

UNIVERSIDADE ESTADUAL DE MARINGÁ  
CENTRO DE TECNOLOGIA  
DEPARTAMENTO DE INFORMÁTICA  
PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

LUCAS GEORGES HELAL

**IDENTIFICAÇÃO DE ESCRITORES USANDO CNN's COM  
ABORDAGEM DE DISSIMILARIDADE.**

Maringá

2019

LUCAS GEORGES HELAL

**IDENTIFICAÇÃO DE ESCRITORES USANDO CNN's COM  
ABORDAGEM DE DISSIMILARIDADE.**

Dissertação apresentada ao Programa de Pós-Graduação em Ciência da Computação do Departamento de Informática, Centro de Tecnologia da Universidade Estadual de Maringá, como requisito parcial para obtenção do título de Mestre em Ciência da Computação.

Orientador: Prof. Dr. Diego Bertolini  
Gonçalves

Coorientador: Prof. Dr. Yandre Maldonado  
e Gomes da Costa

Maringá  
2019

Dados Internacionais de Catalogação na Publicação (CIP)  
(Biblioteca Central - UEM, Maringá, PR, Brasil)

H474i Helal, Lucas Georges  
Identificação de escritores usando CNN's com  
abordagem de dissimilaridade / Lucas Georges Helal.  
-- Maringá, 2019.  
73 f. : il. color., figs., tabs.

Orientador: Prof. Dr. Diego Bertolini Gonçalves.  
Coorientador: Prof. Dr. Yandre Maldonado e Gomes  
da Costa.

Dissertação (mestrado) - Universidade Estadual de  
Maringá, Centro de Tecnologia, Departamento de  
Informática, Programa de Pós-Graduação em Ciência da  
Computação, 2019.

1. *Convolutional Neural Network* (CNN). 2.  
Inteligência artificial. 3. Rede neural  
convolucional. 4. Identificação de escritores.  
dissimilaridade. I. Gonçalves, Diego Bertolini,  
orient. II. Costa, Yandre Maldonado e Gomes da. III.  
Universidade Estadual de Maringá. Centro de  
Tecnologia. Departamento de Informática. Programa de  
Pós-Graduação em Ciência da Computação. IV. Título.

CDD 21.ed. 006.3

# FOLHA DE APROVAÇÃO

LUCAS GEORGES HELAL


## Identificação de escritores usando CNN's com abordagem de dissimilaridade

Dissertação apresentada ao Programa de Pós-Graduação em Ciência da Computação do Departamento de Informática, Centro de Tecnologia da Universidade Estadual de Maringá, como requisito parcial para obtenção do título de Mestre em Ciência da Computação pela Banca Examinadora composta pelos membros:

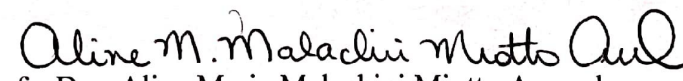
### BANCA EXAMINADORA



Prof. Dr. Diego Bertolini Gonçalves  
Universidade Tecnológica Federal do Paraná – DACOM/UTFPR-CM



Prof. Dr. Yandre Maldonado e Gomes da Costa  
Universidade Estadual de Maringá – DIN/UEM



Prof. Dra. Aline Maria Malachini Miotto Amaral  
Universidade Estadual de Maringá – DIN/UEM



Prof. Dr. Pedro Luiz de Paula Filho  
Universidade Tecnológica Federal do Paraná – UTFPR-MD

Aprovada em: 22 de fevereiro de 2019.  
Local da defesa: Sala 101, Bloco C56, *campus* da Universidade Estadual de Maringá.

## AGRADECIMENTOS

A VIDA, que me despertou a curiosidade para o conhecimento, permitindo-me não ficar à procura de chaves para portas velhas e fechadas e, sim, me permitindo construir chaves para abrir e entrar em novas porta. Iniciando em cada uma delas uma nova etapa, podendo superar defeitos, deixando para trás a ansiedade, as irritações e diversas bobagens, assim cada vez mais incrementando a minha felicidade e autorrealização, reconhecendo o valor da maior empresa do mundo, a vida.

AO MEU PAI YOUSSEF, MINHA MÃE LUZIA E MEU IRMÃO MATHEUS, reconheço imensamente todo apoio dado por vocês em toda minha vida acadêmica. Minha gratidão será eterna, amo vocês. E a nossa linda e arteira cachorrinha TEQUILA sempre nos fazendo sorrir.

AO MEU ORIENTADOR DIEGO BERTOLINI e AO MEU COORIENTADOR YANDRE MALDONADO, que dedicaram tempo e compartilharam suas experiências acadêmicas, me ajudando da melhor forma nas tomadas de decisões para construção desta dissertação.

AO PROF. WAGNER IGARASHI, pela oportunidade de mais uma vez trabalharmos juntos, no estágio em docência do mestrado, fazendo de nossas aulas uma experiência única.

AO AMIGO GUSTAVO BENNEMANN, que me ensinou muito sobre aprendizagem de máquina e também por ter dedicado várias madrugadas do seu tempo revisando meus códigos.

AO CNPq, pelo apoio financeiro concedido a este trabalho.

A TODOS, aos familiares, aos professores, aos velhos amigos e novos amigos, ambos que tive o enorme prazer em conhecer, que ouviram meus desabafos, que respeitaram meu silêncio, que partilharam conhecimento, espero que a maioria de vocês continue presente na minha vida.

# IDENTIFICAÇÃO DE ESCRITORES USANDO CNN's COM ABORDAGEM DE DISSIMILARIDADE.

## RESUMO

A identificação de escritores utilizando manuscritos tornou-se um tema importante de pesquisa para análise de documentos forenses, assim, a escrita pode ser considerada uma característica usada para distinguir uma pessoa. Existem diferentes bases de dados de escritores utilizando manuscritos e contendo diferentes alfabetos. Para atingir os objetivos deste trabalho, foram utilizadas as bases de dados: CVL e a BFL ambas sendo *single-script*, com documentos escritos no mesmo alfabeto, e a base LAMIS-MSHD que é *multi-script*, ou seja, escritas em diferentes alfabetos. Além disso, várias técnicas foram aplicadas no processo de identificação de escritores. Portanto, o objetivo deste trabalho é avaliar o desempenho da técnica de inteligência artificial conhecida como *Convolutional Neural Network* (CNN) na identificação de escritores utilizando manuscritos. Assim, será utilizada uma CNN tanto para a classificação dos escritores, quanto para extração de características que serão submetidas ao classificador SVM e também aplicado a dissimilaridade para novos experimentos. Inicialmente, foram desenvolvidos experimentos utilizando-se a abordagem tradicional de reconhecimento de padrões, baseada na engenharia de características (ou *handcrafted features*). Nestes experimentos, foi utilizado a geração de textura a partir dos documentos originais e posteriormente, foram extraídas as características com os descritores de textura LBP e LPQ. Assim, avaliado o impacto da classificação no SVM com e sem o uso da abordagem de dissimilaridade, obtendo com regras de combinação de classificadores uma decisão consensual em relação à decisão final. Depois de uma série de experimentos, a abordagem com dissimilaridade nas características obtidas por meio da CNN, apresentou resultados superiores em relação à literatura.

**Palavras-chave:** Inteligência Artificial; Rede Neural Convolutacional, Identificação de Escritores, Dissimilaridade.

# WRITER IDENTIFICATION USING CNN's WITH DISSIMILARITY FRAMEWORK.

## *ABSTRACT*

The writer identification using handwritten documents has become an important research topic in documents forensics analysis. That is because it can be used as an identifying characteristic of a person. There are several databases composed of handwritten documents available for research, using different languages and alphabets. To obtain the objectives of this work, the following databases were utilized: CVL and BFL both used for single-script, for documents written in the same alphabet and the LAMIS-MSHD database which is built for multi-script, that is, documents written with different alphabets. Besides that, several techniques were applied in writer identification process. The objective of this work is to evaluate the performance of the artificial intelligence technique known as Convolutional Neural Network (CNN) in writer identification utilizing handwritten documents. For this, a CNN will be used for the classification of the writers, also for feature extraction that will be evaluated in the SVM classifier and dissimilarity procedures are to be applied. Initially, experiments were developed using the traditional pattern recognition approach, based on feature engineering (or handcrafted features). In these experiments, the texture generation is done from the original documents and later, the features were extracted with the texture descriptors LBP and LPQ. Furthermore, the impact of the classification in the SVM was evaluated with and without the use of the dissimilarity approach, obtaining through combination rules of classifiers, a consensual decision in relation to the final decision. After a series of experiments, the approach with feature dissimilarity obtained through CNN, presented superior results in relation to the literature.

**Keywords:** Artificial Intelligence; Convolutional Neural Network, Writer identification, Dissimilarity.

## LISTA DE FIGURAS

Figura - 1.1	Amostra da base de dados CVL mostrando o efeito da troca de canetas na característica da escrita do mesmo escritor (Escritor id: 201, Documento Id: 3). Figura do trabalho de (Fiel e Sablatnig, 2013). . . . .	16
Figura - 3.1	Documento em inglês da base CVL (Kleber et al., 2013). . . . .	29
Figura - 3.2	Documento em francês da base LAMIS-MSHD (Djeddi et al., 2014). . . . .	30
Figura - 3.3	Documento em árabe da base LAMIS-MSHD (Djeddi et al., 2014). . . . .	31
Figura - 3.4	Conteúdo textual da base BFL reproduzido pelos escritores. . . . .	32
Figura - 3.5	Amostra de um documento da base BFL redigida por um dos escritores. . . . .	32
Figura - 3.6	Processo de Geração de Textura. <b>(a)</b> Alinhamento dos componentes e <b>(b)</b> geração da textura. Adaptado de (Hanusiak et al., 2012). . . . .	33
Figura - 3.7	Variações de $P$ e $R$ no LBP. Adaptado de (Mäenpää, 2003). . . . .	34
Figura - 3.8	<i>Convolutional Neural Network</i> adaptada do trabalho de (Sermanet e LeCun, 2011). . . . .	37
Figura - 3.9	Exemplo de aplicação da camada convolucional em uma imagem de dimensões $32 \times 32 \times 3$ com aplicação de 5 filtros (Li et al., 2015). . . . .	38
Figura - 3.10	Exemplo da fase de aplicação do <i>pooling</i> na imagem $9 \times 9$ , adaptado de (Hafemann, 2014). . . . .	39
Figura - 3.11	Exemplo 2 da fase de aplicação do <i>pooling</i> na imagem $8 \times 8$ , adaptado de (Li et al., 2015). . . . .	40
Figura - 3.12	Características de três diferentes classes (à esquerda), transformação com a dissimilaridade de três para duas classes (à direita) (Rivard, 2010). . . . .	42
Figura - 3.13	Distribuições no espaço de características (à esquerda) projetadas no espaço da distância pela transformação da dissimilaridade (à direita) (Rivard, 2010). . . . .	43
Figura - 4.1	Representação esquemática do método proposto para os experimentos com descritores de textura. . . . .	46
Figura - 4.2	Representação esquemática do método proposto para os experimentos com CNN. . . . .	48
Figura - 4.3	Representação da Rede Neural Convolucional utilizada. . . . .	49



