

# Métodos para reconhecimento de caracteres manuscritos

Matheus S. Guerra<sup>1</sup>, Vitor Y. Shnohara<sup>2</sup>

<sup>1,2</sup>Universidade Tecnológica Federal do Paraná (UTFPR)  
Campo Mourão – PR – Brazil

guerramatheus2@gmail.com, vitor.ysh@gmail.com

**Abstract.** *This document describes the classification of a set of approximately 60 thousand images of handwritten characters obtained through the NIST database, where the characteristic extraction procedure is performed by counting the pixels by quadrants and labeling them, sorting the data generated through the classifiers K nearest neighbors, Support Vector Machine, Decision Tree and Majority Vote, and finally getting hit rates and their analysis.*

**Resumo.** *Este documento descreve a classificação de um conjunto de aproximadamente 60 mil imagens de caracteres manuscritos obtidos através da base NIST, onde é realizado o procedimento de extração de características através da contagem de pixels por quadrantes e rotulação das mesmas, classificação dos dados gerados através do classificador K nearest neighbors, Support Vector Machine, Árvore de Decisão e Voto Majoritário, e finalmente obtenção das taxas de acertos e sua análise.*

## 1. Introdução

Com o constante avanço da área de visão computacional, dentro da inteligência artificial, o tema de reconhecimento de imagens vem ganhando atenção de cientistas da área devido principalmente à facilidade de acesso a grandes quantidades de imagens disponibilizadas por sites com esse propósito.

O reconhecimento de caracteres manuscritos em imagens digitais é uma importante e desafiadora área de estudos da Visão Computacional, e embora essa área tenha apresentado grandes avanços, ainda há muito a ser feito para que uma máquina obtenha um desempenho parecido ao do ser humano em uma leitura (CRUZ, 2008). Uma grande dificuldade encontrada, não só em reconhecimento de caracteres, mas em imagens em geral, são os diferentes estilos do mesmo, por exemplo, uma letra 'A' pode ser escrita de diversas formas, além de maiúsculo ou minúscula.

Neste documento, abordaremos um estudo sobre o reconhecimento de caracteres apenas em maiúsculo, através da extração de características de 26 classes.

## **2. Objetivo**

O objetivo deste documento é obter uma taxa de acertos considerável na classificação e reconhecimento de imagens de caracteres em 26 classes diferentes as quais representam todas as letras do alfabeto utilizando diversos métodos de extração de características e o uso de diferentes classificadores, tais quais serão citados em seções futuras, além de informar o(a) leitor(a) os procedimentos para que seja possível realizar a mesma atividade descrita.

## **3. Materiais**

Para realizar os procedimentos, foi utilizado um computador portátil com o sistema operacional Linux Ubuntu versão 16.04 LTS com processador Intel Core i5-5200U e memória de 8 GB DDR3.

Foi utilizado o software *GNU Octave* para a extração de características das diversas imagens.

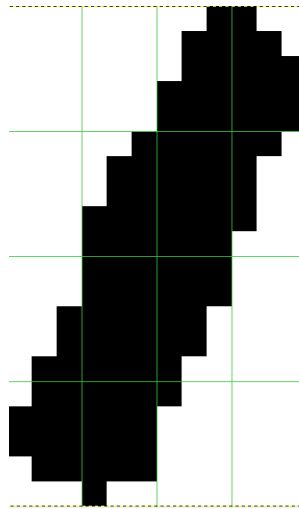
O conjunto de imagens foi retirado da base NIST, a qual continha aproximadamente 221.000 imagens de diversos caracteres manuscritos.

## **4. Fundamentação Teórica**

O procedimento de classificação se baseia em 4 principais etapas as quais serão descritas nas subseções a seguir, as quais são separar imagens para treino, teste e validação, utilizar métodos para retirar características e realizar a normalização das mesmas, utilizar um classificador e finalmente obter as taxas de acerto.

### **4.1. Método para extração de características**

Diante dos diversos métodos para extração de características, tais quais Zenirke, histogramas horizontais e verticais, foi utilizado a quantidade de *pixels* brancos e pretos presentes em cada um dos 16 quadrantes da imagem.



**Figure 1. Ilustração de uma imagem dividida em 16 quadrantes**

Basicamente, seu funcionamento consiste em percorrer a matriz de pixels da imagem e dividir a mesma em 16 quadrantes. Após este procedimento, percorremos todos os *pixels* do quadrante contabilizando a quantidade de *pixels* pretos e brancos e em seguida, é armazenado em um vetor as devidas quantidades.

