

Reconhecimento de moedas de Real em imagens digitais utilizando Hough, SIFT e SVM

Luiz Fernando Machado Silva

Introdução

- Reconhecimento de Padrões
 - Detectar, reconhecer e classificar objetos
 - Não existem algoritmos genéricos e universais
 - Soluções específicas para problemas específicos
- Reconhecimento de Moedas de Real
 - Tema abordado pelo aluno Jory, N. M. (2011) (IFSC)
 - Classificação por comparações de cor e diâmetro
 - Algoritmo com taxa de acerto de aprox. 50%
 - Desenvolvido em MATLAB (baixo desempenho)

Introdução

- Aplicações
 - Inclusão de deficientes visuais
 - Melhorias em acessibilidade (output sonoro)
 - Detecção de moedas falsas em máquinas
 - Automatização na contagem de moedas

Objetivo

- Objetivo geral
 - Algoritmo capaz de reconhecer e classificar as moedas da segunda família de Real
- Objetivos específicos
 - Definir método para aquisição de imagens
 - Melhorar qualidade das imagens usando filtros
 - Detectar objeto circular na imagem
 - Extrair características do objeto circular
 - Classificar objeto circular com um valor de moeda

Metodologia

- Revisão Bibliográfica
 - Processamento de Imagens
 - Reconhecimento de Padrões
 - Visão Computacional
- Bibliotecas de Processamento de Imagens
 - OpenCV
 - Bial
- Obtenção de imagens digitais de moedas
- Desenvolvimento do algoritmo
- Testes de assertividade

Revisão Bibliográfica

- Processamento Digital de Imagens
 - Imagens Digitais
 - Filtragem
 - Segmentação de Regiões
 - Extração de características
 - Reconhecimento de objetos
- Bibliotecas de Processamento de Imagens
 - OpenCV
 - BIAL
- Moedas de Real
 - Morfologia e características

Imagen

- Imagen
 - Definida como função $f(x,y)$ finita e discreta
 - (x,y) denota as coordenadas do pixel no 2D
 - f denota a intensidade (nível de cinza) do pixel

$$f(x,y) = \begin{vmatrix} f(0,0) & f(0,1) & \dots & f(0,N-1) \\ f(1,0) & f(1,1) & \dots & f(1,N-1) \\ \dots & \dots & & \dots \\ f(M-1,0) & f(M-1,1) & \dots & f(M-1,N-1) \end{vmatrix}$$

Imagen

- Exemplo de figura monocromática de 8-bits
 - 00000000 (0) – preto
 - 11111111 (255) – branco
 - Valores intermediários – tons de cinza



Processamento de Imagens

- Processamento de Imagens
 - Trabalham diretamente em imagens
 - Imagens como entrada e saída
- Visão Computacional
 - Emulam aspectos da visão humana
 - Imagens como entrada, características como saída
- Análise de Imagens
 - Compreendem objetos extraídos de imagens
 - Características como entrada, “sentido” como saída

Processamento de Imagens

- Níveis dos algoritmos de PI
 - Nível baixo
 - Operações primitivas
 - Pré-processamento
 - Filtragem de ruídos
 - Nível médio
 - Segmentação de imagens
 - Extração de características
 - Nível alto
 - Interpretação de imagens
 - Reconhecimento de objetos

Filtragem

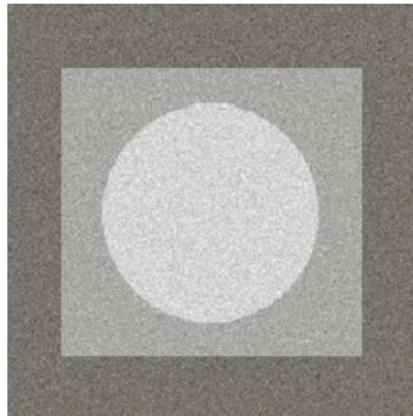
- Filtragem
 - Técnicas utilizadas para melhorar qualidade
 - Ajuste de contraste entre elementos
 - Remoção de ruídos
 - Acentuação de características
- Domínios de atuação
 - Espacial
 - Diretamente nos pixels do plano da imagem
 - Menor custo computacional
 - Frequência
 - Atuação na transformada de Fourier da imagem