Figura - 4.4	Método divisão de blocos. . . . .	50
Figura - 5.1	Características de 30 escritores da base BFL separados por cores.	59
Figura - 5.2	Características positivas (azul) e negativas (laranja) da base BFL.	60
Figura - 5.3	Características de 30 escritores da base CVL separados por cores.	62
Figura - 5.4	Características positivas (azul) e negativas (laranja) da base CVL.	62

## LISTA DE TABELAS

Tabela - 2.1	Síntese da revisão bibliográfica abordagem tradicional. . . . .	26
Tabela - 2.2	Síntese da revisão bibliográfica abordagem CNN. . . . .	27
Tabela - 3.1	Amostras utilizadas na base de dados LAMIS-MSHD . . . . .	30
Tabela - 4.1	Informações sobre a CNN utilizada. . . . .	48
Tabela - 5.1	Resultados Base CVL - Um Contra Todos . . . . .	53
Tabela - 5.2	Resultados Base LAMIS-MSHD - Um Contra Todos . . . . .	53
Tabela - 5.3	Resultados Base CVL - Dissimilaridade . . . . .	55
Tabela - 5.4	Resultados Base LAMIS-MSHD - Dissimilaridade . . . . .	55
Tabela - 5.5	Comparação tamanho da entrada experimentos na base CVL. . .	56
Tabela - 5.6	Comparação tamanho camada totalmente conectada. . . . .	57
Tabela - 5.7	Resultados CNN e SVM média de todas as amostras. . . . .	58
Tabela - 5.8	Resultados SVM média de 1/3 e 1/9 das amostras. . . . .	58
Tabela - 5.9	Resultados dissimilaridade média de 1/3 e 1/9 das amostras. . .	59
Tabela - 5.10	Resultados CNN e SVM média de todas as amostras. . . . .	60
Tabela - 5.11	Resultados SVM média de 1/3 e 1/9 das amostras. . . . .	61
Tabela - 5.12	Resultados dissimilaridade média de 1/3 e 1/9 das amostras. . .	61

## LISTA DE SIGLAS E ABREVIATURAS

- 2D DFT:** *Two-Dimensional Discrete Fourier Transform*
- BDCT:** *Bagged Discrete Cosine Transform*
- BFL:** *Brazilian Forensic Letter Database*
- BW-LBC:** *Block Wise Local Binary Count*
- CNN:** *Convolutional Neural Networks*
- CNPq:** *Conselho Nacional de Desenvolvimento Científico e Tecnológico*
- CSLBCoP:** *Center Symmetric Local Binary Co-occurrence Pattern*
- DWT+LEP:** *Discrete Wavelet Transform-based + Local Extrema Pattern*
- DWTD+LEP:** *Discrete Wavelet Transform-based Directional + Local Extrema Pattern*
- E-SVM:** *Exemplar Support Vector Machine*
- GMM:** *Gaussian Mixture Models*
- GPU:** *Graphical Processing Units*
- KNN:** *K - Nearest Neighbor*
- LBP:** *Local Binary Patterns*
- LPQ:** *Local Phase Quantization*
- LTP:** *Ternary Patterns Histograms*
- LTripD:** *Local Tri - Directional Pattern*
- MLP:** *Multi-Layer Perceptron*
- MLTPH:** *Multi-Scale Local Ternary Patterns Histograms*
- NLPR:** *National Laboratory of Pattern Recognition*
- NN:** *Nearest Neighbor*
- oBIF Columns:** *Oriented Basic Image Feature Columns*
- PCA:** *Principal Component Analysis*
- ReLU:** *Rectified Linear Units*
- RNA:** *Redes Neurais Artificiais*
- SIFT:** *Scale Invariant Feature Transform*
- SR-KDA:** *Spectral Regression Kernel Discriminant Analysis*
- SVD:** *Singular Value Decomposition*
- SVM:** *Support Vector Machine*
- t-SNE:** *t-Distributed Stochastic Neighbor Embedding*
- WD:** *Writer-Dependent*
- WI:** *Writer-Independent*
- WLSR:** *Weighted Label Smoothing Regularization*

# SUMÁRIO

<b>1</b>	<b>Introdução</b>	<b>13</b>
1.1	Motivação . . . . .	15
1.2	Desafios . . . . .	16
1.3	Objetivos . . . . .	16
1.4	Contribuições . . . . .	17
1.5	Organização do Texto . . . . .	18
<b>2</b>	<b>Estado da Arte</b>	<b>19</b>
2.1	Identificação de Escritores com <i>Handcrafted Features</i> . . . . .	19
2.2	Identificação de Escritores com <i>Deep Learning</i> . . . . .	23
2.3	Discussão . . . . .	24
<b>3</b>	<b>Fundamentação Teórica</b>	<b>28</b>
3.1	Bases de Dados . . . . .	28
3.1.1	CVL . . . . .	29
3.1.2	LAMIS-MSHD . . . . .	29
3.1.3	BFL . . . . .	31
3.2	Geração de Textura . . . . .	31
3.3	Descritores de Textura . . . . .	33
3.3.1	<i>Local Binary Pattern</i> - LBP . . . . .	33
3.3.2	<i>Local Phase Quantization</i> - LPQ . . . . .	35
3.4	<i>Deep Learning</i> . . . . .	36
3.4.1	Convolutional Neural Networks (CNNs) . . . . .	36
3.4.2	Camada Convolutacional . . . . .	37
3.4.3	Camada de Pooling . . . . .	38
3.4.4	Camada Totalmente Conectada . . . . .	39
3.4.5	Treino de CNNs . . . . .	40
3.5	Respresentação da Dissimilaridade . . . . .	41
3.6	Combinação de Classificadores . . . . .	42
3.6.1	Regra da Soma . . . . .	42
3.6.2	Regra do Produto . . . . .	44
3.6.3	Regra do Máximo . . . . .	44
3.7	Considerações . . . . .	44

<b>4</b>	<b>Método Proposto</b>	<b>45</b>
4.1	Metodologia: Utilizando Descritores de Textura . . . . .	45
4.2	Metodologia: Utilizando <i>Deep Learning</i> . . . . .	47
4.2.1	Rede neural convolucional . . . . .	47
4.2.2	Divisão e Agrupamento das Bases de Dados . . . . .	49
4.3	Considerações . . . . .	50
<b>5</b>	<b>Resultados Experimentais</b>	<b>51</b>
5.1	Abordagem Tradicional . . . . .	51
5.1.1	Um Contra Todos (SVM) . . . . .	52
5.1.2	Dissimilaridade . . . . .	54
5.2	Rede Neural Convolucional . . . . .	56
5.2.1	Experimentos utilizando a Base BFL . . . . .	57
5.2.2	Experimentos utilizando a Base CVL . . . . .	59
5.3	Discussão dos Resultados . . . . .	61
<b>6</b>	<b>Considerações Finais</b>	<b>64</b>
6.1	Trabalhos Futuros . . . . .	65
	<b>REFERÊNCIAS</b>	<b>66</b>

---

# Introdução

---

As atividades de reconhecimento de padrões automatizadas tentam alcançar desempenho próximo ou superior ao de seres humanos em diferentes áreas de atuação. Esse campo vem crescendo rapidamente no decorrer dos anos e um dos fatores se deve ao fato do poder computacional estar em constante crescimento. Assim, a tarefa de reconhecimento de padrões tem sido aplicada nas mais diversas áreas como: na medicina, com o diagnóstico de doenças (Mangasarian et al., 1990; Wong e Bressler, 2016), na indústria, com o controle de processos (Bradski et al., 2002; Kahng et al., 2018), no comércio, com a previsão de dados econômicos (Li e Chen, 2014; Trippi e Turban, 1992), no campo da biometria, com o reconhecimento de face (Ahonen et al., 2006; Parkhi et al., 2015), dentre muitos outros (Briand et al., 1992; Ding et al., 2015; Martin e Kim, 1998). O foco desta dissertação é a utilização de sistemas biométricos para a identificação de escritores por meio de textos manuscritos.

Jain et al. (2006) definem que um sistema biométrico além de ser um problema de reconhecimento de padrões, também pode ser uma tecnologia com potencial de tornar a sociedade mais segura e também reduzindo fraudes, caso seja utilizada cuidadosamente, isso por meio de três funcionalidades: verificação, identificação e monitoramento.

1. **Verificação:** (A pessoa realmente é quem ela afirma ser?). O sistema de verificação aceita ou rejeita uma solicitação, por meio de uma comparação realizada entre o padrão oferecido e o padrão associado com a identidade solicitada. Assim, para saber se uma certa pessoa escreveu ou não um certo manuscrito, tem-se uma comparação de um para um (1 : 1). (Jain et al., 2006).

2. **Identificação:** (A pessoa está na base de dados?). Dada uma amostra de entrada, o processo de identificação determina se essa amostra está associada a qualquer outra amostra da base de dados. Assim, pode-se determinar o escritor de um manuscrito a partir de um conjunto de escritores, por meio de uma busca de um para muitos ( $1 : N$ ). (Jain et al., 2006).
3. **Monitoramento:** (A pessoa está sendo procurada?). As aplicações de monitoramento determinam se uma pessoa está em uma lista de procurados. Exemplos de aplicações podem ser vistos em segurança de aeroportos, de eventos públicos e em outras aplicações de vigilância. É possível monitorar pessoas por meio de manuscritos também. O monitoramento é uma comparação  $1 : N$ , porém,  $N$  neste caso é uma lista restrita. (Jain et al., 2006).

A aquisição de manuscritos é classificada em dois grupos: *on-line* (dinâmica) ou *off-line* (estática). Manuscritos *on-line* são capturados por meio do uso de *tablets* ou mesas digitalizadoras. Com esta abordagem é possível alcançar melhores desempenhos, pois é possível adquirir informações precisas dos manuscritos (como posição, velocidade e pressão). Já manuscritos *off-line* são obtidos pela digitalização de imagens de manuscritos. Tais métodos de classificação usados em sistemas *on-line* e *off-line* mostram variações de desempenho na classificação (Yilmaz et al., 2011). Neste trabalho será utilizada a abordagem *off-line* devido ao fato de documentos manuscritos em papel serem utilizados como provas legais em questões judiciais.