---

**Algorithm 1** ExtrairCaracteristicas

---

Seja  $I[i,j]$  a matriz de pixels que representa a imagem ;  
 Seja  $i$  a altura da matriz de pixels ;  
 Seja  $j$  a largura da matriz de pixels ;

```

1: function EXTRAIRCARACTERISTICAS(QtdeDivisoies)
2:   VetorLinha = Índices de fronteira de altura
3:   VetorColuna = Índices de fronteira de largura
4:   for  $i \leftarrow 1$  até QtdeDivisoies do
5:     for  $j \leftarrow 1$  até QtdeDivisoies do
6:       for  $a \leftarrow$  VetorLinha[ $i$ ] até VetorLinha[ $i+1$ ] do
7:         for  $b \leftarrow$  VetorColuna[ $j$ ] até VetorColuna[ $j+1$ ] do
8:           if  $I[a,b]$  = preto then
9:             PixelsPreto + 1
10:          else
11:            PixelsBranco + 1
12:          end if
13:        end for
14:      end for
15:      VetorPixelPreto[Last] = PixelsPreto
16:      VetorPixelBranco[Last] = PixelsBranco
17:    end for
18:  end for
19: end function

```

---

**Figure 2. Pseudocódigo para extração de características**

Em seguida, salvamos as devidas características normalizadas separadas por vírgula acompanhando o rótulo a qual pertence a imagem em seu respectivo arquivo, seja de teste, treino ou validação.

## 4.2. Normalização dos dados

Para a aplicação das características em um classificador, é necessário que as mesmas estejam normalizadas, logo, foi implementado um método para a normalização citado na subseção anterior.

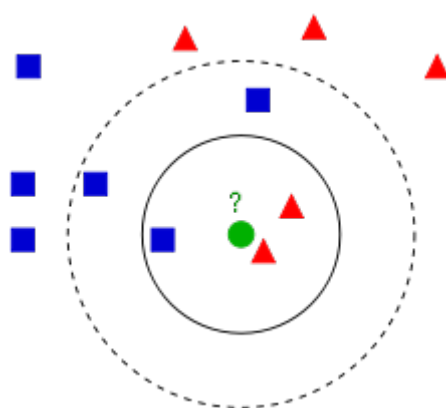
## 4.2. Classificadores

Dentre diversos algoritmos de classificação, foi utilizado 4 listados a seguir, os quais são K nearest neighbors (K-nn), Support Vector Machine (SVM), Árvore de Decisão e Voto Majoritário.

### 4.2.1. *K nearest neighbors*

Um dos classificadores utilizados foi o classificador *k nearest neighbors*, o qual apresentou melhor resultado, seu funcionamento consiste em pegar as instâncias de treino e *plotar* as mesmas em um gráfico de plano cartesiano. Após isso, as instâncias de testes passam pelo mesmo procedimento.

Para determinar a classe de um elemento que não pertença ao conjunto de treinamento, o classificador KNN procura *K* elementos do conjunto de treinamento que estejam mais próximos deste elemento desconhecido, ou seja, que tenham a menor distância. Estes *K* elementos são chamados de K-vizinhos mais próximos. Verifica-se quais são as classes desses *K* vizinhos e a classe mais freqüente será atribuída à classe do elemento desconhecido.



**Figure 3. Funcionamento do algoritmo K-nearest neighbors [3]**

Neste experimento, o cálculo da distância foi calculado através da distância Euclidiana, tendo sua fórmula descrita na figura 4.

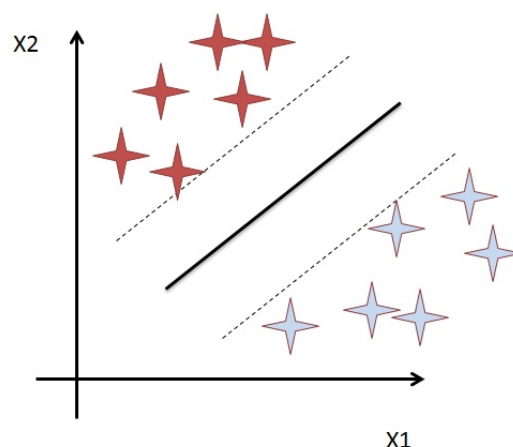
$$d(x, y) = \sqrt{w_1(x_1 - y_1)^2 + w_2(x_2 - y_2)^2 + \dots + w_n(x_n - y_n)^2}.$$

**Figura 4. Fórmula da distância Euclidiana**

KNN é um classificador que possui apenas um parâmetro livre (o número de K-vizinhos) que é controlado pelo usuário com o objetivo de obter uma melhor classificação. O qual foi utilizado  $K = 1, 3$  e  $5$  para realizar as devidas classificações.

