for estimating Manhattan frames in urban imagery. In *European Conference on Computer Vision*, pages 197–210, Marselha, França, 2008. Springer. doi: 10.1007/978-3-540-88688-4_15.
- Roger Fletcher and Sven Leyffer. Nonlinear programming without a penalty function. *Mathematical Programming*, 91(2):239–269, January 2002. ISSN 0025-5610. doi: 10.1007/s101070100244. URL <http://www.springerlink.com/content/qqj37x00y79ygd18/>.

Carsten Rother. A new approach to vanishing point detection in architectural environments. *Image and Vision Computing*, 20(9-10):647–655, 2002. doi: 10.1016/S0262-8856(02)00054-9. URL [http://dx.doi.org/10.1016/S0262-8856\(02\)00054-9](http://dx.doi.org/10.1016/S0262-8856(02)00054-9).

G. Schindler and F. Dellaert. Atlanta World: An expectation maximization framework for simultaneous low-level edge grouping and camera calibration in complex man-made environments. In *Conference on Computer Vision and Pattern Recognition*, pages 203–209, Washington, DC, USA, 2004. IEEE. ISBN 0-7695-2158-4. doi: 10.1109/CVPR.2004.1315033. URL http://ieeexplore.ieee.org/xpl/freeabs_all.jsp?arnumber=1315033.

Noah Snavely, Steven M. Seitz, and Richard Szeliski. Photo Tourism: Exploring photo collections in 3D. In *SIGGRAPH*, pages 835–846, Boston, MA, USA, 2006. ACM. URL <http://dl.acm.org/citation.cfm?id=1141964>.

Nicolau Leal Werneck and Anna Helena Reali Costa. Mapping with monocular vision in two dimensions. *International Journal of Natural Computing Research*, 1(4):56–65, 2010.

Nicolau Leal Werneck and Anna Helena Reali Costa. Speeding up probabilistic inference of camera orientation by function approximation and grid masking. In Gladimir Baranowski and Vaclav Skala, editors, *WSCG 2011 Communication Papers*, pages 127–134, Plzen, Czech Republic, 2011. UNION Agency. URL
<http://nic.hpavc.net/almoxarifado/nic-wscg2011.pdf>.

Nicolau Leal Werneck and Anna Helena Reali Costa. Corisco: Robust edgel-based orientation estimation for generic camera models. *Image and Vision Computing*, 12(31):969—981, 2013. URL
<http://nic.hpavc.net/almoxarifado/imavis2013-final.pdf>.

Nicolau Leal Werneck, Flávio Sales Truzzi, and Anna Helena Reali Costa. Medição de distância e altura de bordas horizontais com visão monocular linear para robôs móveis. In *Workshop de Visão Computacional*, São Paulo, SP, Brasil, 2009. Universidade Presbiteriana Mackenzie. URL http://nic.hpavc.net/almoxarifado/59760_WerneckTruzziCosta.pdf.

Teste de calibração

Estimação da distância focal com a mesma $F(\Psi)$.

Calibração baseada no Corisco - Função objetivo variando com a distância focal