Bases de dados de manuscritos *off-line* normalmente são definidas como *single-script*, pois utilizam apenas um estilo de escrita. Por outro lado, existem bases de dados *multi-script* que utilizam mais de um estilo de escrita, ou seja, o mesmo escritor produz documentos em diferentes alfabetos, por exemplo, o alfabeto romano e o árabe.

O texto de um manuscrito pode ser classificado em dependente ou independente. Assim o texto-dependente ou cópia, é quando todos os escritores transcrevem um mesmo texto, por exemplo: o conjunto de escritores de uma base de dados copiaram uma determinada carta. Como ele requer o mesmo conteúdo para todos os escritores, esse método não é apto para algumas aplicações práticas. No texto-independente ou livre, os escritores não precisam escrever um texto padrão, escrevem um texto próprio, não existindo um número determinado de linhas ou palavras. Embora ele tenha uma maior aplicabilidade, o texto-independente possui taxas de acertos menores que as do texto-dependente (Idicula, 2011). Isso porque as amostras muitas vezes possuem pouco conteúdo escrito, diferente do texto-dependente na qual todos os escritores cedem um vasto conteúdo manuscrito.

O processo de identificação de escritores pode empregar duas diferentes abordagens para a classificação, sendo descritas como abordagem escritor-dependente e abordagem escritor-independente. A abordagem escritor-dependente, *writer-dependent (WD)*, costuma produzir bons resultados. Porém, para cada novo escritor inserido, o modelo deve ser reconstruído. Outra questão importante é que existe a necessidade de uma grande quantidade de dados para treinar um modelo confiável. Essa abordagem tem sido amplamente utilizada para verificação de assinaturas (Pavelec et al., 2008).

Uma alternativa para a abordagem escritor-dependente é a escritor-independente, *writer-independent (WI)*, essa é uma estratégia que consiste em treinar um modelo global que é baseada na abordagem de exame de documentos forenses questionada e classificando a escrita em termos de autenticidade, podendo ser definida como genuína ou falsificada (Pavelec et al., 2008).

A proposta para esta dissertação aqui apresentada é a identificação de escritores utilizando manuscritos *off-line*, por meio de características obtidas com uma *convolutional neural network* (CNN) em conjunto com uma abordagem escritor-independente. Ainda será investigada a hipótese de classificação dos escritores de manuscritos em alfabetos diferentes (*multi-script*).

## 1.1 Motivação

A análise de manuscritos feita por peritos da Ciência Forense, permite identificar a possibilidade de fraudes em documentos. Manuscritos podem ser considerados elementos chave na resolução de crimes, na investigação de atentados terroristas, cartas de falso suicídio entre outros (Bertolini, 2014). Assim, a motivação computacional deste trabalho é analisar alternativas de soluções que sejam úteis na identificação de escritores a partir de manuscritos.

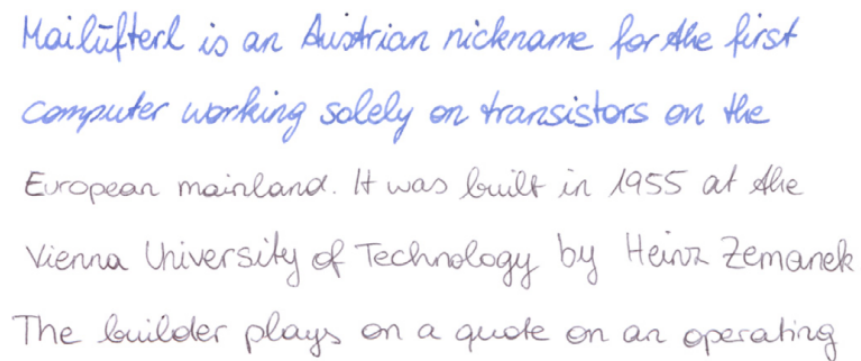
Outra motivação é que as técnicas tradicionais de aprendizagem de máquina costumam requerer conhecimentos de humanos para extração de características, em cada domínio específico, a fim de serem utilizados em tarefas de classificação (Hafemann, 2014). Assim, o uso de técnicas de *deep learning* que não dependem de extratores de características de cor, forma, textura entre outros sejam definidos por especialistas, facilita a criação de modelos de aprendizagem de máquina eficazes.

Dessa maneira, o método proposto para solucionar o problema de identificação de escritores, em documentos *single-script* ou *multi-script*, utilizando características extraídas por uma CNN, poderá ser comparado com métodos da abordagem tradicional de reconhecimento de padrões.



## 1.2 Desafios

Um dos grandes desafios da identificação de escritores a partir de manuscritos está relacionado com a característica de escrita. Cada pessoa possui uma característica única e duas pessoas não escrevem da mesma maneira. Além disso, uma pessoa não escreve exatamente da mesma maneira duas vezes (Fiel e Sablatnig, 2013). Para complicar ainda mais, a caneta utilizada na escrita também afeta a identificação de escritores, conforme pode ser observado na Figura - 1.1, que mostra um exemplo extraído da base de dados CVL (Kleber et al., 2013). Nela as duas primeiras linhas foram escritas com uma caneta e as três linhas seguintes foram escritas com outra caneta. Assim, extrair características relevantes indiferentemente da escrita e do alfabeto utilizado nas amostras, torna-se um desafio.



*Mailüfterl is an Austrian nickname for the first  
computer working solely on transistors on the  
European mainland. It was built in 1955 at the  
Vienna University of Technology by Heinz Zemanek.  
The builder plays on a quote on an operating*

**Figura 1.1:** Amostra da base de dados CVL mostrando o efeito da troca de canetas na característica da escrita do mesmo escritor (Escritor id: 201, Documento Id: 3). Figura do trabalho de (Fiel e Sablatnig, 2013).

O número de classes existentes em sistemas de identificação de escritores também é um desafio. Um problema na identificação de escritores é a diferença intraclasse, ocorre quando amostras do mesmo escritor são muito distintas, por exemplo, escrita caixa alta e escrita caixa baixa no mesmo manuscrito. Tem-se também a similaridade interclasses, ocorre quando as escritas de escritores diferentes são muito parecidas, também sendo um desafio. Assim, a dificuldade devido à similaridade interclasses tem seu aumento proporcional ao número de escritores (Pekalska e Duin, 2002).

## 1.3 Objetivos

Esse trabalho tem como objetivo principal elaborar um método capaz de realizar a tarefa de identificação de escritores por meio de documentos manuscritos. A abordagem utilizada

está embasada em quatro etapas principais, sendo elas: a etapa de pré-processamento das imagens digitalizadas, a geração de textura, a utilização de uma CNN como extrator de características e a classificação. Para atingir o objetivo principal deve-se destacar os seguintes objetivos secundários:

1. Avaliar o impacto da abordagem escritor-independente  $\times$  escritor-dependente usando descritores de textura e o classificador *Support Vector Machine* (SVM);
2. Avaliar o impacto dessas abordagens usando *single-script* e *multi-script*;
3. Avaliar o tamanho da imagem dada como entrada para uma CNN. A maioria dos trabalhos com tarefas relacionadas a *deep learning* obtiveram bons resultados utilizando imagens de tamanhos pequenos. Exemplos de bases de dados com o tamanho das imagens dada com entrada: MNIST ( $28 \times 28$  pixels), STL-10 ( $96 \times 96$ ), Norb ( $108 \times 108$ ), Cifar-10 e Cifar-100 ( $32 \times 32$ ) (Hafemann, 2014);
4. Pesquisar na literatura e definir um modelo eficiente de rede neural convolucional para o problema de identificação de escritores;
5. Utilizar a CNN como extrator de características, bem como a utilização dessas características no classificador SVM;
6. Avaliar o impacto das dimensões das camadas na CNN, assim como, possíveis configurações dos hiperparâmetros da rede;
7. Aplicar a abordagem da dissimilaridade usando as características extraídas com a CNN.

## 1.4 Contribuições

O desenvolvimento deste trabalho proporcionou algumas contribuições importantes no contexto de identificação de escritores. A modelagem de uma arquitetura de rede neural convolucional que pode ser usada tanto para a classificação quanto para a extração de características de escritores. Ainda, destaca-se a proposta de uma abordagem que utiliza as características extraídas com a CNN, posteriormente aplicada a técnica de dissimilaridade, que alcançou as melhores taxas dessa dissertação. Ainda, durante o processo de desenvolvimento desta dissertação dois artigos foram publicados (Bertolini et al., 2017; Helal et al., 2017).

## **1.5 Organização do Texto**

Esta dissertação está dividida em seis capítulos, organizados como segue: A Introdução no capítulo 1, em que é descrita uma contextualização sobre manuscritos, os principais desafios, objetivos e contribuições do trabalho. O Estado da Arte no capítulo 2, com trabalhos relacionados a pesquisa. A Fundamentação Teórica no capítulo 3, abordando a teoria base para o desenvolvimento do trabalho. O Método Proposto no capítulo 4, explicando como ele é utilizado no desenvolvimento deste trabalho. Os Resultados no capítulo 5, demonstrando os resultados de experimentos obtidos ao longo desta pesquisa. Por fim, as Considerações Finais são feitas e são apresentadas propostas para trabalhos futuros no capítulo 6.

---

## Estado da Arte

---

Neste capítulo é apresentada a revisão da literatura, nela os trabalhos descritos são relacionados ao processo de identificação de escritores por meio de documentos manuscritos. A divisão das seções deste capítulo é feita de modo que a Seção 2.1 agrupa os trabalhos que empregam a abordagem tradicional de classificação, ou seja, extraem características manualmente (engenharia de características) e a Seção 2.2 agrupa os trabalhos que utilizam *Deep Learning*. Em geral nas duas seções, são citadas as principais informações de cada trabalho, como: base de dados utilizada, quantidade de escritores, classificador, medidas de avaliação e principais resultados. Por fim, a Seção 2.3 encerra a revisão bibliográfica resumando os trabalhos descritos nas seções anteriores e apresenta algumas considerações sobre o que foi levantado.

### 2.1 Identificação de Escritores com *Handcrafted Features*

Fiel e Sablatnig (2013) empregam características de textura para identificação de escritores em seu trabalho. O sistema proposto consiste em inicialmente computar características locais normalizando a imagem por meio do *Scale Invariant Feature Transform* (SIFT). As características de um conjunto de treinamento são agrupadas utilizando o *Gaussian Mixture Models* (GMM) que é usado como vocabulário, permitindo calcular o vetor *Fisher* de cada imagem da base. A classificação foi realizada usando a regra de distância mínima. As bases de dados empregadas foram a CVL com 309 escritores e ICDAR 2011 com 26

escritores. Os autores apresentam taxas de identificação de 97,8% e 91,3% para as bases CVL e ICDAR 2011, respectivamente.