#### **4.2.2. Support Vector Machine**

O Support Vector Machine é um dos métodos de classificação por aprendizagem supervisionada mais conhecidos, onde o algoritmo toma como entrada um conjunto de dados e prediz, para cada entrada qual classe a entrada faz parte. Dados um conjunto de exemplos de treinamento, o SVM constrói um modelo que atribui novos exemplos a uma determinada categoria. Um modelo SVM é uma representação de exemplos como pontos no espaço, mapeados de maneira que os exemplos de cada categoria sejam divididos por um espaço claro que seja tão amplo quanto possível. Os novos exemplos são então mapeados no mesmo espaço e preditos como pertencentes a uma categoria baseados em qual o lado do espaço eles são colocados.



**Figura 4. Esquema de classificação do SVM**

### 4.2.3. Árvore de Decisão

Outro método bem conhecido para se realizar a classificação de instâncias é a árvore de decisão, a qual tem a capacidade de aprender a tomar decisões onde podemos comparar seu funcionamento com a regra de *if-then*.

Para qualquer problema de árvore de decisão, deve-se inicialmente definir atributos para descrever classes e assim realizar a classificação de forma efetiva.

Um exemplo clássico para fixar a ideia de seu funcionamento, é o exemplo de um paciente o qual deve ser classificado e o médico.



Figura 5. Dados representados em uma tabela para tomada de decisão [6]

A primeira pergunta que pode ser feita ao paciente é se ele tem sentido dor (corresponderia ao nó-raiz da árvore de decisão). A seguir, outras perguntas podem ser feitas, dependendo da resposta obtida. Por exemplo, se o paciente está tendo febre ou enjoos, ou ainda se tem notado alguma mancha no corpo, sendo assim, é possível utilizar uma árvore de decisão para classificar um novo paciente como saudável ou doente. Para isso, basta partir do nó raiz da árvore e ir percorrendo-a, através das respostas aos testes dos nós internos, até chegar em um nó folha, o qual indica a classe correspondente do novo paciente. [6]

## 5. Procedimentos e Resultados

Primeiramente, foi feito um algoritmo na linguagem *Octave*, o qual percorria os arquivos de teste (NIST\_Test\_Upper.txt), treino (NIST\_Train\_Upper.txt) e validação (NIST\_Valid\_Upper.txt), os quais continham nomes de imagens e seus respectivos diretórios para a separação de conjuntos e foi extraído as características através do método discutido na subseção 4.1 e armazenados em seus devidos arquivos (*train.txt*, *test.txt* e *valid.txt*). Após este procedimento, executamos o algoritmo dos classificadores

passando seus devidos parâmetros e os arquivos com as características geradas, obtendo após aproximadamente 20 minutos, a taxa de acerto e sua respectiva matriz de confusão expressa a seguir.

## 5.1 Taxas de acerto

Segundo os resultados obtidos, o melhor classificador foi o k-nn o parâmetro  $k = 5$ , obtendo uma taxa de 83.5% de acertos como mostra a tabela 1.

**Tabela 1. Taxas de acertos, parâmetros e classificadores utilizados**

Classificador	K-nn			SVM	Árvore de Decisão	Voto Majoritário
Parâmetro	K = 1	K = 3	K = 5	-	-	-
Taxa de Acerto (%)	82.380035 1729	83.033246 7968	83.544091 7846	74.147893 8112	69.742902 6045	81.726823 5491

## 5.2 Matrizes de confusão

Nesta seção será apresentadas as matrizes de confusão para os diferentes classificadores com os diferentes parâmetros

