

# Odometria Visual

Paulo Lilles Jorge Drews Junior

Trabalho Prático – Visão Computacional

Universidade Federal de Minas Gerais - VeRLab

[paulol@dcc.ufmg.br](mailto:paulol@dcc.ufmg.br)

# O que é odometria?

- **Wikipedia**

A odometria é um dos métodos mais amplamente utilizados para **estimar a posição** de um robô.

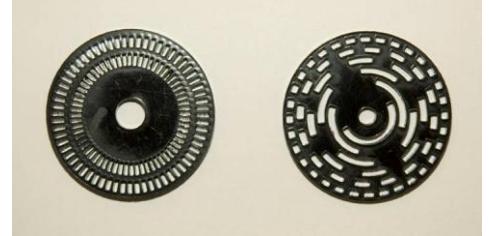
Sabe-se bem que a odometria proporciona uma **boa precisão em curto prazo**, é **barata de implantar**, e **permite taxas de amostragem muito altas**.

A idéia fundamental da odometria é a **integração de informação incremental do movimento** ao longo do tempo, o qual envolve uma inevitável **acumulação de erros**.

A acumulação de **erros** de orientação causa grandes erros na estimativa da posição, os quais vão **aumentando proporcionalmente com a distância percorrida** pelo robô.

Apesar destas limitações, muitos pesquisadores concordam que a odometria é uma parte importante do sistema de navegação de um robô, e que deve **usar-se com medidas do posicionamento absolutas** para proporcionar uma estimativa de posição mais confiável.

# Introdução



- Alternativa a Odometria Convencional
- Erros Acumulativos
- Visual Dependente do Ambiente
  - Dinâmico / Estático
  - Monótono
  - Estruturado / Não-Estruturado
- Processo de Odometria Visual
  - Determinação de *Features*
  - Correlação de *Features*
  - Estimativa de Movimento

# Metodologia

## Câmera

- Captura Imagem

Imagem

## Pré-Processamento

- Corrige Distorções do Sistema Óptico
- Calibração da Câmera

Imagem  
Corrigida

## Determinação de Características e Descritores

- SIFT
- SURF
- LESH

X,Y e Descritor  
dos Pontos

## Matching com Imagem Anterior

- Distância Euclidiana
- Remoção de falsos matches
- Matriz Fundamental – LMedS / RANSAC

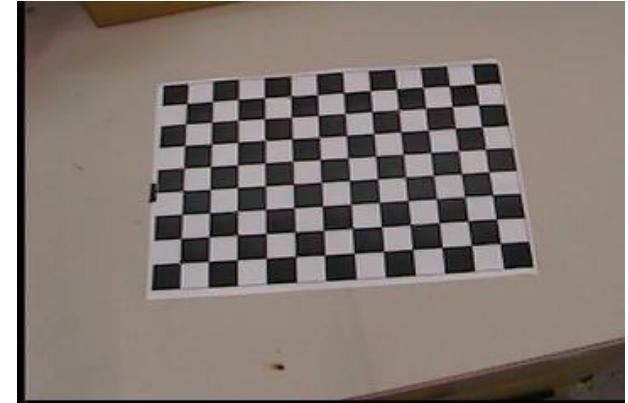
Par de Pontos  
Correlacionados

## Determinação do Movimento

- Restrição Epipolar – Matriz Essencial

# Calibração da Câmera e Correção das Distorções do Sistema Óptico

- Método de Zhang
  - Padrão Planar 2D
  - Biblioteca OpenCV
  - Parâmetros Intrínsecos



- Correção da Distorção Radial
  - LUT – Lookup Table

$$x' = x + x \cdot [k_1 \cdot (x^2 + y^2) + k_2 \cdot (x^2 + y^2)^2]$$
$$y' = y + y \cdot [k_1 \cdot (x^2 + y^2) + k_2 \cdot (x^2 + y^2)^2]$$

# Determinação de *Features*

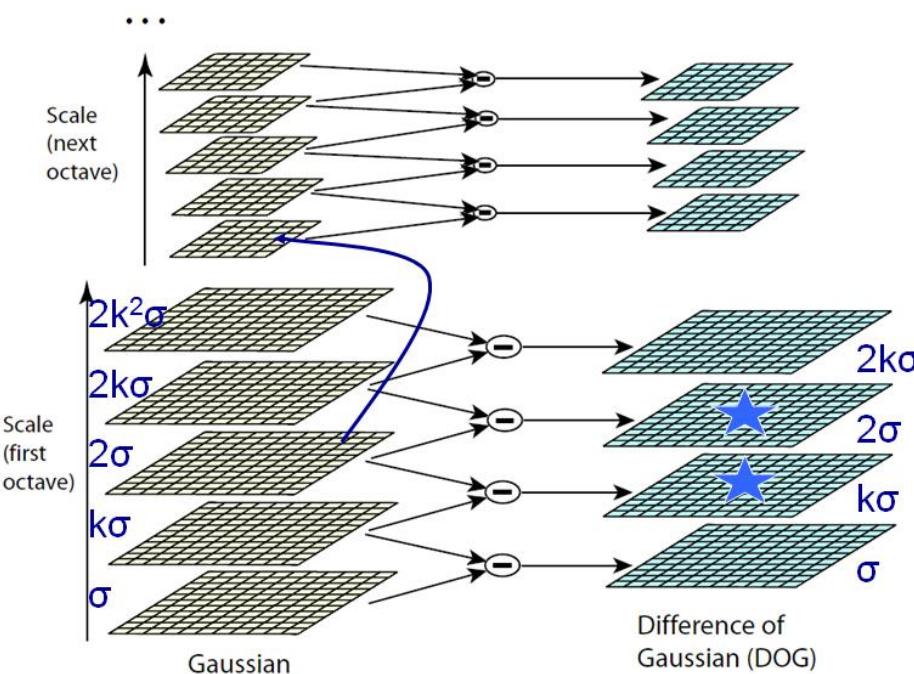
## Utilizando Algoritmos Descritores

- SIFT
  - Detecção de extremos no espaço de escala
  - Localização de pontos-chaves
  - Definição de orientação
  - Descriptor dos pontos-chaves
- SURF
  - Criação da Integral da Imagem
  - Determinação de Pontos de Interesse através da *Fast-Hessian*
  - Descriptor dos pontos-chaves
    - USURF
    - SURF
    - SURF128