A abordagem proposta por Bertolini et al. (2013) utiliza um processo de geração de textura na etapa de pré-processamento. Desta forma, são reduzidos custos computacionais e possíveis erros, pois espaços em branco são diminuídos e alguns caracteres especiais são segmentados (retirados acentos e pontos). Posteriormente, o processo de extração de características utiliza os descritores *Local Binary Patterns* (LBP) e *Local Phase Quantization* (LPQ). Foi utilizada uma abordagem de dissimilaridade, a qual mapeia um problema de  $n$  classes em duas classes. Os modelos foram gerados com o *Support Vector Machine* (SVM). Os autores avaliaram o método proposto em duas bases de dados, a BFL com 315 escritores, que foi dividida em 200 escritores para treino e 115 para teste e a IAM, utilizando 650 escritores, sendo 410 escritores no conjunto de treinamento e 240 no conjunto de teste. Os melhores resultados apresentados no trabalho foram obtidos utilizando o descritor de textura LPQ com taxa de identificação para Top-1 de 99,2% e 96,7% nas bases BFL e IAM respectivamente.

O trabalho Newell e Griffin (2014) utiliza o descritor *Oriented Basic Image Feature Columns* (*oBIF Columns*) para identificação dos escritores. É utilizada também a hipótese de *Delta encoding* que faz a junção de *oBIF Columns*. A classificação foi feita utilizando *Nearest Neighbor* (NN) com a distância euclidiana. A base de dados utilizada foi a IAM com 301 escritores, a base ICDAR 2011 com 54 e ICDAR 2012 com 204 escritores. A acurácia com o descritor *oBIF Columns* para a base dados IAM foi 99%. Para outras bases, foi utilizado *Delta encoding*, a base ICDAR 2011 teve sua taxa de acerto em 100% e a base ICDAR 2012 teve taxa de 95,3%.

Khan et al. (2016) propõem o uso do *Multi-Scale Local Ternary Patterns Histograms* (MLTPH) que é uma variação do *Local Ternary Patterns* (LTP) como descritor de textura. A proposta dos autores foi de aplicar o MLTPH em duas bases de dados, uma com manuscritos em Inglês (IAM), e outra em com manuscrito árabe (AHTID/MW). Foi utilizado um conjunto de classificadores, *Spectral Regression Kernel Discriminant Analysis* (SR-KDA), junto com a regra do voto majoritário para a classificação. Nesse trabalho o método foi aplicado isoladamente em cada uma das bases de dados, para a base IAM os autores utilizam na identificação 100 escritores em um conjunto bem definido em treino, teste e validação, a taxa de acerto utilizando o Top-1 para as escritas em inglês foi de 99,4%, por outro lado a base AHTID/MW com 53 escritores foi dividida em dois terços para treino e o restante para teste, sua taxa de acerto utilizando o Top-1 foi de 84,6%.

O trabalho de Hannad et al. (2016) realiza uma binarização dos documentos manuscritos para posteriormente encontrar os componentes conexos. A partir dos frag-

mentos segmentados os autores empregam descritores de textura *Local Ternary Patterns* (LTP), *Local Binary Patterns* (LBP) e *Local Phase Quantization* (LPQ). Em seguida são calculadas medidas de distâncias entre os vetores de características dos escritores (dissimilaridade). As bases de dados utilizadas foram a IAM, que possui 657 escritores, sendo que 60% deles foram utilizados para treinamento e 40% para teste e a base em Árabe IFN/ENIT contendo 411 escritores, porém, cada amostra de texto manuscrito dos escritores foi reduzida, foram selecionadas de forma aleatória 30 palavras a fim de gerar 2-3 linhas de texto. A identificação foi feita com base no critério de vizinho mais próximo. Foram avaliados os Top-1, Top-5 e Top-10, para cada descritor de textura nas diferentes bases de dados. Os melhores resultados apresentados com Top-1 foram obtidos utilizando o LPQ, as taxas de acerto foram de 94,89% para base IFN/ENIT e de 89,54% para IAM.

O trabalho de Khan et al. (2017), propõe a abordagem *Bagged Discrete Cosine Transform* (BDCT) para identificação de escritores. O propósito da utilização do DCT foi devido a seus coeficientes que são normalmente robustos a distorções que podem ocorrer durante o processo da escrita ou de varredura (ruído, borrão, mudança no contraste, etc). O BDCT foi aplicado em quatro bases de dados, sendo duas em inglês a IAM com 657 escritores e CVL com 310 escritores (somente as amostras em inglês foram utilizadas), e outras duas bases em árabe: AHTID/MW com 53 escritores e a IFN/ENIT com 411 escritores. O sistema proposto foi classificado com *Spectral Regression Kernel Discriminant Analysis* (SR-KDA) com o voto majoritário para a combinação de classificadores e o Top-1 que apresentou acurácia de 97,2% para a base IAM, para a base CVL reportam acurácia de 99,6%, para as bases AHTID/MW e IFN/ENIT 71,6% e 76,0%, respectivamente.

Christlein et al. (2017) propõem um método robusto para identificação de escritores *off-line* chamado RootSIFT. A extração de *features* usada no trabalho consistiu na utilização do descritor *Scale Invariant Feature SIFT* em conjunto com o *Gaussian Mixture Model* (GMM) e para classificação é utilizado *Exemplar Support Vector Machine* (E-SVM). As bases de dados utilizadas para validar o método foram ICDAR 2013 com 350 escritores divididos em 100 para treino e 250 para teste, CVL com 310 escritores divididos em 27 para treino e 283 para teste, e a base KHATT com 1000 escritores dividida em 70% para treino, 15% para teste e 15% para validação. Os resultados deste trabalho para a base de dados ICDAR 2013 com Top-1 de 100% para as amostras em grego e de 99% para as amostras em Inglês, a base de dados CVL teve seu Top-1 de 99,2% e a base KHATT com Top-1 de 99,5%.

O trabalho Mohammadi et al. (2018) realiza a identificação de escritores em bases de dados de diversos idiomas. Os autores utilizam uma abordagem de mineração de imagens

para textos *off-line* e independentes. Na parte de mineração de imagens da abordagem proposta são utilizadas certas técnicas de filtro de Gabor e XGabor, incluindo classificador de *Support Vector Machine* (SVM) junto com *K - Nearest Neighbor* (KNN) e algoritmos genéticos. Para avaliar o método e mostrar seu desempenho em diferentes idiomas, as bases de dados utilizadas foram: CASIA para chinês com 50 escritores com 10 páginas de amostra para cada escritor, um conjunto da base IAM em inglês com 31 escritores e 9 amostras cada, dois conjuntos de dados para caligrafia Kannada com a base KDTH usando 54 escritores com 8 amostras cada e por fim a base PDS100 com 50 escritores e 10 amostras cada. Os conjuntos foram divididos em 90% para treino e 10% para teste, os resultados dos experimentos mostraram que o método possui acurácia de 99% para as bases CASIA e KDTH e acurácia de 100% para as bases IAM e PDS100.

Singh et al. (2018) empregam uma abordagem com base em descritores de textura para a identificação de escritores em diversas bases de dados. Os descritores *Local Binary Pattern* (LBP), *Local Phase Quantization* (LPQ), *Discrete Wavelet Transform-based e Local Extrema Pattern* (DWT+LEP), *Discrete Wavelet Transform-based Directional e Local Extrema Pattern* (DWTD+LEP), *Center Symmetric Local Binary Co-occurrence Pattern* (CSLBCoP), e *Local Tri - Directional Pattern* (LTriDP) são utilizado para extrair características das bases IAM com 657 escritores, KHATT com 1000 escritores e DEVNAGRI (*scripts* indianos populares) com 60 escritores. Para classificação foram utilizados o *Support Vector Machine* (SVM) e o *K - Nearest Neighbor* (KNN). As melhores taxas de acerto foram: 97,62% para a base de dados IAM usando o LBP e SVM, para KHATT o resultado foi de 95,60% usando LBP e KNN e por fim a acurácia de 65,80% para os *scripts* DEVNAGRI com o LPQ e KNN.

Chahi et al. (2018) utilizam texto independente na identificação de escritores. Os autores propõem um descritor conceitualmente simples com boa taxa de identificação chamado *Block Wise Local Binary Count* (BW-LBC). O método BW-LBC proposto caracteriza o estilo de escrita de cada escritor por um conjunto de histogramas calculados a partir de todos os componentes conectados na escrita. Cada histograma é construído calculando a distribuição de ocorrência de pixels correspondentes à escrita dentro de pequenos blocos em cada componente conectado extraído e recortado da amostra do manuscrito. As amostras são classificadas de acordo com as características dos histogramas utilizando a regra do *Nearest-Neighbor* (1-NN) usando distância de Hamming. Os experimentos utilizam as bases de dados IFN/ENIT com 411 escritores com 60% para treino e 40% teste, AHTID/MW com 53 escritores com 4 amostras sendo 3 para treino e uma para teste, CVL com 310 escritores com 4 amostras no treino e 3 para teste e a base IAM com

657 escritores com 60% para treino e 40% teste. Os resultados para a base IFN/ENIT, AHTID/MW, CVL e IAM foram de 96,47%, 99,53%, 98,38% e 88,99% respectivamente.

## 2.2 Identificação de Escritores com *Deep Learning*

No trabalho de Hafemann (2014) foram consideradas duas bases de dados para a identificação de escritores: a *Brazilian Forensic Letter Database* (BFL) com 315 escritores e a base IAM com 650 escritores. O trabalho utilizou a mesma representação de textura feita no trabalho de Bertolini et al. (2013). Para a base BFL foram selecionados de forma aleatória 115 escritores dos 315 originais. A partir da geração de textura, foram criados blocos de tamanho  $256 \times 256$ . Na base IAM foram selecionadas 240 classes aleatoriamente, e os blocos de textura com tamanho de  $128 \times 256$ . Os conjuntos foram divididos em três: treino, teste e validação, em ambas as bases. Utilizando um filtro  $7 \times 7$  com *patch size* de  $48 \times 48$ , foram reportadas taxas de identificação usando a CNN de 91,67% na base IAM e de 95,65% para a BFL.