Filtragem de ruído

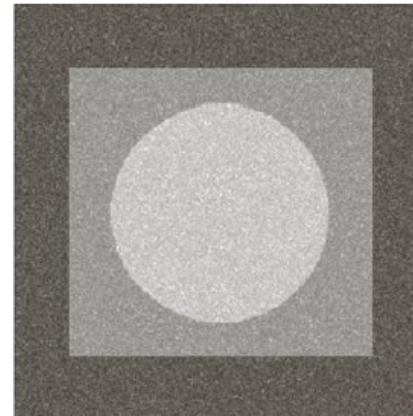
- Ruído

- Ruído é a variação indesejada de intensidade nos pixels que compõe uma imagem
- Fenômeno aleatório descrito por funções de densidade de probabilidade
 - PDF (Probability Density Function)
- Causas
 - qualidade dos equipamentos de captura
 - variações na iluminação
 - variações na temperatura
 - variações eletromagnéticas

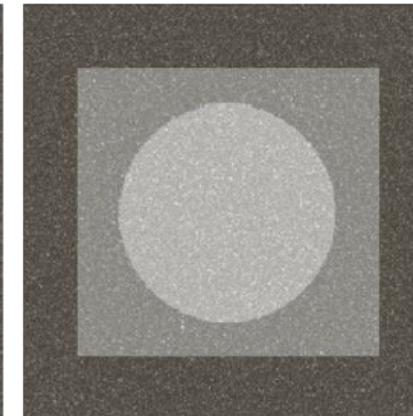
Filtragem de ruído



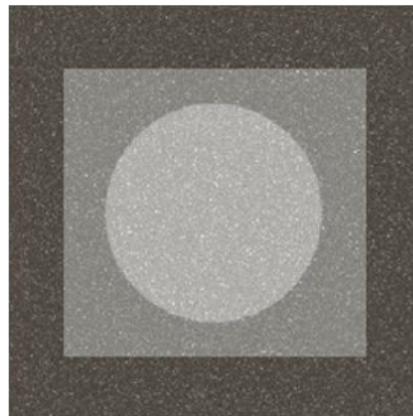
Gaußiano



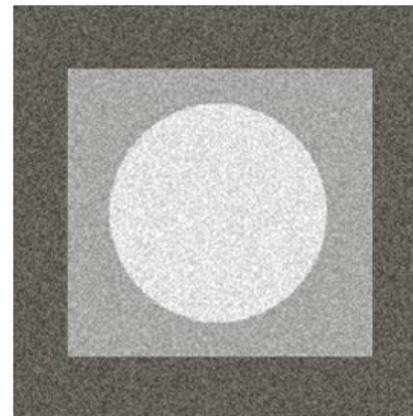
Rayleigh



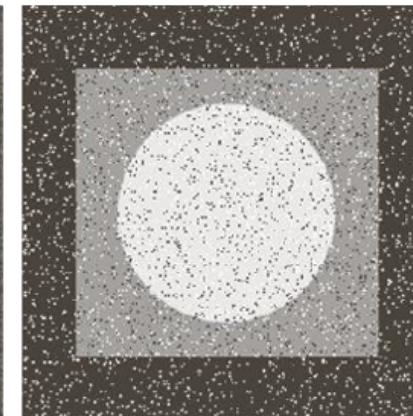
Gama



Exponencial



Uniforme



Sal e pimenta

Segmentação de regiões

- Segmentação de regiões
 - Algoritmos que recebem imagens como entrada e retornam informações sobre regiões destas imagens
- Métodos
 - Método de Canny
 - Transformada de Hough