# SIFT

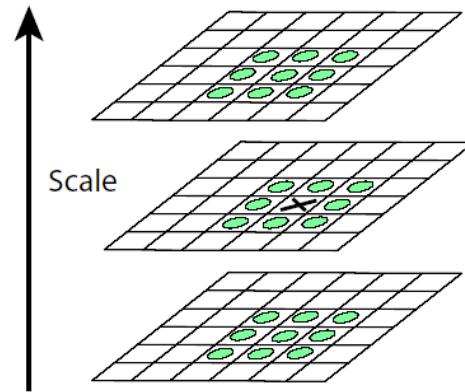
- Detecção de extremos no espaço de escala

$$\begin{aligned} D(x, y, \sigma) &= (G(x, y, k\sigma) - G(x, y, \sigma)) * I(x, y) \\ &= L(x, y, k\sigma) - L(x, y, \sigma) \end{aligned}$$



# SIFT

- Localização de pontos-chaves



- Definição de orientação

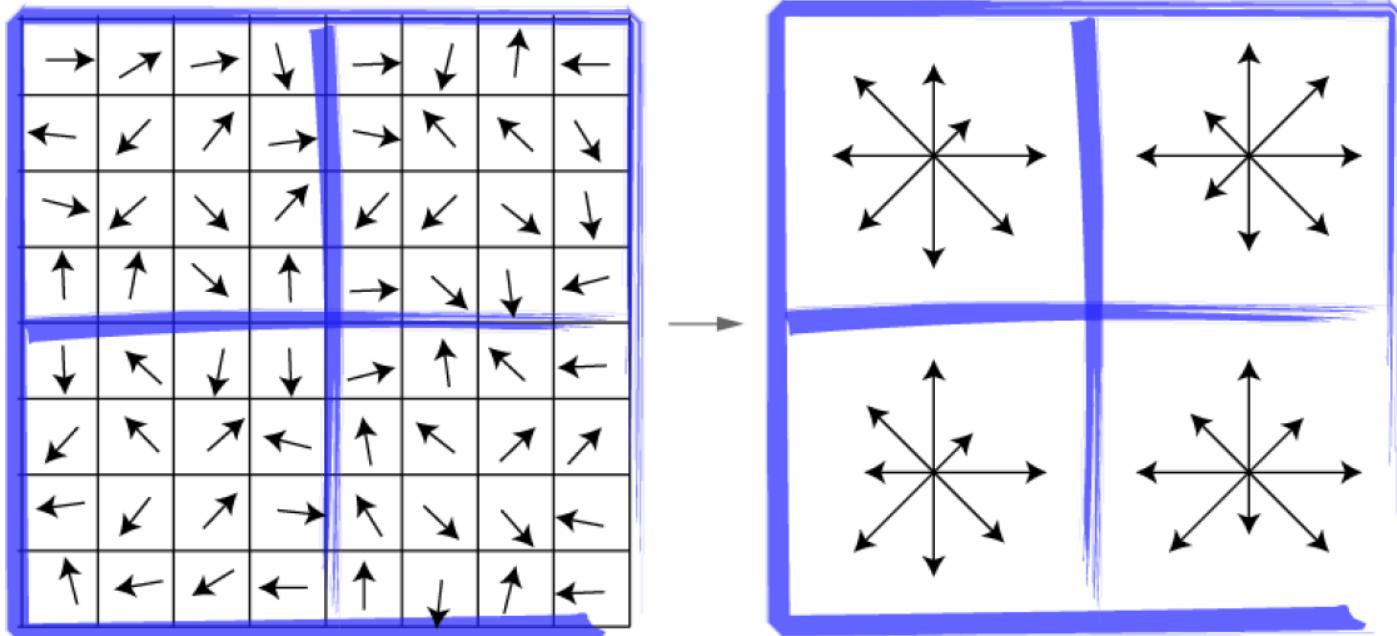
$$m(x, y) = (L(x + 1, y) - L(x - 1, y))^2 + \\ (L(x, y + 1) - L(x, y - 1))^2)^{1/2}$$

$$\theta(x, y) = \tan^{-1}((L(x, y + 1) - L(x, y - 1)) \\ / (L(x + 1, y) - L(x - 1, y)))$$

# SIFT

- Descritor de Pontos-Chaves

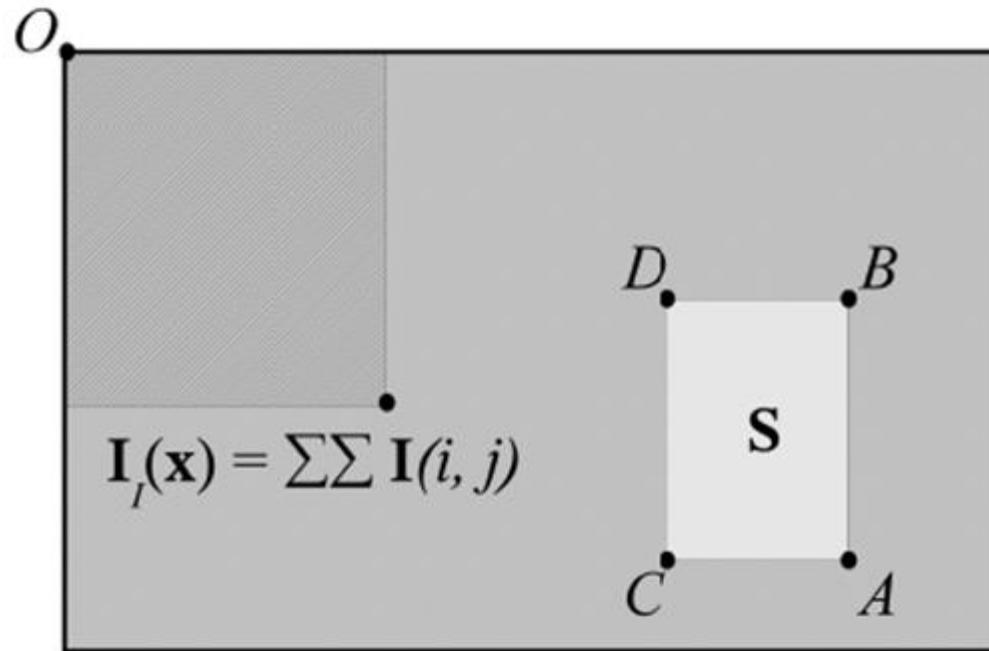
- Regiões  $n \times n$  com  $k \times k$  pixels ( $n=4, k=4$ )
- ‘r’ orientações por região ( $r = 8$ )
- Descritores de 128 posições –  $n \times n \times r$