O trabalho de Yang et al. (2016) utilizou escrita *on-line* e texto-independente para fazer um sistema de identificação de escritores, chamado *DeepWriterID* utilizando *Convolutional Neural Networks* (CNN). O método propôs o *DropSegment*, que remove de forma aleatória vários segmentos de caracteres de uma amostra original de manuscritos, mantendo a informação de identificação nela contida. O método foi projetado para gerar dados de forma artificial por meio da técnica de *data augmentation*, melhorando a aplicabilidade generalizada da CNN e para evitar o *overfitting* do modelo. Os experimentos foram conduzidos utilizando o banco de dados de manuscritos *National Laboratory of Pattern Recognition* (NLPR). As taxas de identificação para textos em chinês foram de 95,75% e para textos em inglês de 98,51%.

O trabalho de Tang e Wu (2016) utiliza texto independente para identificação de escritores *off-line*. O método consiste em um grande número de imagens para extração de características utilizando *Convolutional Neural Network* (CNN), com base nos dados extraídos por meio da CNN é utilizado o método de junção Bayesiana para a identificação dos manuscritos. O método proposto é testado em duas bases de dados, a base ICDAR2013 com 350 escritores e cada um com quatro amostras, sendo 100 para treino e 250 para teste e a base CVL com 310 escritores sendo 27 com 7 amostras para treino e 283 com 5 amostras de texto para teste. Os resultados foram de 99% para a base de dados ICDAR2013 e de 99,7% para a base de dados CVL.

O trabalho de Xing e Qiao (2016) utiliza texto independente na identificação de escritores. Os autores do trabalho propõem um método chamado *DeepWriter*, que é



uma *Convolutional Neural Network* (CNN) com abordagem *multi-stream*, ou seja, a CNN recebe como entrada duas imagens, cada uma delas vai para um rede, o treinamento das imagens é feito em paralelo, assim, após finalizados os treinos ambos resultados se juntam retornando apenas uma saída. As bases de dados utilizadas no método é a base IAM com 657 e 301 escritores com amostras em inglês e a base HWDB com 300 escritores com suas amostras dos documentos em chinês. As amostras foram divididas em treino, teste e validação nas proporções 4:1:1 respectivamente. Os resultados dos experimentos para a base IAM com 657 escritores têm uma taxa de acerto de 97,03%, com 301 escritores de 99,01% e para a base HWDB uma acurácia de 93,85%.

Chen et al. (2018) utilizaram o aprendizado semi-supervisionado para identificação de escritores *off-line*, foram treinados dados extras não rotulados (dados de outras bases) e os dados rotulados originais simultaneamente. Especificamente, o método proposto *Weighted Label Smoothing Regularization* (WLSR), que atribuiu a distribuição de rótulos uniforme ponderada aos dados extra não rotulados. Junto aos WLSR foi utilizada uma *Convolutional Neural Network* (CNN), que permitiu aprender recursos mais discriminativos para representar as propriedades de diferentes estilos de escrita. Por meio do método as bases de dados utilizadas foram: ICDAR2013 com 350 escritores, ficou com 100 para treino e 250 para teste, nesta base foram utilizadas as amostras em grego e inglês, a base CVL com 310 escritores, ficou com 27 para treino e 283 para teste, já a base IAM com aproximadamente 400 escritores, teve seus documentos adicionados junto com ao treino da base CVL. Os resultados mostraram que o aprendizado de recursos semi-supervisionado mostrou bom desempenho para a base ICDAR2013 o Top-1 foi de 96,6% e para a base CVL de 99,2%.

O trabalho de Nguyen et al. (2018) utiliza uma abordagem de texto independente para identificação de escritores. O método proposto utiliza uma CNN que é utilizada para treinar e classificar amostras das bases de dados JEITA-HP com manuscritos de 100 escritores em japonês e as bases de dados Firemaker e IAM utilizadas em conjunto com o total de 900 escritores em inglês. A base de dados JEITA-HP teve sua taxa em 99,97%, a base FIREMAKER com os 250 escritores teve um Top-1 de 92,38% de acerto e a base IAM com 650 escritores teve seu Top-1 em 90,12% e juntas o método proposto teve o Top-1 em 91,81%.

## 2.3 Discussão

Esse capítulo apresentou um breve estado da arte descrevendo as principais características de cada trabalho na tarefa de identificação de escritores utilizando manuscritos, mostrando

de forma resumida nas Tabelas - 2.1 e 2.2, nas quais os trabalhos são apresentados em ordem cronológica. Nelas são apresentadas, as bases de dados utilizadas, a quantidade de escritores, qual o método de classificação e o resultado de identificação. Contudo, é difícil fazer uma comparação direta entre os resultados dos trabalhos, pois na maioria dos casos as bases de dados são diferentes, sendo escritas usando alfabetos diferentes. Outra dificuldade é o uso de subconjuntos de dados, de forma que não é possível afirmar que as taxas apresentadas para aquele subconjunto se mantém caso seja utilizada a base toda. Os conjuntos de treinamento e teste também não são claramente identificados em diversos trabalhos.

**Tabela 2.1:** Síntese da revisão bibliográfica abordagem tradicional.

Ref.	Base	Escritores	Classificação	Identificação
Fiel e Sablatnig (2013)	CVL	309	Regra da Distância Mínima	97,80 %
	ICDAR 2011	26		91,30 %
Bertolini et al. (2013)	BFL	315	SVM	99,20 %
	IAM	650		96,70 %
Newell e Griffin (2014)	IAM	301	Vizinho Próximo	99%
	ICDAR 2011	54		100%
	ICDAR 2012	204		95,3%
Khan et al. (2016)	IAM	100	SR-KDA	99,40 %
	AHTID/MW	53		84,60 %
Hannad et al. (2016)	IAM	657	Vizinho Próximo	89,54 %
	IFN/ENIT	411		94,89 %
Khan et al. (2017)	IAM	657	SR-KDA	97,20 %
	CVL	310		99,60 %
	AHTID/MW	53		71,60 %
	IFN/ENIT	411		76,00 %
Christlein et al. (2017)	ICDAR2013 - Grego - Inglês	350	E-SVM	100% 99%
	CVL	310		99,2%
	KHATT	1000		99,5%
Mohammadi et al. (2018)	CASIA	50	SVM + KNN	99%
	IAM	31		100%
	KDTH	54		99%
	PDS100	50		100%
Singh et al. (2018)	IAM	657	SVM	95,60%
	KHATT	1000	KNN	97,62%
	DEVNAGRI	60	KNN	65,80%
Chahi et al. (2018)	IAM	657	Vizinho Próximo	88,99%
	CVL	310		98,38%
	AHTID/MW	53		99,53%
	IFN/ENIT	411		96,47%

**Tabela 2.2:** Síntese da revisão bibliográfica abordagem CNN.

<b>Ref.</b>	<b>Base</b>	<b>Escritores</b>	<b>Classificação</b>	<b>Identificação</b>
Hafemann (2014)	BFL	315	CNN	95,65%
	IAM	650		91,67%
Yang et al. (2016)	NLPR - CN	-	CNN	95,75%
	NLPR - EN	-		98,51%
Tang e Wu (2016)	ICDAR 2013	350	CNN	99%
	CVL	310		99,7%
Xing e Qiao (2016)	IAM	657	CNN	97,03%
	IAM	301		99,01%
	HWDB	300		93,85%
Chen et al. (2018)	ICDAR 2013	350	CNN	96,6%
	CVL + IAM	310 + 400		99,2%
Nguyen et al. (2018)	JEITA-HP	100	CNN	99,97%
	IAM	650		90,12%
	IAM + FIREMAKER	900		91,81%

---

## Fundamentação Teórica

---

Neste capítulo, são apresentadas as bases de dados utilizadas nos experimentos e também discutidas as principais técnicas computacionais empregadas no trabalho, tendo como objetivo dar suporte ao leitor, contribuindo para um melhor entendimento. O capítulo está dividido em sete seções. A Seção 3.1 apresenta as bases de dados; a Seção 3.2 mostra o processo de textura; a Seção 3.3 apresenta os descritores de textura LBP e LPQ, utilizados para os experimentos com *handcrafted features*. Em seguida, a Seção 3.4 apresenta conceitos de *deep learning* que norteiam a metodologia. A Seção 3.5 apresenta conceitos de dissimilaridade. A Seção 3.6 apresenta a combinação de classificadores. Finalmente, a Seção 3.7 faz as considerações finais deste capítulo.

### 3.1 Bases de Dados

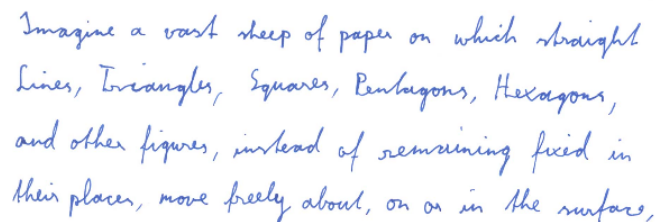
Atualmente, existe um grande número de bases de dados de escritores utilizando manuscritos como: CEDAR (Arora et al., 2002), QUWI (Al Máadeed et al., 2012), RIMES (Grosicki et al., 2008), IFN/ENIT (Pechwitz et al., 2002), KHATT (Mahmoud et al., 2012), IAM (Marti e Bunke, 2002), etc. Para a abordagem usando características (*handcrafted*), foram utilizadas duas bases de dados, sendo primeiro a base CVL proposta por (Kleber et al., 2013), que é *single-script* com dois idiomas: inglês e alemão. A segunda base é a LAMIS-MSHD proposta por (Djeddi et al., 2014), que é *multi-script*, ou seja, possui cartas em dois alfabetos distintos, sendo o alfabeto latim com documentos em francês e o alfabeto árabe. Para a abordagem com *deep learning* também foi utilizada a base CVL e a base BFL proposta por (Baranoski et al., 2007), posteriormente estendida no trabalho de (AMARAL, 2014), indo 315 para 600 escritores.

### 3.1.1 CVL

A base de dados CVL <sup>1</sup> consiste em imagens coloridas com resolução de 300 *dpi*, cujo conteúdo são cópias (texto-dependente) de sete textos manuscritos com letra cursiva, composta por manuscritos de 311 escritores. A base possui amostras de documentos em alemão e inglês, foi composta a partir de textos de obras literárias. Foram utilizadas amostras dos seguintes textos:

- Texto-1 Edwin A. Abbot - Flatland: A Romance of Many Dimension (90 palavras).
- Texto-2 William Shakespeare - Mac Beth (47 palavras).
- Texto-3 Wikipedia - Mailufterl (74 palavras).
- Texto-4 Charles Darwin - Origin of Species (52 palavras).
- Texto-5 Johann Wolfgang von Goethe - Faust. Eine Tragodie (50 palavras).
- Texto-6 Oscar Wilde - The Picture of Dorian Gray (65 palavras).
- Texto-7 Edgar Allan Poe - The Fall of the House of Usher (73 palavras).