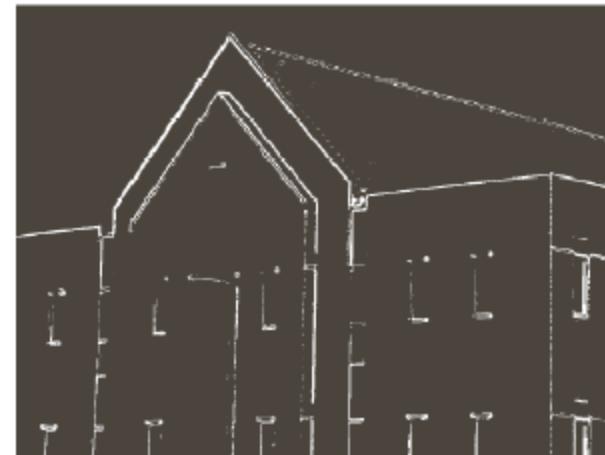
Segmentação de regiões

- Método de Canny
 - Algoritmo de detecção de bordas considerado ideal
 - Baixa taxa de erro
 - Pontos de borda bem localizados
 - Resposta única para cada ponto de borda
 - Maior complexidade, porém obtém melhores resultados, comparado a outros métodos de detecção de borda
 - Saída: imagem binária
 - pixels de borda
 - pixels de não-borda

Segmentação de regiões



Original



Gradiente Limiarizado



Algoritmo Mar-Hildreth



Algoritmo de Canny

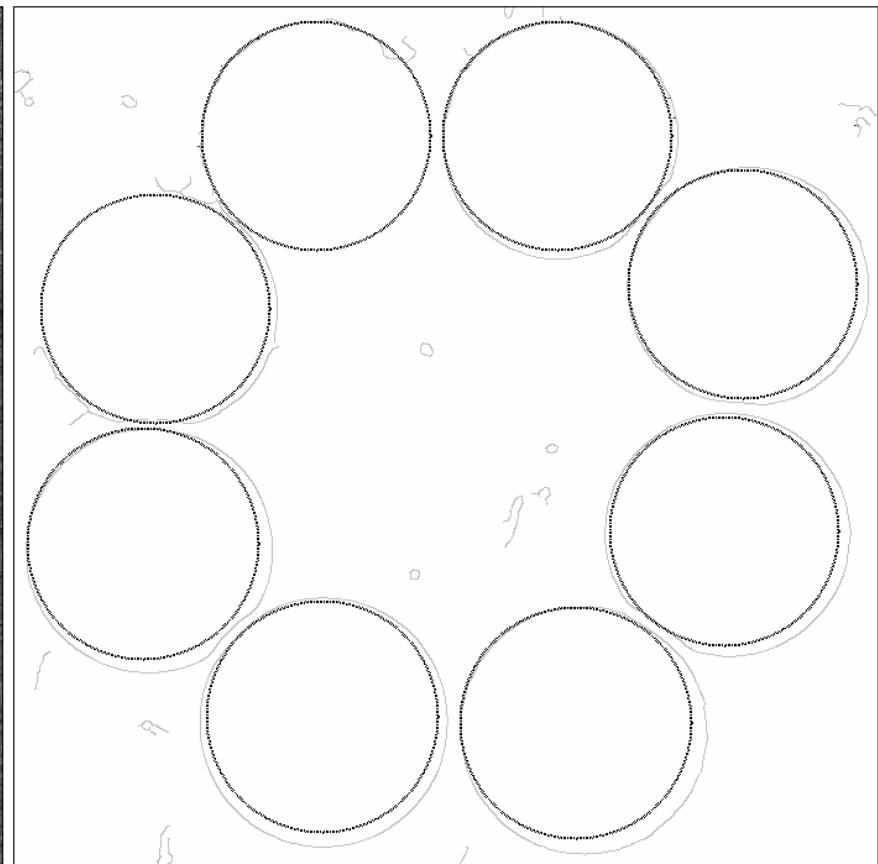
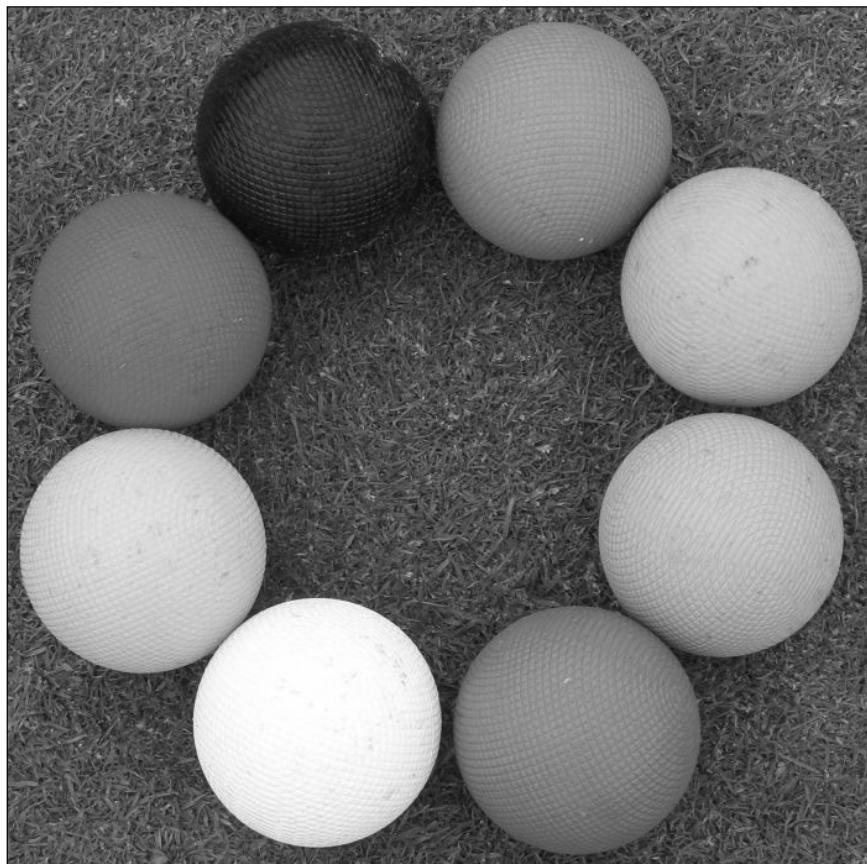
Segmentação de regiões