# SURF

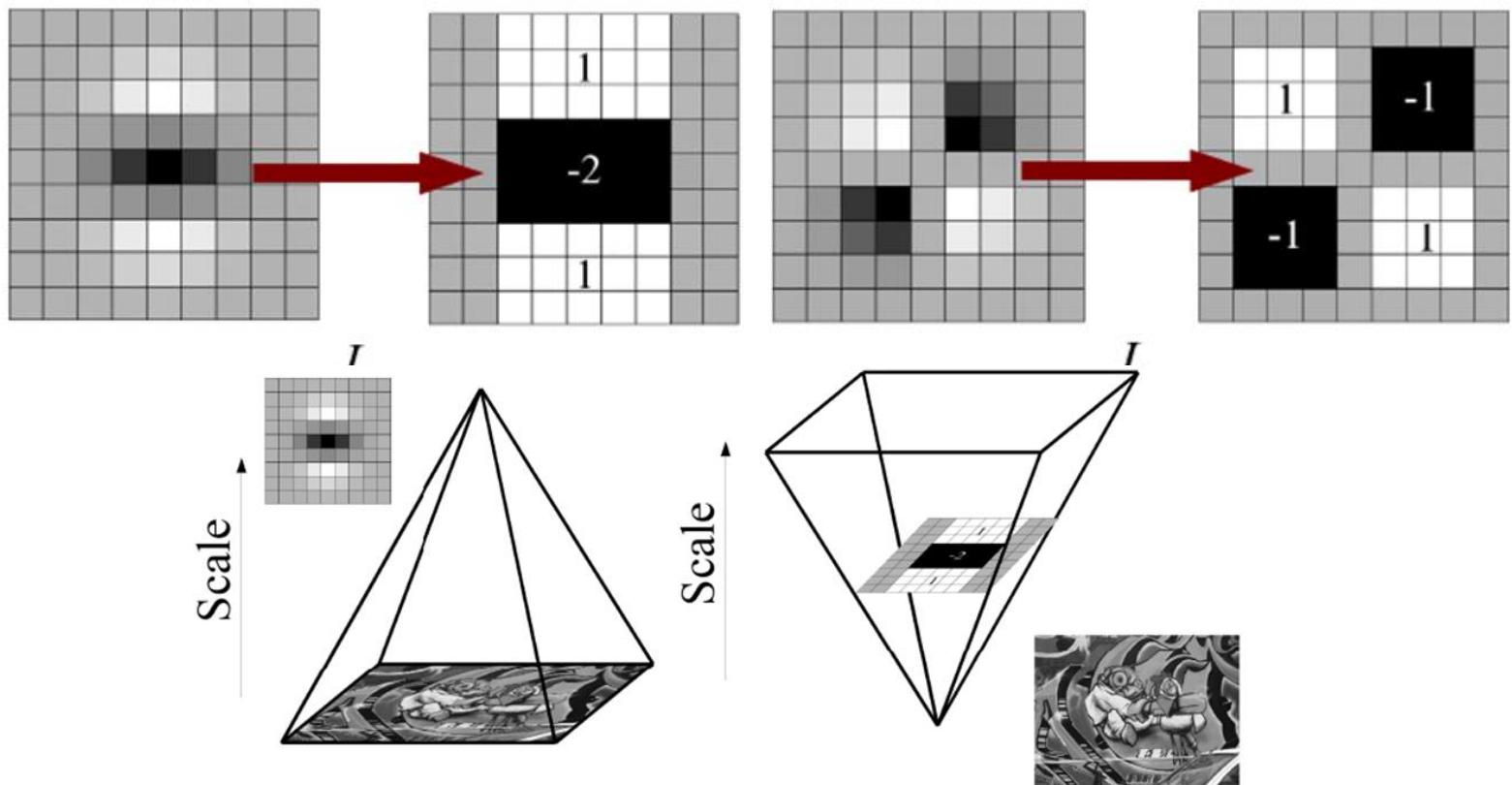
- Integral da Imagem

$$I_{\sum}(X) = \sum_{i=1}^{i \leq x} \sum_{j=1}^{j \leq y} I(i, j)$$



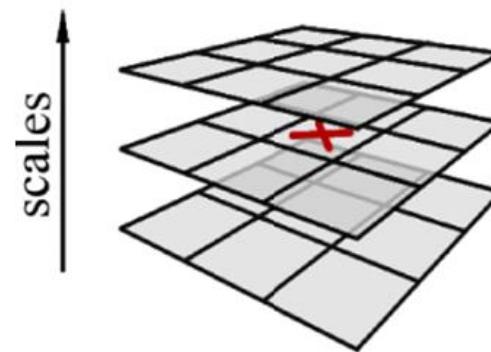
# SURF

- Determinação de Pontos de Interesse através da *Fast-Hessian*



# SURF

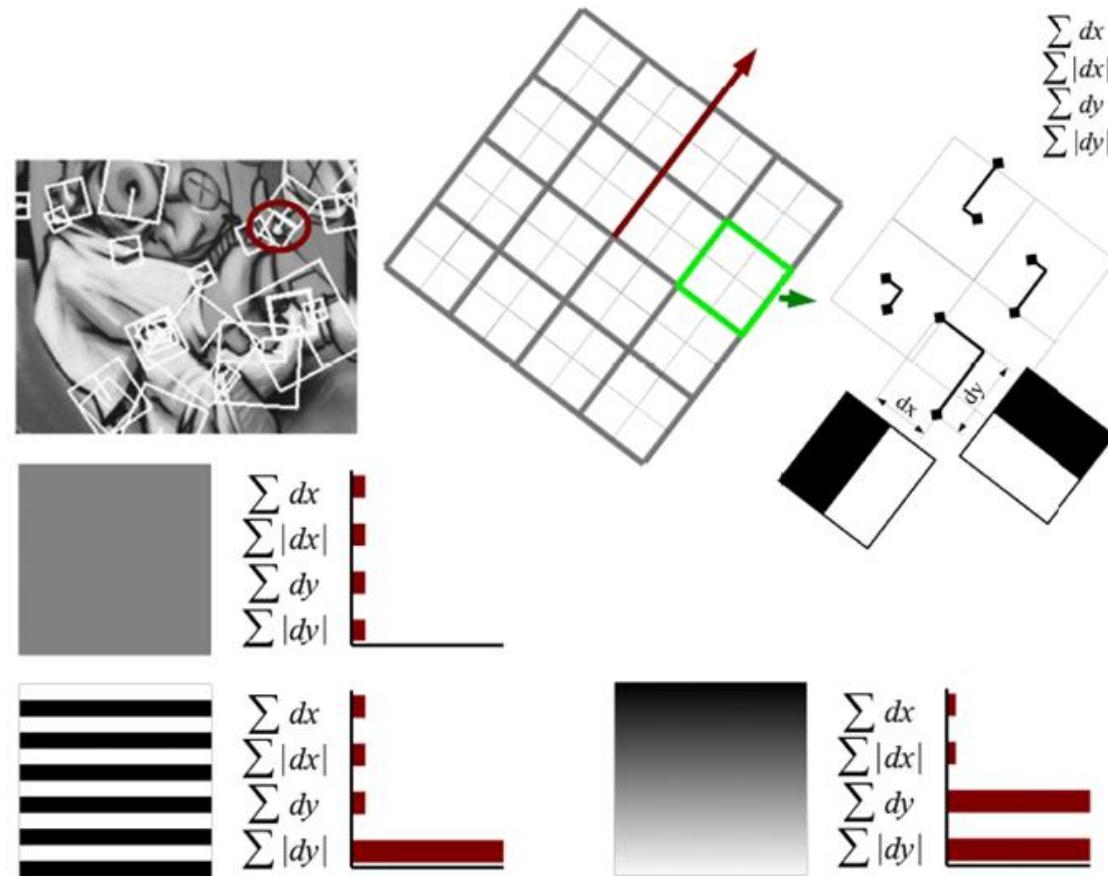
- Máximos na Fast-Hessian



- Descritor dos pontos-chaves
  - USURF – Variante a Rotação
  - SURF – Descritor Padrão
  - SURF128 – Descritor Aumentado

# SURF

- Descritor dos pontos-chaves
  - Haar-Wavelet em região  $4 \times 4$



# Correlação entre Descritores de Features

- SIFT
  - Variação KD-TREE – BBF (Best-Bin-First)
    - Busca Heurística
    - Princípio da Menor Distância Euclidiana
    - Razão com o Segundo Menor Vizinho
- SURF
  - Busca Exaustiva – Menor Dimensionalidade
    - Sinal do Laplaciano
    - Razão com o Segundo Menor Vizinho

# Remoção Robusta de Outliers utilizando RANSAC

- Ransac através da Matriz Fundamental
  - Selecionar randomicamente um subconjunto de oito pontos de correlacionados, retirados do conjunto total de pontos correlacionados.
  - Para cada subconjunto, indexado por  $j$ , calcular a matriz fundamental  $F_j$  através do algoritmo dos oito pontos.
  - Para cada matriz  $F_j$  computada, determinar o número de pontos com distância até a linha epipolar, ou residual, menor que um limiar.
  - Selecionar a matriz  $F$  que apresenta o maior número de pontos com residual inferior ao máximo definido.
  - Recalcular a matriz  $F$  considerando todos os pontos inliers.

$$\tilde{m}^T F \tilde{m}' = 0$$

# Remoção Robusta de *Outliers* utilizando LMedS

- Least Median Square

- Semelhante ao RANSAC
  - Resíduo Calculado Pela equação Abaixo

$$r_{ji}^2 = d(\tilde{m}_i', F_j \tilde{m}_i)^2 + d(\tilde{m}_i, F_j^T \tilde{m}_i')^2$$

- Através do Resíduo para cada pontos, selecionando matriz com menor resíduo da do valor mediano.