A Figura - 3.1 apresenta um documento manuscrito da base CVL.



*Imagine a vast sheet of paper on which straight lines, Triangles, Squares, Pentagons, Hexagons, and other figures, instead of remaining fixed in their places, move freely about, on or in the surface,*

**Figura 3.1:** Documento em inglês da base CVL (Kleber et al., 2013).

### 3.1.2 LAMIS-MSHD

A base de dados LAMIS-MSHD <sup>2</sup> consiste em um conjunto de 13 documentos manuscritos, incluindo seis textos em árabe, seis textos em francês e um com dígitos. Todos os documentos manuscritos foram escaneados com uma resolução de 300 *dpi* e coloridos.

<sup>1</sup>Download *database* disponível em: <<http://bit.ly/download-CVL>>

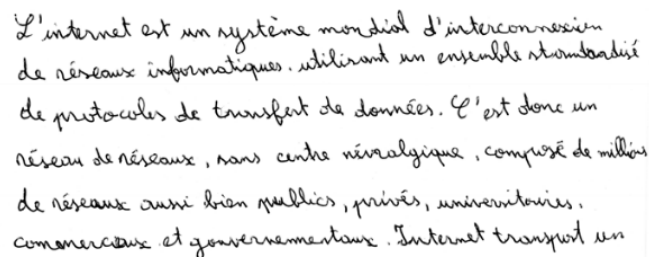
<sup>2</sup>Download *database* disponível em: <<http://bit.ly/download-LAMIS>>

Esta base de dados foi projetada visando principalmente a identificação e verificação de escritores em um ambiente de *multi-scripts*, ou seja, o escritor escreveu o mesmo manuscrito em diferentes alfabetos. A base de dados foi formada por manuscritos de 100 escritores. O conteúdo da base de dados pode ser observado na Tabela - 3.1.

**Tabela 3.1:** Amostras utilizadas na base de dados LAMIS-MSHD

Texto ID	Assunto	Escrita	Contador
1	Dígitos	Árabe	211 Dígitos
2	Medicina e Saúde	Árabe	123 Palavras
3	Economia e Negócios	Árabe	88 Palavras
4	Arte e Cultura	Árabe	109 Palavras
5	Ecologia e Meio Ambiente	Árabe	152 Palavras
6	Esportes	Árabe	128 Palavras
7	Processamento de Dados	Árabe	147 Palavras
8	Medicina e Saúde	Francês	124 Palavras
9	Economia e Negócios	Francês	109 Palavras
10	Arte e Cultura	Francês	97 Palavras
11	Ecologia e Meio Ambiente	Francês	104 Palavras
12	Esportes	Francês	94 Palavras
13	Processamento de Dados	Francês	98 Palavras

A Figura - 3.2 apresenta um trecho do manuscrito da base LAMIS-MSHD em francês e a Figura - 3.3 em árabe.



L'internet est un système mondial d'interconnexion de réseaux informatiques, utilisant un ensemble standardisé de protocoles de transfert de données. C'est donc un réseau de réseaux, sans centre névralgique, composé de milliers de réseaux aussi bien publics, privés, universitaires, commerciaux et gouvernementaux. Internet transporte un

**Figura 3.2:** Documento em francês da base LAMIS-MSHD (Djeddi et al., 2014).

الكشف جاحنون كديون أنة من الممكن قلب الأضرار الناجمة عن الإصابة  
بمرض الزهايمر من خلال اللجوء الى تقنية التحفيز الكهربائي العميق .  
ونقل موقع نيو ساينتست البريطاني عن أندرس لورانق من مستشفى  
تورنمو قوله انه مرض الزهايمر يسبب بتقلص الدماغ وضمورها منطقة  
الضمير . وأشار الى انه وفريقه لخصوا

**Figura 3.3:** Documento em árabe da base LAMIS-MSHD (Djeddi et al., 2014).

### 3.1.3 BFL

A base de dados *Brazilian Forensic Letter Database* (BFL) <sup>3</sup> utilizada neste trabalho foi a versão composta por manuscritos produzidos por 315 escritores distintos, sendo que cada um deles possui 3 amostras, totalizando 945 documentos. Cada imagem foi digitalizada com resolução de 300 *dpi* e com 256 níveis de cinza (Freitas et al., 2008). Os textos dos documentos foram escritos em português, composta com todas as letras do alfabeto latino em maiúsculo e minúsculo, além de particularidades da língua (á, ç, é, à, etc.) e números entre 0 (zero) e 9 (nove), em diferentes combinações (Justino, 2002).

A base BFL se refere a uma base de texto-dependente, em que os escritores escrevem o mesmo conteúdo textual. Segundo os autores dela, foram obedecidas todas as regras de transcrição, como: uso de caneta esferográfica de cor azul ou preta e a escrita do texto sem auxílio de linhas-guia (Bertolini, 2014).

A Figura - 3.4 apresenta o texto reproduzido pelos escritores e a Figura - 3.5 apresenta uma amostra original de um manuscrito redigida por um escritor.

## 3.2 Geração de Textura

Hanusiak et al. (2012) propõem o processo de geração de textura, utilizada na etapa de pré-processamento deste trabalho. Ela consiste em reorganizar todos os componentes conexos em um novo espaço mantendo a inclinação original da escrita do documento, mas reduzindo os espaços entre linhas, palavras e caracteres do texto. O resultado do processo de segmentação produz uma imagem que contém as características do estilo de escrita, porém com a textura mais densa.

O método para a geração de textura consiste em binarizar a imagem de entrada utilizando o algoritmo de segmentação Otsu (Otsu, 1975), assim, percorrendo a imagem e desconsiderando pequenos componentes conexos, evitando possíveis ruídos. Os demais delimitadores dos componentes conexos são utilizados para extrair os componentes originais

<sup>3</sup>Download *database* disponível em: <<http://bit.ly/download-BFL>>



De  
 Fernando Quintas Zanon  
 Rua Luiz Kirt Walterez, 87 - Ap. 300  
 Xenópolis, Nova Yolanda 14506-159

Para  
 Dr. Onório Bob Grant

Soube, através de publicação pela imprensa local, que V. Sas. necessitam de um funcionário na Seção de Correspondência do Departamento Pessoal. Venho, portanto, candidatar-me a esta vaga.

Sou brasileiro, solteiro, com 18 anos, curso a 3ª série do Curso Técnico de Contabilidade do Colégio Horácio Alves - Escola Municipal de 2º Grau - e possuo alguma prática de datilografia e arquivos.

Trabalhei durante dois anos nas Lojas Universais Rayon S.A. onde exerci as funções de Auxiliar de Escritório Júnior.

Inicialmente, coloco-me à disposição de V. Sas. para um período de experiência, quando, então, poderão tranquilamente avaliar minhas aptidões.

Na expectativa de uma resposta apresento-lhes cordiais saudações,

Fernando Zanon

**Figura 3.4:** Conteúdo textual da base BFL reproduzido pelos escritores.

De  
 Fernando Quintas Zanon  
 Rua Luiz Kirt Walterez, 87 - Ap. 300  
 Xenópolis, Nova Yolanda, 14506-159

Para  
 Dr. Onório Bob Grant

Soube, através de publicação pela imprensa local, que V. Sas. necessitam de um funcionário na Seção de Correspondência do Departamento Pessoal. Venho, portanto, candidatar-me a esta vaga.

Sou brasileiro, solteiro, com 18 anos, curso a 3ª série do Curso Técnico de Contabilidade do Colégio Horácio Alves - Escola Municipal de 2º Grau - e possuo alguma prática de datilografia e arquivos.

Trabalhei durante dois anos nas Lojas Universais Rayon S.A. onde exerci as funções de Auxiliar de Escritório Júnior.

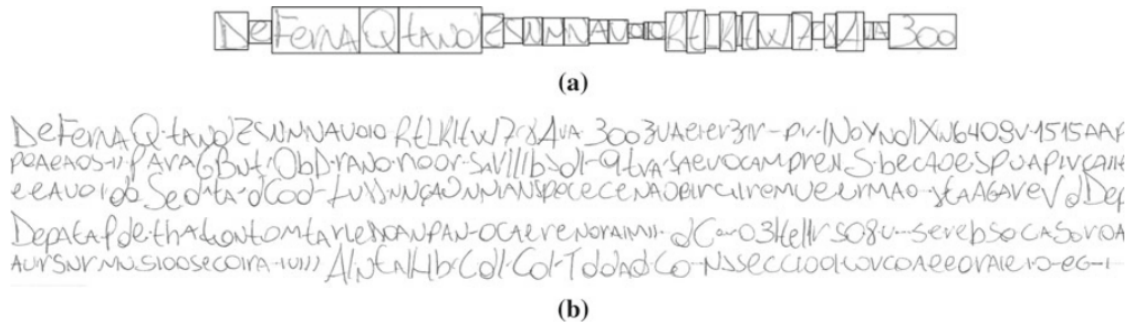
Inicialmente, coloco-me à disposição de V. Sas. para um período de experiência, quando, então, poderão tranquilamente avaliar minhas aptidões.

Na expectativa de uma resposta apresento-lhes cordiais saudações,

Zanon  
 Fernando Zanon

**Figura 3.5:** Amostra de um documento da base BFL redigida por um dos escritores.

da imagem, em escala de cinza. Finalizando, esses componentes são alinhados gerando uma nova imagem, conforme pode ser observado na Figura - 3.6.



**Figura 3.6:** Processo de Geração de Textura. (a) Alinhamento dos componentes e (b) geração da textura. Adaptado de (Hanusiak et al., 2012).

### 3.3 Descritores de Textura

Assim como a cor e as formas, a textura é um atributo importante em uma imagem. Com ela é possível que humanos identifiquem distribuição espacial, variação de luminosidade e arranjo estrutural da superfície (Pedrini e Schwartz, 2008). Mesmo sendo algo facilmente perceptível ao olho humano, a textura não possui uma definição formal fácil.

Para realizar a extração de *features* do trabalho foram explorados dois descritores de textura de abordagem estrutural, o *Local Binary Pattern* (LBP) e o *Local Phase Quantization* (LPQ) discutidos em 3.3.1 e 3.3.2. A justificativa para utilização desses descritores nos experimentos se deve aos bons resultados apresentados na literatura (Du et al., 2010), tanto para aplicações de identificação e verificação de escritores quanto para as mais diversas áreas (Almeida et al., 2013; Costa et al., 2012).