- Transformada de Hough
 - Método de detecção de figuras geométricas que podem ser descritas por equações parametrizadas
 - Círculos
 - Elipses
 - Retas
 - Aplicada quando há a necessidade de corrigir descontinuidades em pixels de borda
 - Recebe como entrada pixels de borda previamente detectados e tenta “encaixá-los” em uma equação parametrizada conhecida

Segmentação de regiões

- Transformada de Hough para círculos
 - Encontrando o centro (x, y)
 - Ponto de encontro entre as retas normais à tangente de cada ponto de borda detectado
 - Encontrando o raio r
 - Distância média entre o centro calculado no passo anterior e cada ponto de borda detectado

Segmentação de regiões



Extração de características

- Extração de características
 - Segmentação de imagens delimita área de interesse
 - Para compreender a região segmentada, é preciso extrair as características que a descrevem
- Tipos de características
 - Características morfológicas
 - Análise de pixels de borda
 - Características regionais
 - Análise de pixels internos

Reconhecimento de Objetos

- Reconhecimento de Objetos
 - Métodos usados para classificar e dar sentido a regiões segmentadas de uma imagem
 - Conceito de associar uma amostra desconhecida à uma classe de um grupo de classes conhecidas

Bibliotecas de PI

- Bibliotecas de Processamento de Imagens
 - Bibliotecas livres, de código-aberto com funcionalidade voltada para as áreas de PI e VC
 - OpenCV (Open Source Computer Vision Library)
 - Intel / Willow Garage / Itseez
 - C/C++, alta performance, tempo real
 - BIAL (Biomedical Image Analysis Library)
 - ICT-UNIFESP
 - Especialmente projetada para análise de imagens biomédicas