### 5.2.1 Classificador K-nn

```

372 009 000 007 000 002 001 015 001 000 008 000 008 007 002 005 002 014 000 001 000 000 001 003 001 000
016 288 003 019 008 002 014 003 000 000 003 000 008 000 016 000 016 026 005 000 004 001 002 001 000 000
000 000 495 000 001 002 002 000 000 000 001 010 000 000 001 001 000 001 002 000 002 000 000 000 000 000
000 019 006 318 000 001 000 000 000 002 000 000 000 001 030 006 005 000 000 000 000 007 001 000 000 000
001 006 025 002 263 028 007 000 001 000 002 002 000 000 002 001 003 018 003 000 000 000 000 000 000 001
000 000 001 000 013 332 000 000 002 000 000 000 001 000 000 052 001 001 000 011 000 000 001 000 002 002
000 008 010 004 003 001 305 000 000 000 001 003 007 000 000 011 000 020 001 002 000 005 001 007 000 000 000
005 004 000 002 000 000 001 336 000 000 003 000 013 013 000 001 000 000 000 001 007 001 014 000 001 000
000 000 005 002 007 006 000 000 685 046 004 004 000 000 000 002 000 001 006 005 000 001 000 003 006 032
000 001 000 001 000 001 001 000 011 368 000 000 000 000 003 000 005 001 019 008 000 000 000 000 004 003
002 001 004 001 008 002 001 004 000 000 303 009 004 000 000 000 000 000 013 000 000 002 004 006 012 001 000
000 001 007 000 001 000 001 000 001 000 000 483 000 000 000 000 000 000 000 000 000 000 001 001 000 000 000
016 009 000 001 000 000 000 085 000 000 013 000 268 045 000 001 001 002 000 000 002 003 014 000 000 000
007 002 001 000 000 000 001 010 000 000 003 001 013 372 000 001 002 002 000 000 004 001 018 001 000 000
000 008 016 073 000 000 003 000 001 000 000 000 000 000 000 319 002 022 002 000 000 013 000 000 000 000 000
000 003 001 015 001 025 000 000 000 000 000 000 000 000 000 002 407 003 001 000 002 000 005 000 000 002 000
001 016 006 005 000 001 036 000 000 000 000 000 000 000 001 026 002 333 013 000 000 012 000 000 000 000 000
010 026 006 002 024 007 000 004 000 000 023 003 004 000 004 006 010 307 000 000 001 000 001 004 000 004
000 001 001 003 004 004 002 000 001 008 001 000 000 000 003 000 002 001 412 000 001 000 000 000 001 000
000 000 000 001 000 006 001 000 001 004 000 000 001 000 000 002 000 000 000 000 451 000 000 000 000 002 000
000 001 000 000 001 000 000 002 000 003 001 000 000 001 002 000 000 000 000 000 000 391 043 012 000 001 000
000 000 000 000 000 000 000 000 000 002 001 000 000 001 000 000 000 000 000 000 000 021 442 001 002 012 000
000 000 000 000 000 000 002 006 000 001 010 000 007 029 000 000 001 001 000 000 039 013 365 001 000 000
002 000 000 000 000 002 000 002 000 002 017 002 003 000 000 000 000 000 002 000 000 009 004 415 010 002
000 000 000 000 000 000 000 001 002 001 000 000 000 000 000 004 000 000 001 008 000 034 001 004 397 000
000 015 000 012 007 000 001 000 007 002 000 002 001 000 000 002 000 003 000 000 000 001 000 001 003 410