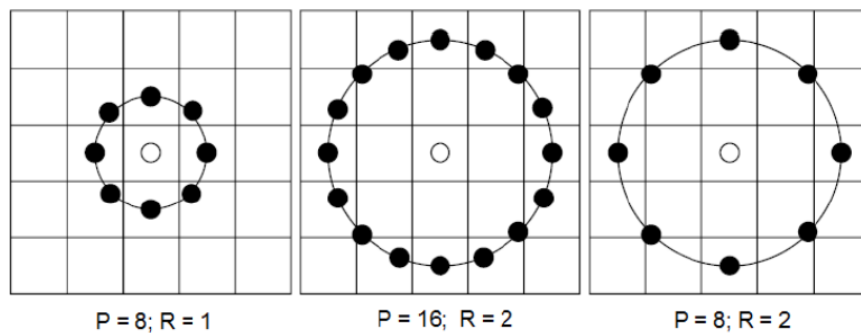
#### 3.3.1 *Local Binary Pattern* - LBP

Ojala et al. (2002) propôs o *Local Binary Pattern* (LBP) que tem por finalidade realizar o reconhecimento de padrões de binários locais em uma dada imagem e seus pixels adjacentes para encontrar um histograma. Tais características fazem sua utilização como um dos principais descritores de textura empregados em sistemas de reconhecimento de imagem.

O LBP inicialmente considera que cada pixel existente na imagem será, em algum momento, o ponto central para uma matriz de convolução de tamanho  $3 \times 3$ . Avaliando sua vizinhança em relação ao pixel central e considerando uma distância de um pixel, totalizam-se oito vizinhos. Comparando a intensidade do pixel central com os seus vizinhos, (caso seu vizinho possua intensidade maior que a do pixel central), este será

classificado como 1, caso contrário 0. Logo após, é realizada uma multiplicação entre os valores binários e os decimais gerados, atribuindo ao pixel central o somatório desta função.

O operador  $LBP_{P,R}$ , em  $R$  produz  $2^P$  valores de saída diferentes, correspondendo aos  $2^P$  padrões binários diferentes que podem ser formados pelos  $P$  pixels no conjunto de vizinhos. A Figura - 3.7 demonstra algumas possibilidades de valores para  $P$  e  $R$ .



**Figura 3.7:** Variações de  $P$  e  $R$  no LBP. Adaptado de (Mäenpää, 2003).

Os padrões fundamentais do LBP são considerados uniformes, quando ele contém até duas transições, indo de 0 para 1 ou vice-versa, considerando a sequência binária circular (Ojala et al., 1996, 2002). Um padrão é considerado uniforme quando em um dado número de *bits* há um número de transições menor ou igual a dois. Toma-se como exemplo as seguintes sequências binárias: 00000000, 11111111, 00111110 e 00010000. Nos dois primeiros casos, os números binários são compostos por 8 elementos idênticos, sendo um padrão uniforme, por não conter nenhuma transição numérica. Nos dois exemplos seguintes, observa-se que os mesmos podem ser divididos em 3 partes, ou seja, duas transições de estado, também sendo um padrão uniforme. Tem-se agora como exemplo a seguinte sequência binária: 00011011 observa-se que ela é constituída por 4 partes e conseqüentemente três transições de estado, o que acaba por desconsiderar a mesma como um padrão uniforme, pois extrapola o número máximo estipulado pelos autores.

Assim, o número de rótulos para um vizinhança de 8 pixels no LBP é de 256, mas de apenas 58 uniformes no  $LBP^{u2}$ , porém todas as combinações que apresentam um padrão não uniforme devem participar de uma coluna adicional no histograma gerado (Ojala et al., 1996). Devido a esse fato, o vetor de características LBP para a configuração de 8 vizinhos com distância 2 possui 59 atributos em sua constituição. Neste trabalho, os resultados apresentados foram obtidos usando a configuração tradicional ( $LBP_{8,2}^{u2}$ ), gerando um vetor de 59 atributos.

### 3.3.2 Local Phase Quantization - LPQ

Originalmente proposto para uso com imagens borradas ou afetadas pela iluminação não uniforme, porém ele também apresenta bons resultados para imagens que não contenham borramentos. Assim, Ojansivu e Heikkilä (2008) apresentam o *Local Phase Quantization* (LPQ).

O descritor LPQ se baseia na propriedade de invariância ao borramento de fase de Fourier. Ele utiliza a informação de fase local extraída de uma *Two-Dimensional Discrete Fourier Transform* (2D DFT) computada sobre uma vizinhança retangular, chamada janela local, para cada pixel da imagem. A informação da fase de uma imagem de tamanho  $N \times N$  é baseada nas propriedades de espectro de fases de uma *Short-time Fourier Transform* (STFT), que é definida pela equação 3.1.

$$\hat{f}_{u_i}(x) = (f * \phi_{u_i})(x) \quad (3.1)$$

Ojansivu e Heikkilä (2008) atribuem que a função  $f(x)$  de uma imagem é resultado de um processo de primeira ordem de Markov, em que o coeficiente de correlação entre dois pixels  $x_i$  e  $x_j$  é relacionado exponencialmente com a sua distância  $d_{i,j}^s$ . Para o vetor  $f$  é definida uma matriz de covariância  $C$  de tamanho  $m^2 \times m^2$ , dada pela equação 3.2. A matriz de covariância dos coeficientes de Fourier pode ser obtida por  $D = \phi C \phi^T$ . Assim, considerando que  $D$  não é uma matriz diagonal, os coeficientes são correlatos e podem deixar de ser por meio da transformação  $E = V^T \hat{F}$ , em que  $V$  é uma matriz ortogonal derivada do valor de decomposição singular, *Singular Value Decomposition* (SVD), da matriz  $D$ , com  $D' = V^T D V$ .

$$C_{i,j} = \sigma^{\|x_i - x_j\|} \quad (3.2)$$

Logo, os coeficientes são quantizados através da equação 3.3, em que  $e_{ij}$  são componentes de  $E$ . Esses elementos são valores inteiros transformados de binário para decimal por meio da equação 3.4 e representam valores entre 0 e 255. Então, o LPQ por meio de todas as posições da imagem gera seu vetor de características que correspondem a um histograma com 256 posições.

$$q_{ij} = \begin{cases} 1, & \text{se } e_{ij} \geq 0 \\ 0, & \text{caso contrário} \end{cases} \quad (3.3)$$

$$b_j = \sum_{i=0}^7 q_{ij} 2^i \quad (3.4)$$

## 3.4 *Deep Learning*

A aprendizagem profunda, do inglês *deep learning* visa aprender automaticamente em vários níveis de abstração, permitindo que o sistema aprenda funções complexas mapeando recursos de entrada para a saída dos dados, sem depender de tarefas extraídas a mão (*handcrafted*). O que é importante para recursos de nível superior, os quais o ser humano não sabe como especificar qual a melhor característica de um determinado problema (Bengio et al., 2009).

A *deep learning* permite ainda que os modelos computacionais que são compostos de várias camadas de processamento, aprendam representações de dados com múltiplos níveis de abstração (LeCun et al., 2015). Esse modelo de aprendizagem tem como principal característica a utilização de camadas, tanto para realizar a extração de características quanto para a classificação (LeCun et al., 2015).

Deng et al. (2014) complementam que a *deep learning* refere-se à uma classe bastante ampla de técnicas e arquitetura de aprendizagem, dentre as quais estão as Redes Neurais Artificiais (RNA), modelos de probabilidade hierárquica e uma variedade de métodos de aprendizagem de máquina supervisionadas ou não.

Para a tarefa de classificação de imagens, os melhores resultados publicados utilizam uma arquitetura em particular, chamada de *Convolutional Neural Networks* (CNNs). (Hafemann, 2014). Ela é descrita na subseção em 3.4.1. Nesta seção também são descritas as camadas de uma CNN: a convolucional, a de *pooling* e a totalmente conectada, respectivamente nas subseções 3.4.2, 3.4.3 e 3.4.4. Para finalizar a seção, será apresentado na subseção 3.4.5 o processo de treinamento de uma CNN.

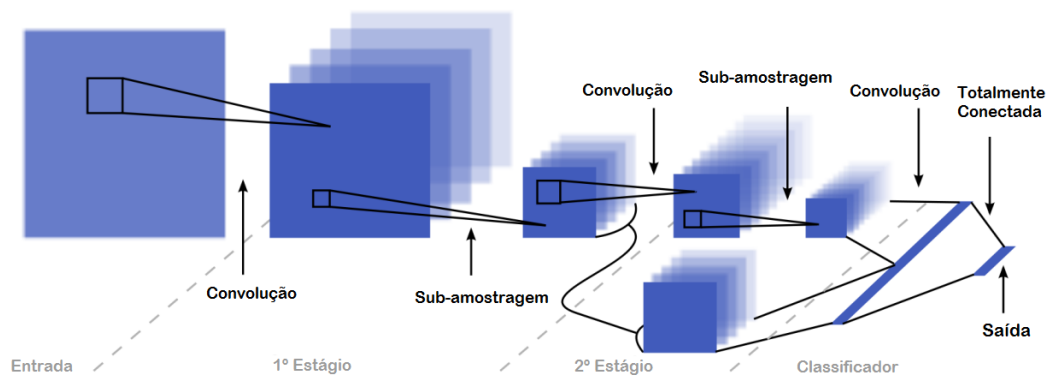
### 3.4.1 Convolutional Neural Networks (CNNs)

As *Convolutional Neural Networks* (CNNs) combinam três ideias de arquitetura para garantir invariância de deslocamento, escala e distorção, sendo: campos receptivos locais; pesos compartilhados; e amostragem espacial ou temporal (LeCun et al., 1998). CNNs foram inspiradas na estrutura do sistema visual (Bengio et al., 2009).

A arquitetura das CNNs fornece um nível de invariância para deslocamento, escala e rotação como o campo receptivo local, permitindo que o neurônio ou unidade de processamento acesse as características elementares, tais como arestas orientadas ou bordas (Arel et al., 2010).

Um exemplo de CNN retirado da literatura é apresentado na Figura - 3.8 (Sermanet e LeCun, 2011). Nessa arquitetura a entrada é uma imagem 2D, processada de maneira

*feedforward*. No primeiro estágio tem-se a primeira camada convolucional, depois uma camada de *pooling* aplica uma sub-amostragem (*downsampling*) nas características mapeadas pela primeira convolução. No segundo estágio, mais uma sequência de camadas convolucional e de *pooling* são aplicadas nos resultados de saída do primeiro estágio. Por fim, é aplicada mais uma convolução no produto da segunda camada *pooling* e a arquitetura termina com uma rede neural tradicional.