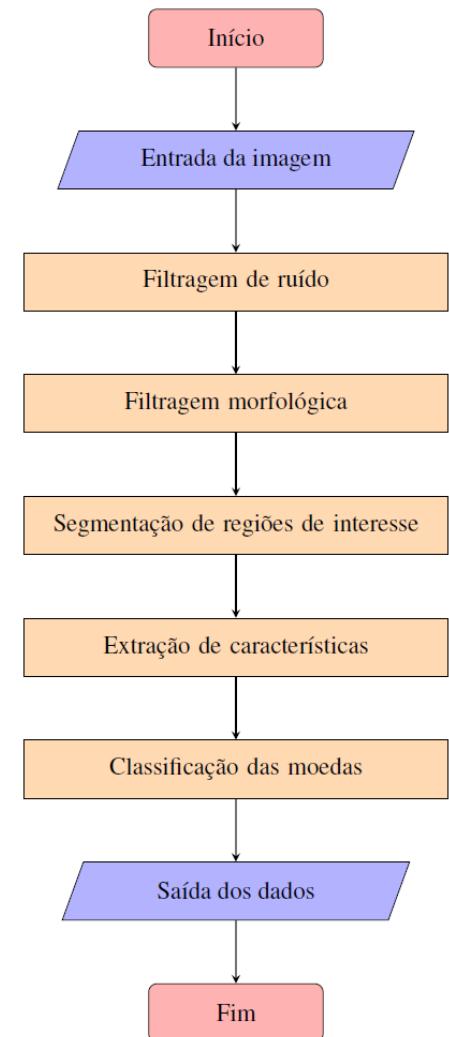
Moedas do Real

- História
 - Segunda família de moedas, lançadas em 98
 - Diâmetro variando entre 17mm e 27mm
 - Anverso: valor da moeda e referência à Bandeira
 - Reverso: busto de figuras históricas



Desenvolvimento

- Desenvolvimento do trabalho
 - a. Organização de arquivos
 - b. Sistema de cores
 - c. Filtragem de ruído
 - d. Detecção de áreas de interesse
 - e. Extração de descriptores
 - f. Dicionário de características
 - g. Treinamento SVM
 - h. Classificação



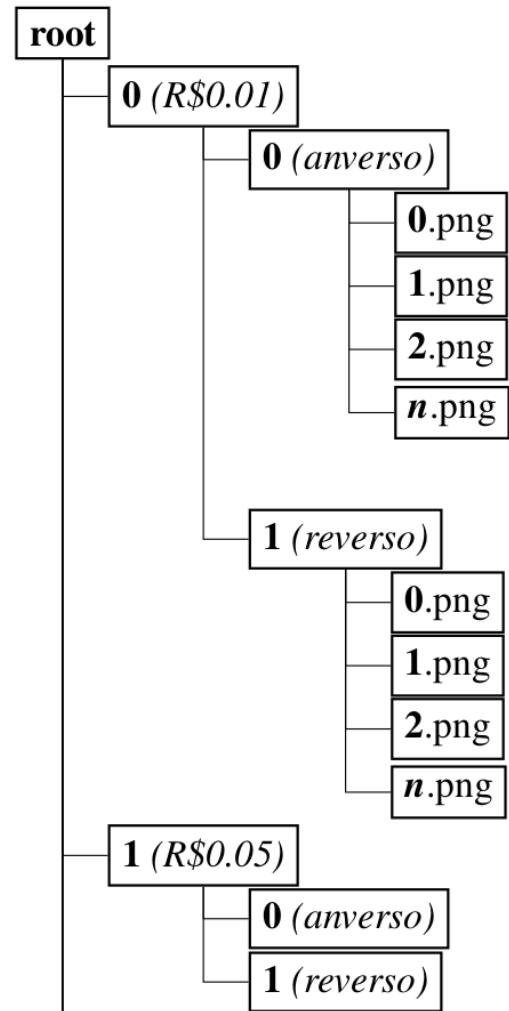
Organização de Arquivos

- Protocolo de obtenção de imagens
 - Definição de um protocolo padronizado para aquisição das imagens de moedas
 - Controle de variantes como resolução, iluminação