```

**Figura 6: Matriz de confusão para K = 1 utilizando K-nn**



404	009	000	009	001	001	002	009	001	000	002	000	005	001	001	003	001	005	000	001	000	000	001	002	001	000
024	315	003	019	006	000	009	001	000	000	001	000	005	000	013	000	012	014	004	000	004	001	003	001	000	000
000	000	504	000	000	002	000	000	000	000	000	008	000	000	001	001	000	000	001	000	001	000	000	000	000	000
000	018	004	332	000	001	000	000	000	003	000	000	000	000	024	002	004	000	000	000	005	003	000	000	000	000
000	011	035	002	266	018	010	000	001	000	001	004	000	000	001	001	009	002	000	000	000	000	000	000	000	003
000	001	001	000	006	338	000	000	002	000	000	000	000	000	000	056	000	000	000	013	000	000	001	000	001	000
003	013	013	004	007	000	304	000	000	000	001	003	006	000	008	000	017	000	001	000	003	001	005	000	000	000
007	005	000	002	000	001	001	349	000	001	000	000	012	007	000	000	000	000	000	001	007	001	008	000	000	000
000	001	006	004	010	013	000	000	691	045	002	002	000	000	000	002	000	000	006	005	000	000	000	001	002	025
000	002	001	002	001	004	000	000	020	367	000	000	000	000	002	000	002	000	015	008	000	000	000	000	002	000
006	001	006	001	009	004	001	003	000	000	301	011	003	000	000	000	000	008	000	000	003	003	008	009	000	000
000	001	010	001	000	000	000	000	000	000	000	482	000	000	000	000	000	000	000	000	000	001	001	000	000	000
019	012	000	000	000	003	000	113	000	000	012	000	247	030	000	001	000	001	000	000	004	003	014	001	000	000
009	005	001	001	001	000	001	015	000	000	007	001	009	370	000	000	001	002	000	000	006	001	007	001	001	000
001	014	012	086	000	000	002	000	001	000	000	000	000	000	324	001	013	000	000	000	005	000	000	000	000	000
003	002	001	016	001	012	000	000	000	000	000	000	000	001	000	420	002	001	000	001	000	005	000	000	002	000
001	017	011	011	002	001	036	000	000	000	000	000	000	001	025	002	333	005	001	000	006	000	000	000	000	000
020	035	017	005	025	008	000	004	001	000	021	003	002	000	002	005	008	284	000	000	000	001	000	002	000	003
000	001	001	005	003	006	003	000	000	007	000	000	000	000	003	000	000	000	414	000	001	000	000	000	001	000
000	000	000	000	000	006	001	000	002	004	000	000	001	000	000	001	000	000	000	452	000	000	000	000	002	000
001	000	001	000	000	000	002	001	000	004	001	000	000	002	002	000	001	000	000	000	396	036	010	000	001	000
000	001	000	000	000	000	000	000	001	001	000	000	000	000	000	000	000	000	000	000	016	453	000	000	010	000
002	000	001	001	002	000	000	007	000	002	006	000	008	031	000	000	001	000	000	000	043	021	349	001	000	000
005	000	001	002	002	004	000	003	001	000	023	001	001	000	000	000	001	001	000	000	000	011	002	406	006	002
000	000	000	000	000	003	000	001	002	000	001	000	000	000	000	005	000	001	001	002	000	031	001	004	401	000
000	020	001	017	005	001	001	000	003	002	001	001	000	000	000	001	000	001	000	000	000	000	000	000	000	413

Figura 7: Matriz de confusão para K = 3 utilizando K-nn

401	008	000	008	000	002	002	010	001	000	003	000	003	002	003	004	001	006	000	000	000	000	002	002	001	000
018	316	002	015	008	001	011	001	000	000	001	000	005	001	015	000	013	018	004	000	004	001	001	000	000	000
000	000	499	001	002	002	002	000	000	000	001	009	000	000	001	001	000	000	000	000	000	000	000	000	000	000
002	014	000	340	000	001	002	000	000	001	002	000	000	000	025	001	002	000	000	000	005	003	000	000	000	000
000	012	036	002	273	013	010	000	002	000	000	004	000	000	001	001	001	007	002	000	000	000	000	000	000	001
000	001	001	000	008	335	000	000	001	000	000	000	001	000	000	056	000	000	000	015	000	001	000	000	000	000
002	005	012	003	005	000	311	000	000	000	001	002	006	000	011	000	016	002	002	000	005	001	005	000	000	000
007	003	000	002	001	000	001	352	000	001	002	000	010	007	000	001	000	000	000	001	007	001	006	000	000	000
000	001	006	003	007	009	001	000	692	043	001	003	000	000	000	002	000	000	009	006	000	000	000	001	002	029
000	000	001	003	000	002	000	000	017	369	000	000	000	000	004	000	001	000	018	009	000	000	000	000	002	000
002	000	005	001	008	004	000	003	000	000	299	012	006	001	000	000	000	010	000	000	002	004	010	010	000	000
000	000	006	000	000	000	000	000	001	000	000	485	000	000	000	000	000	000	001	000	000	001	002	000	000	000
014	009	000	000	000	003	000	116	000	000	009	000	257	029	000	001	000	002	000	000	004	003	011	002	000	000
006	004	001	000	001	000	001	019	000	000	004	001	013	368	000	000	000	001	000	000	007	001	011	000	001	000
002	006	008	083	000	001	002	000	001	000	000	000	000	000	336	001	014	000	000	000	005	000	000	000	000	000
001	003	001	020	001	008	000	000	002	000	000	000	000	000	000	420	003	000	000	001	000	004	000	000	003	000
001	016	008	011	001	001	029	000	000	000	000	000	000	001	030	001	339	004	000	000	008	000	001	000	001	000
014	035	011	007	024	009	001	002	000	000	022	008	004	001	003	005	006	285	001	000	001	001	000	003	000	003
000	000	001	004	003	004	002	000	001	007	000	000	000	000	004	000	001	000	417	000	000	000	000	000	001	000
000	000	000	000	000	006	000	000	001	002	000	000	001	000	000	002	001	000	000	452	000	000	000	000	004	000
000	001	001	000	000	000	001	001	000	003	001	000	000	000	001	000	000	000	000	000	406	034	008	000	001	000
000	001	000	000	000	000	000	000	001	001	001	000	000	000	000	000	000	000	000	000	023	445	000	001	009	000
001	000	000	000	002	000	000	004	000	003	003	000	010	034	000	000	000	000	000	000	050	018	350	000	000	000
004	000	001	003	001	001	000	000	001	001	024	002	001	000	000	000	001	001	000	000	000	009	001	410	009	002
000	000	000	000	000	000	000	002	002	000	000	000	000	000	000	004	000	000	001	003	000	029	001	007	404	000
000	018	001	015	003	001	000	000	001	003	002	002	000	000	000	001	000	003	000	002	000	000	000	000	000	415