# Determinação do Movimento Com Matriz Essencial

- Matriz Essencial a partir da Fundamental e Parâmetros Intrínsecos

$$E = KFK^T$$

- Rotação e Translação a partir da Essencial

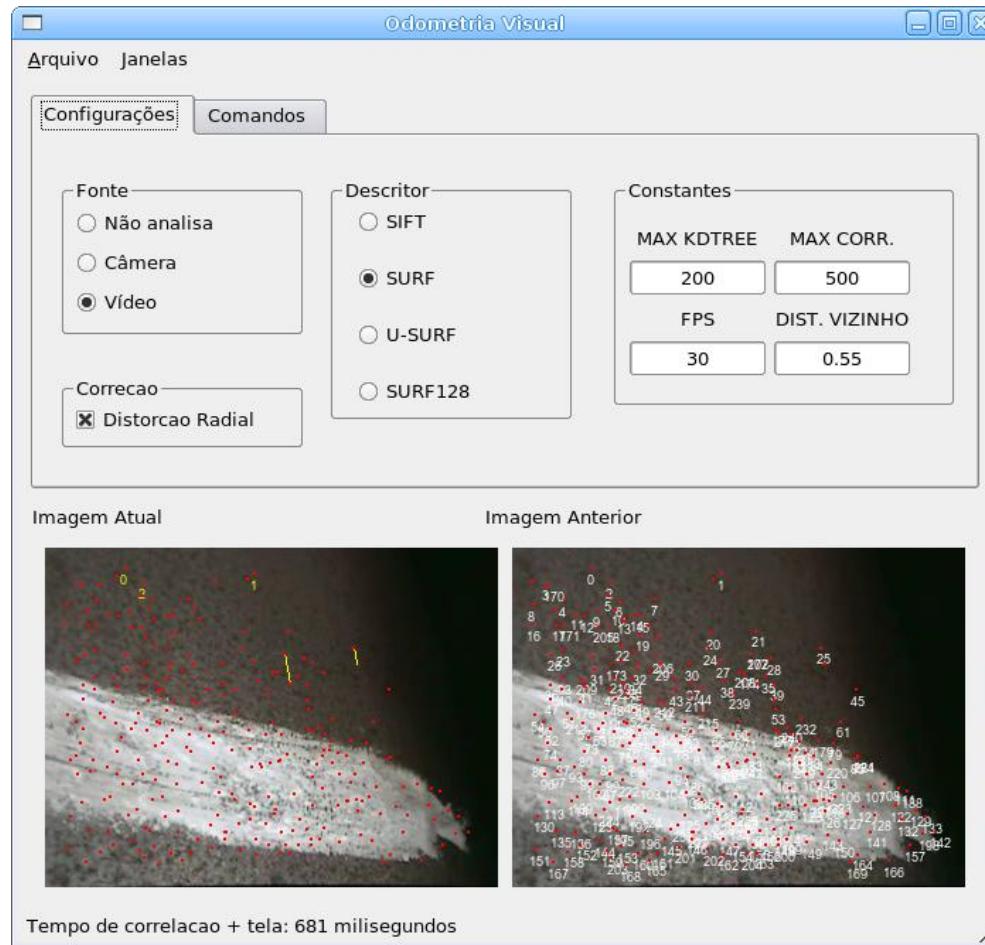
$$E = UDV^T$$

$$R^1 = U \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} V^T \quad T = V \begin{bmatrix} 0 & 1 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} V^T$$

$$R^2 = U \begin{bmatrix} 0 & 1 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} V^T \quad T = \begin{bmatrix} 0 & -z & y \\ z & 0 & -x \\ -y & x & 0 \end{bmatrix}$$