Organização de Arquivos

- Organização de Arquivos
 - Para cada valor de moeda
 - 100 imagens
 - 50 anverso
 - 50 reverso
 - Organização em diretórios



Sistema de cores

- Conversão para escala de cinza
 - Experimentos com classificação usando histograma de cores não apresentaram resultados promissores
 - Moedas do mesmo valor podem apresentar variações no material de confecção
 - Redução no custo computacional das operações



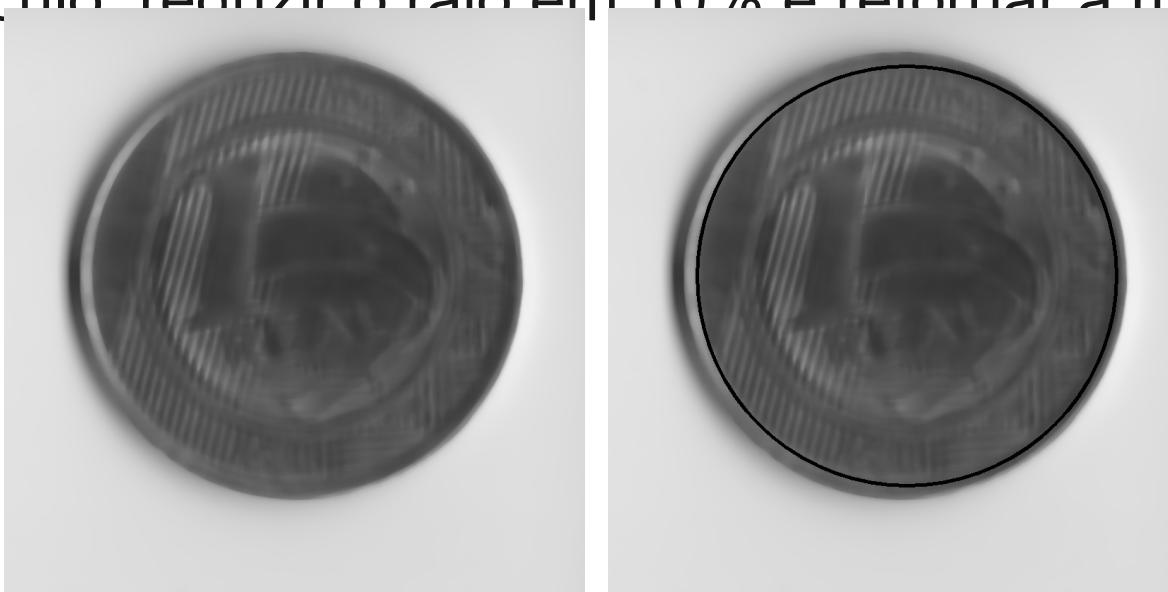
Filtragem de ruído

- Filtro de mediana
 - Aplicação de filtro de mediana de tamanho 5 para preservar detalhes importantes das imagens



Detecção de áreas de interesse

- Transformada de Hough para Círculos
 - Aplicação do filtro de mediana de tamanho 21 para borrar as bordas interiores da moeda
 - Aplicação da função HoughCircles() para detectar o círculo, reduzir o raio em 10% e retornar a máscara