Figura 8: Matriz de confusão para K = 3 utilizando K-nn



### 5.2.1 Classificador SVM

337	025	000	002	003	004	002	022	002	000	017	002	016	004	003	003	000	009	000	000	000	000	003	002	003	000	
009	213	003	031	015	003	022	000	000	000	006	000	011	002	029	000	030	029	011	000	006	000	006	006	001	002	
000	000	494	000	006	001	002	000	000	000	000	010	000	000	002	001	000	000	000	000	002	000	000	000	000	000	
000	029	001	304	000	000	000	000	001	001	000	000	002	002	027	002	004	000	000	000	017	004	002	000	000	000	
000	006	071	001	219	008	022	000	000	000	006	006	000	000	000	002	001	008	007	001	000	000	000	001	000	006	
000	001	002	000	009	316	001	001	000	001	000	000	000	000	000	000	054	000	000	001	032	000	001	000	000	000	
003	005	011	001	003	004	266	000	000	000	003	005	007	000	008	000	022	000	000	000	014	000	037	000	000	000	
012	004	000	001	001	002	000	270	000	001	003	000	046	017	003	002	000	000	000	000	003	015	018	000	004	000	
001	000	002	002	019	009	000	000	529	065	000	000	000	000	000	000	001	000	001	047	006	000	000	000	020	004	109
000	001	000	025	010	003	000	000	017	328	000	000	000	000	003	000	002	000	025	008	000	000	000	000	000	004	
008	002	016	000	010	003	000	001	000	000	266	016	006	001	000	001	000	010	000	000	004	001	019	013	000	000	
000	000	007	000	001	000	001	000	001	000	003	473	000	000	000	000	000	000	000	001	000	007	001	001	000	000	000
024	011	000	001	000	000	000	087	000	000	005	000	275	013	001	006	002	006	000	000	002	004	014	001	008	000	
017	006	000	001	000	001	000	017	000	000	010	000	053	271	001	000	002	003	000	000	009	007	034	004	003	000	
000	017	005	082	000	000	001	000	002	000	000	000	001	000	293	001	040	000	000	000	017	000	000	000	000	000	
003	004	001	017	000	009	000	001	001	000	000	000	000	000	002	410	004	000	000	001	001	004	000	000	009	000	
000	014	004	014	002	000	023	000	000	001	005	000	000	002	017	002	332	004	001	000	023	001	003	000	000	004	
007	021	016	002	022	010	001	003	001	000	057	007	009	001	002	003	012	239	000	000	008	002	004	014	000	005	
000	002	001	024	004	001	007	000	003	007	001	001	000	000	004	000	002	000	382	002	000	000	000	001	003	000	
000	004	001	000	000	001	000	001	001	009	000	000	003	000	000	000	000	000	000	441	000	000	000	000	008	000	
000	000	001	002	000	000	001	002	000	000	001	010	000	002	003	000	003	000	001	000	393	029	010	000	000	000	
000	000	000	000	001	002	000	002	001	001	002	000	002	023	000	000	000	000	002	000	031	392	001	002	020	000	
001	000	000	002	000	000	001	004	000	000	011	001	012	031	002	000	002	000	001	000	067	027	311	002	000	000	
018	002	000	002	000	003	000	000	004	000	031	000	001	004	000	000	001	003	003	000	000	007	006	362	024	001	
000	000	000	001	000	004	000	009	000	001	003	000	001	001	000	002	000	000	000	011	000	033	000	010	377	000	
000	015	003	017	009	003	000	000	003	017	004	008	001	000	000	001	002	013	002	001	000	000	000	007	000	361	