# Resultados

- # • Plataforma Implementada em C++/Qt



# Plataforma de Testes

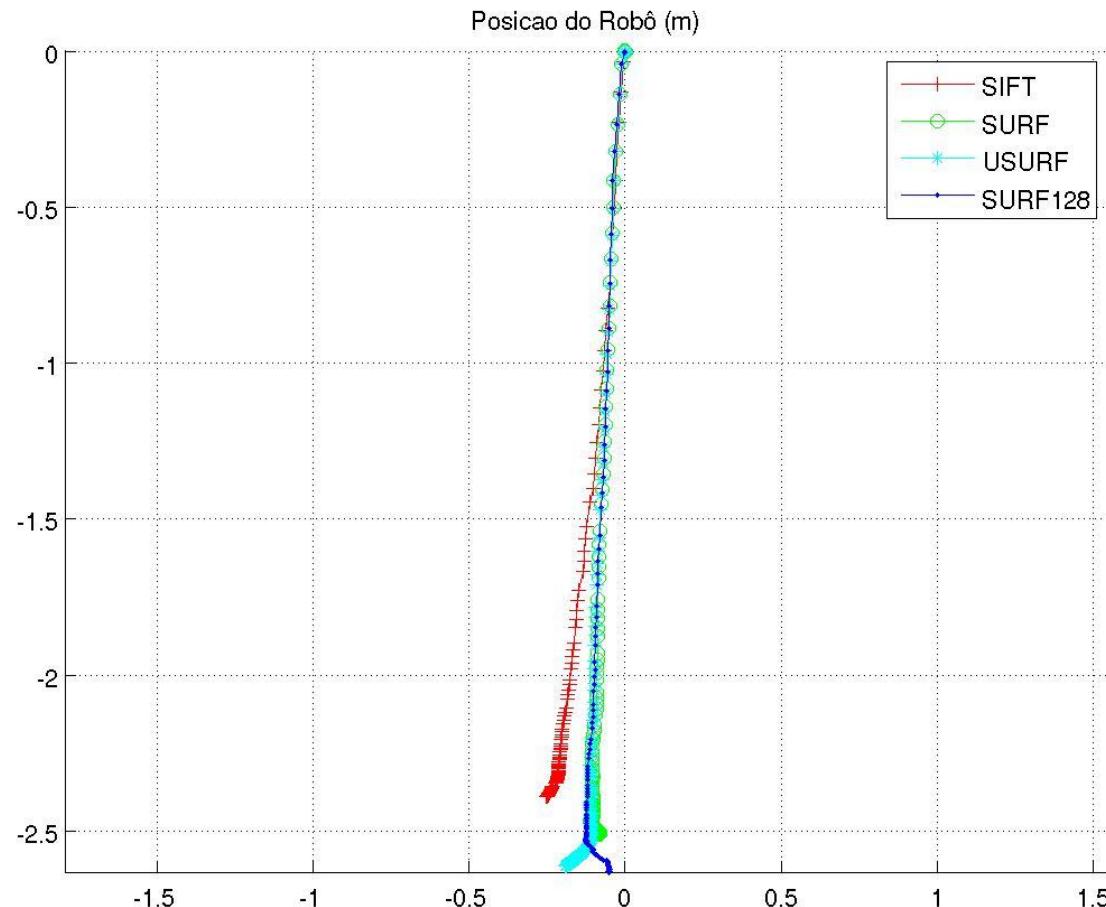


# Resultado da Calibração

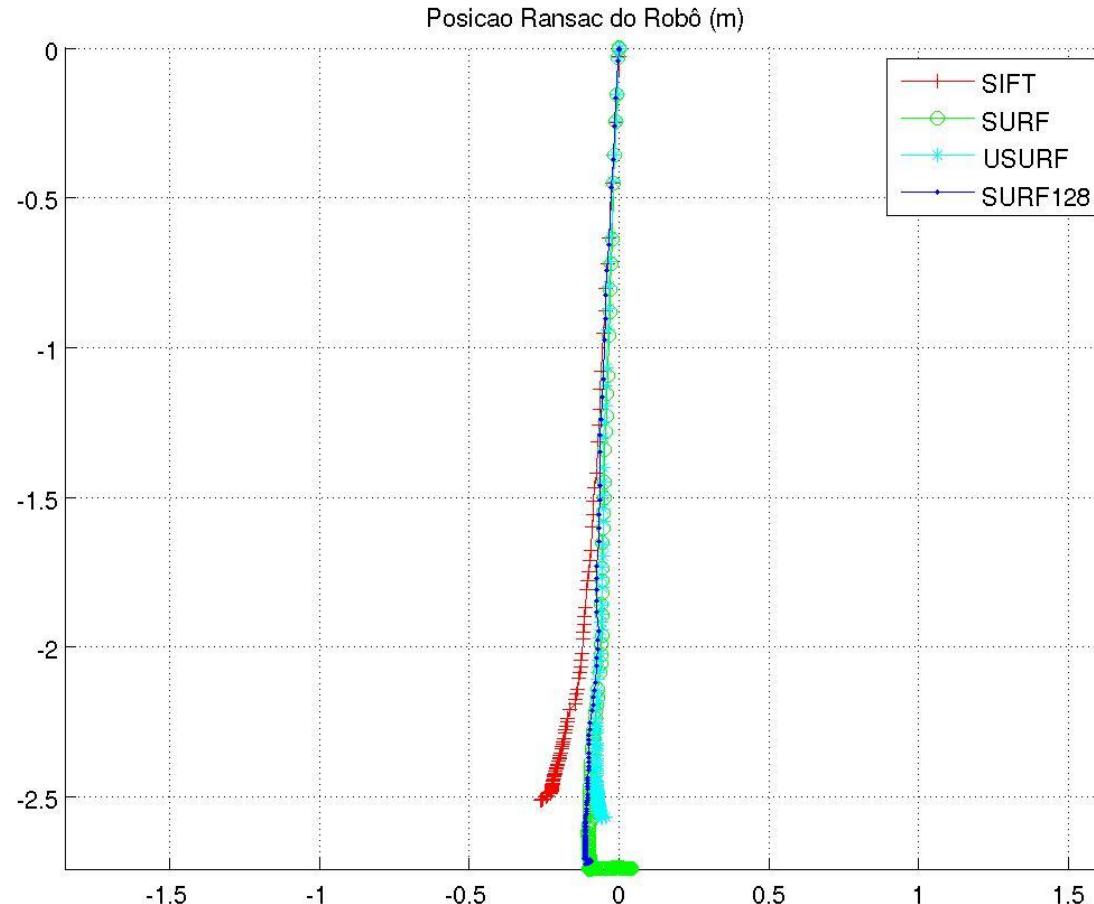
Modelo da Câmera	$f_1$	$f_2$	$u_0$	$v_0$	$k_1$	$k_2$
Samsung SCD-363	458.7	418.2	195.4	102.4	-0.18	0.22