Extração de descritores

- SIFT x SURF

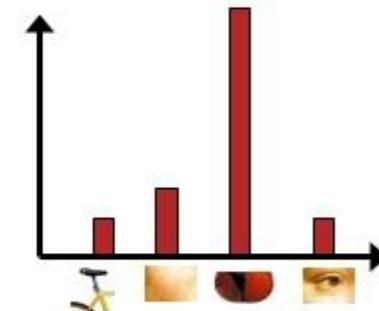
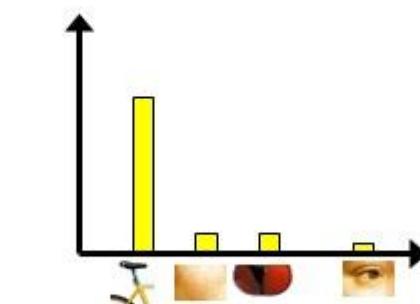
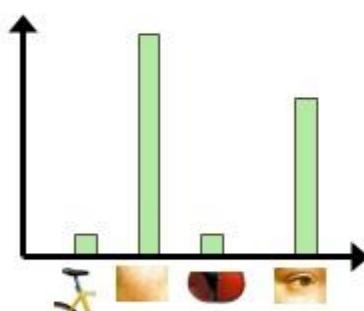
- Algoritmos de extração de características
- Dada uma região segmentada, extrai um conjunto de “palavras” que podem descrever tal região
- SIFT retornou os melhores “pontos de interesse”



Dicionário de características

- Adaptação do modelo Bag-Of-Words
 - Como a quantidade de “pontos de interesse” varia de acordo com a imagem analisada, é necessário colocá-los em caráter de equivalência
 - Este modelo analisa as imagens de um conjunto, seleciona as “palavras” que aparecem com mais frequência neste conjunto e as coloca em um “dicionário de palavras” de tamanho definido
 - Computacionalmente, este é o passo mais longo do método proposto. Por isso, o dicionário criado é salvo e lido nas execuções posteriores

Dicionário de características



Treinamento do SVM

- Support Vector Machine
 - Após a criação do “dicionário”, as imagens tem suas características extraídas novamente, porém considerando apenas as “palavras” do “dicionário”
 - Com isso, obtemos vetores descritores de tamanho fixo, contendo a incidência de cada “palavra”
 - Treinamento supervisionado SVM
 - Para cada amostra de moeda, é fornecido seu vetor descriptor e um label de com sua classe
 - O resultado é uma SVM treinada que pode ser salva e lida de um arquivo
 - Tempo de execução de aproximadamente 3min

Classificação

- Predição SVM
 - Imagem que desejamos classificar é lida
 - Seu vetor descritor é extraído utilizando o “dicionário de palavras” criado anteriormente
 - Com base na SVM criada, é realizada uma predição que informa a qual classe tal imagem pertence
 - Tempo de execução desta etapa é inferior a 1s

Resultados

- Dicionário de palavras
 - Testes realizados com diferentes tamanhos de dicionário e sua relação com a taxa de acurácia
 - Melhor resultado: Dicionário de 1000 palavras

Tamanho do dicionário	Tempo de criação	Média de acurácia
100	9min	60.41%
500	20min	75%
1000	29min	76.6%
2500	1h 14min	70.83%
5000	1h 57min	67.5%
10000	2h 33min	67.5%

Resultados

Tamanho do dicionário		1000
Tempo de criação do dicionário		29min
Valor da Moeda		Acurácia
R\$0.01	anverso	60%
	reverso	65%
R\$0.05	anverso	65%
	reverso	55%
R\$0.10	anverso	70%
	reverso	100%
R\$0.25	anverso	35%
	reverso	100%
R\$0.50	anverso	70%
	reverso	100%
R\$1.00	anverso	100%
	reverso	100%
Média de acurácia		76.66%

Resultados

- Comparação com o trabalho de Jory (2011)
 - Classificação individual de moedas
 - Não requer multiplas moedas na imagem
 - Classificação feita através de características SIFT
 - Não leva em conta cor, raio e diametro
 - Tempo de predição inferior a 1s
 - No trabalho anterior, cerca de 4min por moeda

Conclusão

- Desenvolvimento de algoritmo eficaz, capaz de identificar e classificar moedas de real com assertividade superior à 75%
- Metodologia de encadeamento de técnicas de PI aplicados para transformar imagens brutas em dados inteligíveis
- Trabalhos futuros podem trazer melhorias na parametrização dos métodos utilizados