Figura 9: Matriz de confusão do classificador SVM

299	020	001	007	002	008	004	021	003	001	008	002	013	010	003	003	007	027	004	001	003	000	006	005	001	000	
018	200	005	027	012	013	024	008	003	004	001	001	010	002	017	002	023	031	008	000	007	001	008	001	001	008	
000	000	461	000	007	010	005	000	000	001	001	013	001	000	003	003	001	006	000	002	004	000	000	000	000	000	
003	021	001	277	003	001	001	002	000	002	000	001	006	007	035	007	008	004	000	001	013	001	001	000	000	001	
003	011	028	000	211	022	028	001	000	002	002	002	000	000	002	003	002	026	007	001	000	000	001	001	000	012	
011	001	002	000	014	310	010	000	002	002	000	000	001	000	000	042	001	002	003	013	000	000	000	001	003	001	
002	007	011	006	014	007	244	003	000	000	004	003	008	000	007	000	029	005	006	000	027	003	003	000	000	000	
013	007	001	001	001	002	003	260	000	000	005	001	029	023	003	001	001	007	000	000	007	007	028	000	001	001	
003	004	008	003	011	008	000	002	542	064	002	009	001	001	003	012	009	011	029	020	003	005	001	017	009	038	
001	003	002	010	001	000	004	001	012	326	000	000	000	000	001	003	001	002	000	030	010	001	000	003	001	006	008
005	005	014	000	021	004	004	011	000	001	233	018	005	004	000	001	001	015	006	001	002	003	009	013	001	000	
002	000	021	000	003	001	002	000	002	001	002	446	004	001	000	000	000	000	001	002	000	002	000	002	004	000	000
022	013	003	003	001	003	000	081	000	000	013	000	218	034	003	003	006	013	001	002	004	002	024	006	005	000	
010	003	003	002	001	000	003	034	002	000	010	000	019	289	002	001	005	011	002	001	003	006	027	002	001	002	
001	006	021	082	000	001	001	001	000	002	000	000	001	002	283	000	024	006	001	000	026	001	000	000	000	000	
003	007	005	015	004	032	001	002	002	001	000	000	002	002	001	352	009	005	000	006	001	007	003	000	007	000	
004	037	007	009	005	002	036	002	000	001	000	000	005	004	020	001	287	011	002	000	015	000	002	000	001	001	
011	024	012	002	028	006	009	003	002	000	024	005	010	001	002	002	019	260	001	001	006	000	003	009	000	006	
002	005	003	004	006	004	006	000	008	016	001	000	001	002	000	000	002	002	369	002	002	001	001	001	005	002	
000	002	000	003	000	006	000	000	004	013	001	000	002	000	000	008	000	001	005	412	000	001	000	001	003	007	
000	001	001	005	000	000	001	007	000	000	003	003	005	006	004	003	002	000	002	000	364	039	010	000	002	000	
000	000	000	000	000	001	001	002	002	002	001	000	004	005	001	000	000	000	002	001	028	379	016	006	031	000	
001	002	000	002	001	002	005	009	000	002	010	000	023	042	003	001	006	002	002	000	029	026	305	001	001	000	
009	003	002	003	003	006	002	005	004	009	030	007	003	003	000	000	000	006	005	001	000	010	003	334	014	010	
003	000	000	002	000	002	000	004	004	002	004	000	001	005	000	002	000	002	003	017	002	049	004	028	319	000	
000	015	004	008	014	006	001	001	010	017	002	006	000	000	000	003	006	016	000	001	001	001	000	007	000	348	