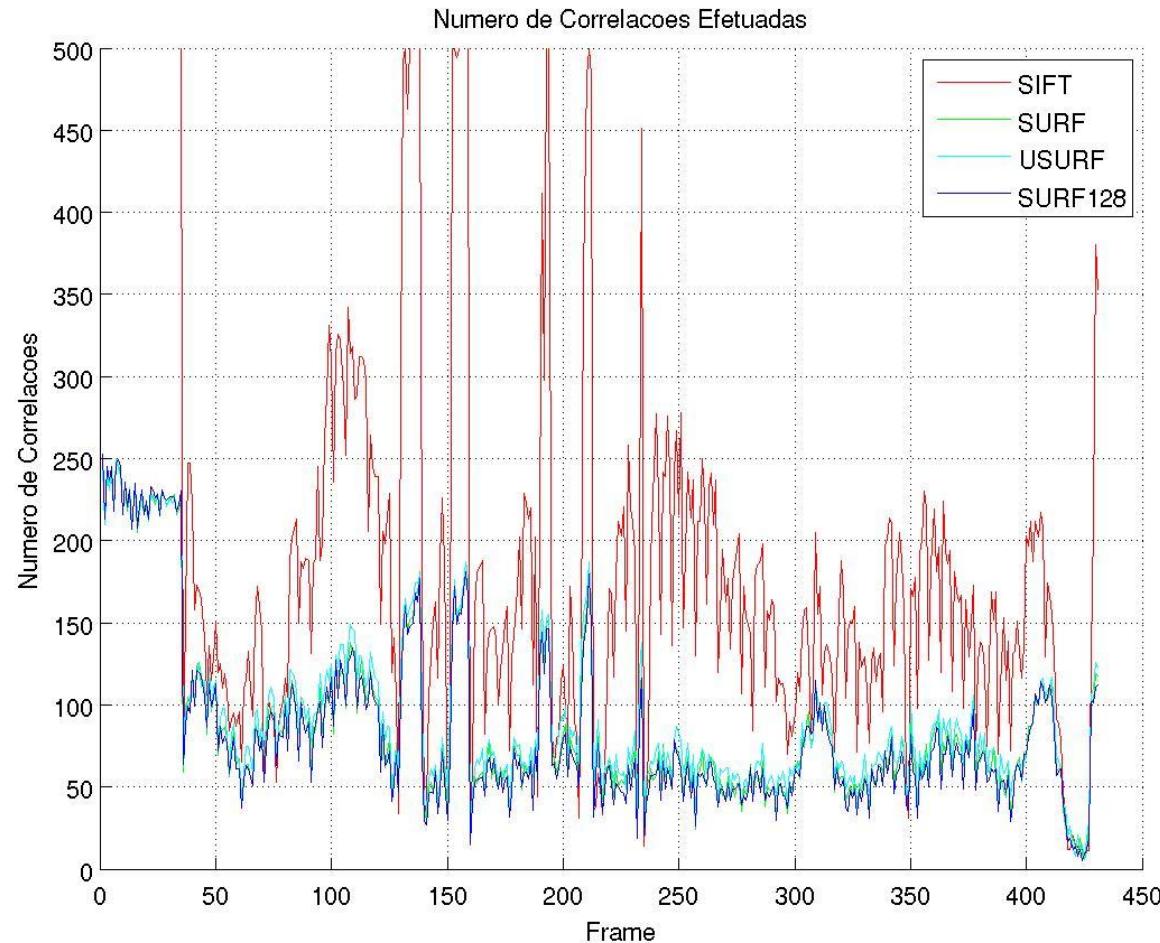
# Resultados da Odometria



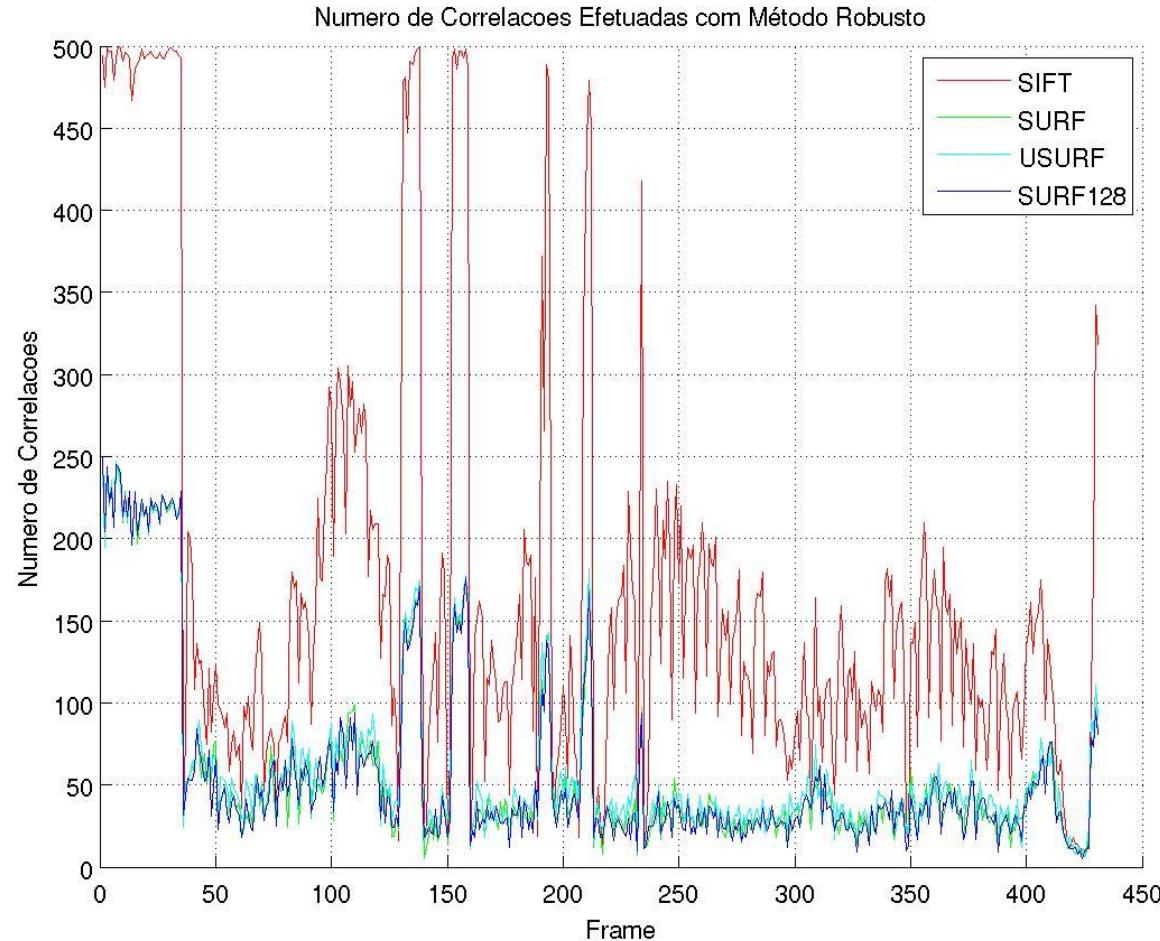
# Resultados da Odometria Robusta



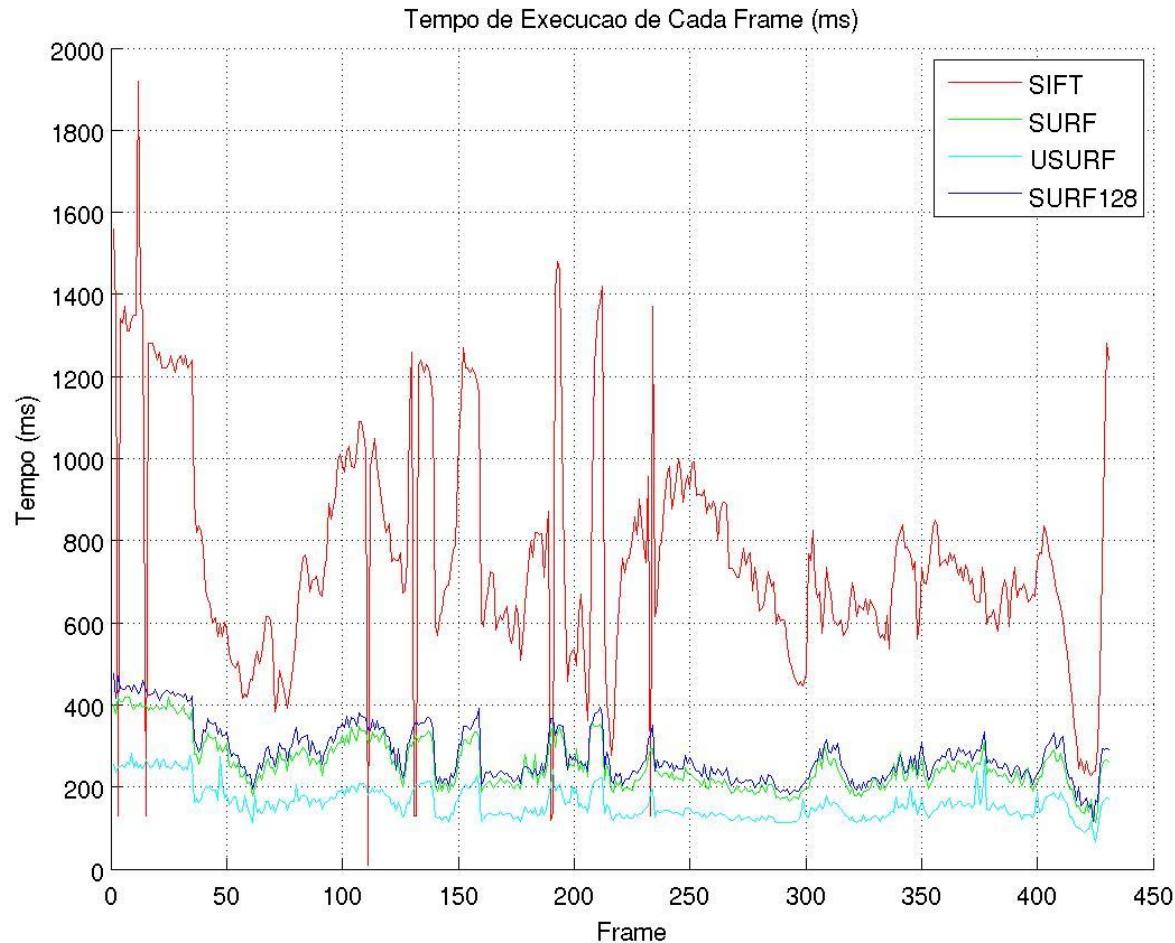
# Correlações Efetuadas



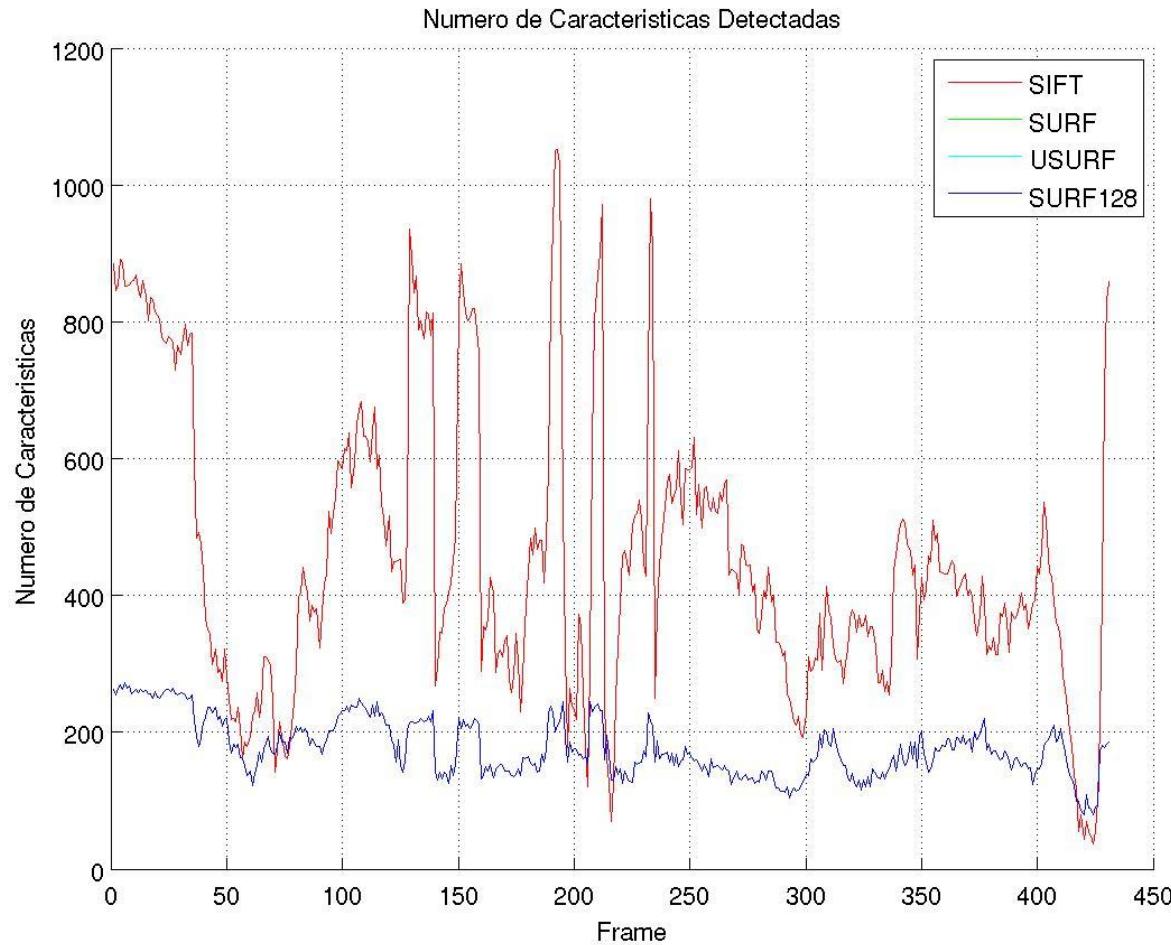
# Correlações Verdadeiras Efetuadas



# Tempo de Execução



# Número de *Features* Detectadas



# Conclusão

- Odometria Visual apresentou bons resultados com movimento translacional
- SIFT apresentou melhores resultado porém bem mais lento
- SURF apresentou ótimo desempenho e precisão
- Remoção de *Outliers* nem sempre a melhor solução
- Fusão Sensorial e Kalman pode melhorar resultados

# Vídeos da Câmera do Robô



# Vídeo do Robô



# Dúvidas?

Paulo Lilles Jorge Drews Junior  
Universidade Federal de Minas Gerais - VeRLab  
[paulol@dcc.ufmg.br](mailto:paulol@dcc.ufmg.br)