**Figura 10: Matriz de confusão do classificador Árvore de Decisão**

```

391 008 000 003 001 002 001 019 001 000 002 001 007 002 003 004 001 008 000 000 000 000 000 004 001 000
022 293 001 014 007 001 011 005 002 000 000 000 003 001 014 000 025 023 003 000 003 000 001 004 001 001
000 000 504 001 001 001 002 000 000 001 000 005 000 000 001 002 000 000 000 000 000 000 000 000 000 000
003 019 000 333 000 000 000 001 001 001 000 000 001 001 020 003 005 000 000 000 005 003 000 000 000 000
000 012 044 000 245 007 013 001 001 001 001 001 000 000 001 004 001 018 007 001 000 000 000 001 000 006
000 001 002 000 008 335 000 000 001 000 000 000 001 000 000 057 000 000 000 012 000 000 000 000 001 001
003 007 010 005 003 000 290 002 000 000 003 002 004 000 006 000 024 003 002 000 010 000 015 000 000 000
007 002 000 002 001 000 000 336 000 001 002 001 022 007 001 001 000 000 000 000 005 003 010 000 001 000
000 000 004 001 009 002 000 001 661 043 000 001 000 000 000 003 000 000 012 006 000 000 000 024 004 044
001 000 000 005 001 001 000 001 017 366 000 000 000 000 002 001 003 000 016 007 000 000 000 001 004 000
006 002 013 000 004 002 000 007 000 000 289 010 002 001 000 001 000 009 002 000 002 002 013 012 000 000
000 000 009 000 000 000 000 000 001 000 002 481 000 000 000 000 000 000 001 000 000 000 001 001 000 000
022 009 001 000 000 000 000 113 000 000 005 000 258 014 001 004 002 008 000 000 002 002 010 005 004 000
015 005 001 000 000 000 001 034 001 000 008 001 015 326 000 001 005 004 000 001 005 002 009 003 002 000
002 012 006 069 000 000 000 000 002 001 000 000 001 000 322 001 036 000 000 000 007 000 000 000 000 000
003 003 000 018 000 007 000 001 000 000 000 000 000 000 001 427 003 000 000 000 000 003 000 000 001 000
000 017 007 010 001 000 014 001 000 000 000 000 000 001 014 003 366 005 000 000 010 000 001 000 001 001
017 023 011 001 011 008 000 004 002 000 024 003 004 000 002 005 019 288 001 000 002 000 001 016 000 004
001 001 002 004 003 000 003 000 001 011 001 000 000 000 003 000 003 000 407 001 000 000 000 001 003 000
000 000 001 001 000 001 000 001 001 003 000 000 002 000 000 002 001 000 000 451 000 000 000 000 005 000
000 000 002 000 000 000 001 002 000 000 002 000 000 002 002 001 002 000 001 000 406 029 007 000 001 000
000 000 000 000 000 000 000 000 001 002 001 000 000 003 000 000 000 000 000 001 024 428 001 003 018 000
002 000 000 002 000 000 001 006 000 002 009 000 011 031 001 000 002 000 001 000 053 016 336 001 001 000
006 000 000 003 001 002 000 001 001 001 022 001 000 001 000 000 001 000 000 000 000 004 000 415 011 002
000 000 000 000 000 001 000 003 001 001 001 000 000 001 000 004 000 000 000 007 000 026 000 013 395 000
000 013 001 009 002 000 000 001 001 006 004 001 000 000 000 002 005 006 000 001 000 000 000 005 000 410

```

**Figura 11: Matriz de confusão do classificador Voto Majoritário**

## 6. Conclusão

Dentre diversos outros métodos para extração de características já citados anteriormente, a contagem de *pixels* por quadrante, apesar de sua fácil implementação, possui muitas vezes uma taxa de acertos considerável dependendo da quantidade de quadrantes a qual uma imagem é dividida, além de possuir um desempenho alto, juntamente com o classificador *K nearest neighboor* com o parâmetro de quantidade de vizinhos igual a 5, foi obtido um total de 83.54% de acertos.

## 7. Referências

- [1] SILVA, F. A. et al. Um algoritmo rápido para o reconhecimento de caracteres. In: WORKSHOP DE VISÃO COMPUTACIONAL (WVC 2011), 7. Anais... Curitiba: UFPR, 2011. p. 149-154.
- [2] SILVA, F. A. et al. Reconhecimento de caracteres baseado em regras de transições entre pixels vizinhos. Avanços em visão computacional. Curitiba: Omnipax, 2012. p. 265-282.
- [3] Wikipedia. “*k*-nearest neighbors algorithm”, [https://en.wikipedia.org/wiki/K-nearest\\_neighbors\\_algorithm](https://en.wikipedia.org/wiki/K-nearest_neighbors_algorithm), Junho.
- [4] Wikipedia. “Máquina de Vetores de Suporte”, [https://pt.wikipedia.org/wiki/M%C3%A1quina\\_de\\_vetores\\_de\\_suporte](https://pt.wikipedia.org/wiki/M%C3%A1quina_de_vetores_de_suporte), Junho.
- [5] WU, M. P. Handwritten character recognition. St Lucia: The School of Information technology and Electrical Engineering the University of Queensland: 2003.4
- [6] ZUBEN, V. J. et al. “Árvore de Decisão”. Campinas. [ftp://ftp.dca.fee.unicamp.br/pub/docs/vonzuben/ia004\\_1s10/notas\\_de\\_aula/topico7\\_I\\_A004\\_1s10.pdf](ftp://ftp.dca.fee.unicamp.br/pub/docs/vonzuben/ia004_1s10/notas_de_aula/topico7_I_A004_1s10.pdf), Junho.