**Figura 3.8:** *Convolutional Neural Network* adaptada do trabalho de (Sermanet e LeCun, 2011).

CNNs e as técnicas de classificação relacionadas, começaram ter um grande impacto por meio de resultados na classificação de imagens da competição ImageNet de 2012 e 2013 (Deng et al., 2014).

O processo de treinamento de CNNs tem-se beneficiado com as modernas GPUs, acrônimo para *Graphical Processing Units*. Pois, o processamento das convoluções pode ser otimizado em GPUs modernas, que possuem até milhares de núcleos de processamento, assim, possibilitando o processamento em paralelo das convoluções.

### 3.4.2 Camada Convolucional

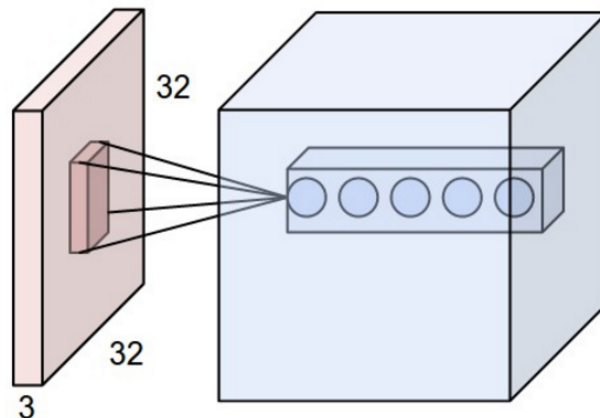
A camada convolucional possui filtros treináveis, chamados também de mapa de características ou máscaras, sendo aplicados em toda imagem de entrada. Nos filtros, cada neurônio é conectado apenas a um subconjunto dos neurônios na camada anterior. Além disso, existem os pesos que são compartilhados entre os neurônios, levando os filtros a aprender padrões frequentes que ocorrem em qualquer parte da imagem (Hafemann, 2014).

Um hiperparâmetro desta camada consiste em um conjunto de filtros que são aprendidos conforme a entrada que recebem. Outros hiperparâmetros são a profundidade dos filtros; os *strides* que são a distância que um filtro percorre até a próxima aplicação e a função de ativação dos neurônios artificiais.

Um filtro é espacialmente pequeno (largura e altura), mas estende-se através da profundidade da imagem de entrada. Por exemplo, no caso de uma imagem de entrada 2D, os filtros seriam de tamanho  $(5 \times 5)$  ou  $(8 \times 8)$  pixels de largura e altura, no caso de uma imagem colorida um exemplo seria  $(5 \times 5 \times 3)$ , ou seja, 5 pixels de largura e altura e a profundidade 3, pois a imagem em questão tem três canais de cores (Hafemann, 2014; Li et al., 2015).

Li et al. (2015) complementam que a rede vai aprender filtros que se ativam quando veem algum tipo de recurso visual, como uma borda de alguma orientação ou uma mancha de alguma cor.

Um exemplo de aplicação da camada convolucional pode ser observado na Figura - 3.9. Nela o bloco em vermelho é o sinal de entrada, ou seja, uma imagem colorida (três canais cores) de tamanho  $32 \times 32 \times 3$ . Independente de quantidade de canais de cores que compõem a imagem, um filtro convolucional deve gerar uma imagem de apenas duas dimensões. O bloco em azul é a primeira camada da convolucional, nele são aplicados cinco filtros.



**Figura 3.9:** Exemplo de aplicação da camada convolucional em uma imagem de dimensões  $32 \times 32 \times 3$  com aplicação de 5 filtros (Li et al., 2015).

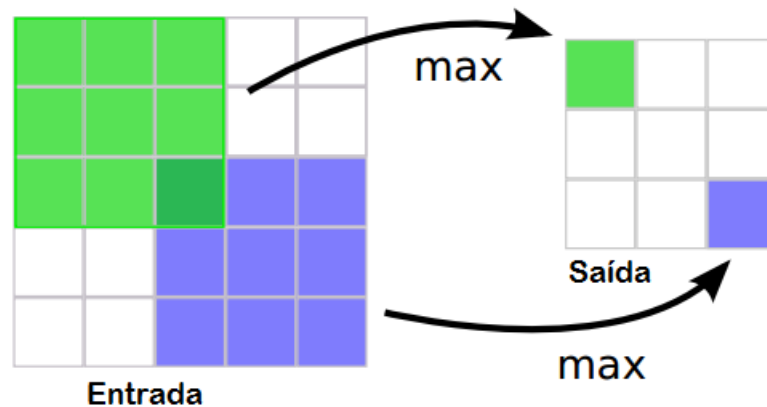
### 3.4.3 Camada de Pooling

As camadas de *pooling* usualmente são intercaladas com as camadas convolucionais. A função desta camada é implementar uma função não linear de subamostragem, ou seja,

diminuir o tamanho da imagem, para capturar pequenas invariâncias que representam padrões que podem ocorrer nas imagens.

A configuração dos hiperparâmetros dessa camada são o tamanho do *pooling*, caso seja uma imagem em 2D, o tamanho do *stride* que é a distância do deslizamento da máscara na camada e qual é o modo de tratamento de bordas. Na camada de *pooling* existe uma aplicação comum utilizada na literatura das CNNs que é chamada de *max-pooling*, na qual em uma unidade de *pooling* tem-se a ativação máxima na região de entrada.

A Figura - 3.10 ilustra o exemplo da camada de *pooling*, aplicando um *max-pooling* de tamanho  $3 \times 3$  e com *stride* =  $1 \times 1$ . A Figura - 3.11 ilustra outro exemplo da camada de *pooling*, também aplicando o *max-pooling* de tamanho  $2 \times 2$  e com *stride* =  $2 \times 2$ .



**Figura 3.10:** Exemplo da fase de aplicação do *pooling* na imagem  $9 \times 9$ , adaptado de (Hafemann, 2014).

O trabalho de Scherer et al. (2010) avalia diferentes arquiteturas de *pooling*, assim observa que o *max-pooling* possui os melhores resultados em aplicações.

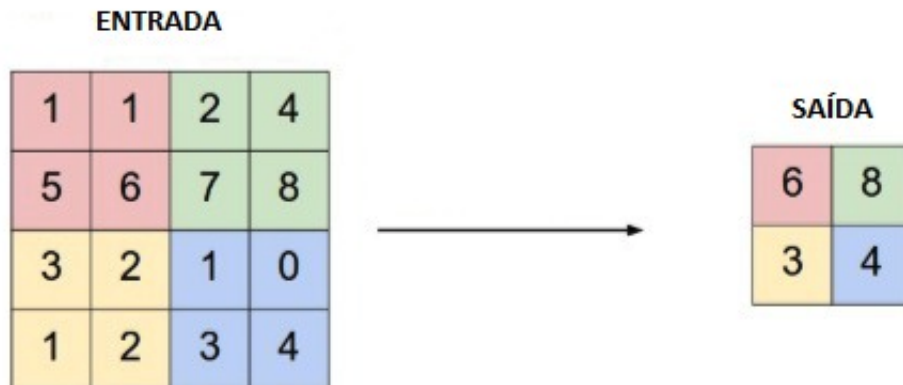
#### 3.4.4 Camada Totalmente Conectada

A camada totalmente conectada é similar as redes neurais multicamadas, do inglês *Multi-Layer Perceptron* (MLP). Ela é responsável por conectar as camadas de forma similar a camada convolucional, porém sem compartilhar os pesos (Hafemann, 2014).

A configuração desta camada consiste no número de neurônios que compõem cada camada da rede e sua função de ativação. As arquiteturas de CNNs estudadas na literatura costumam ter pelo menos uma camada totalmente conectada.

É comum encontrar implementações de MLPs que utilizam a função de ativação tangencial hiperbólica, porém nos trabalhos mais recentes que utilizam CNNs com MLPs





**Figura 3.11:** Exemplo 2 da fase de aplicação do *pooling* na imagem  $8 \times 8$ , adaptado de (Li et al., 2015).

é comum utilizar a função de ativação ReLU (*Rectified Linear Units*) (Hafemann, 2014). Por fim, além de realizar a classificação, ter uma camada totalmente conectada se torna uma forma de aprender as combinações com um baixo custo computacional.

### 3.4.5 Treino de CNNs

O processo de treinamento de CNNs é feito usualmente em GPUs. É possível realizar a implementação de CNNs por meio de *frameworks* nas mais diversas linguagens de programação. O treinamento de uma CNN é similar ao de uma MLP, que utiliza as etapas de *feedforward* e *feed backward*, ou seja, o processo de *backpropagation*.

O que usualmente ocorre no treinamento de CNNs é o *overfitting*, que é o sobreajustamento da base de treino, ou seja, o modelo de aprendizagem deve aprender atributos que descrevam as classes sem estarem demasiadamente ajustados às amostras de treino. Existem algumas formas para tentar evitar o *overfitting*, como ter uma base de dados grande. Duas técnicas podem ser aplicadas para tentar combater o *overfitting*. A primeira é o *dropout*, que consiste em desativar neurônios na etapa de treinamento. A segunda é o *data augmentation*, essa técnica consiste em aumentar a base de dados no treino por meio algumas transformações geométricas básicas como rotações, inclinações nas imagens, alteração de escala, entre outras. Ambas técnicas têm se mostrado eficazes na literatura Krizhevsky et al. (2012); Perez e Wang (2017).

### 3.5 Representação da Dissimilaridade

O objetivo da abordagem de dissimilaridade é transformar um problema de reconhecimento de padrões aparentemente difícil, no qual o número de classes é muito grande ou não especificado em um problema de duas classes (Cha e Srihari, 2000).

Cha e Srihari (2000) descrevem a dissimilaridade como uma transformação de um problema de múltiplas classes em um problema binário. Não tendo a necessidade de retreinar um modelo quando inseridas novas classes. Pekalska e Duin (2002) complementam que a abordagem da dissimilaridade pode ser viável quando a descrição baseada em características de objetos forem de difícil obtenção ou se apresentarem pequenas quantidades de dados para fins de aprendizagem.

Cha e Srihari (2000) descrevem a formulação matemática e Rivard (2010) a adapta para o contexto de escritas. Assim, sejam  $x_q, x_r$  dois vetores de características do domínio das características rotuladas  $y_q, y_r$ , respectivamente e  $u_r$  é o vetor de distância no domínio da distância resultante da transformação da dissimilaridade, representado na Equação 3.5.

$$u_r = |x_q - x_r| \quad (3.5)$$

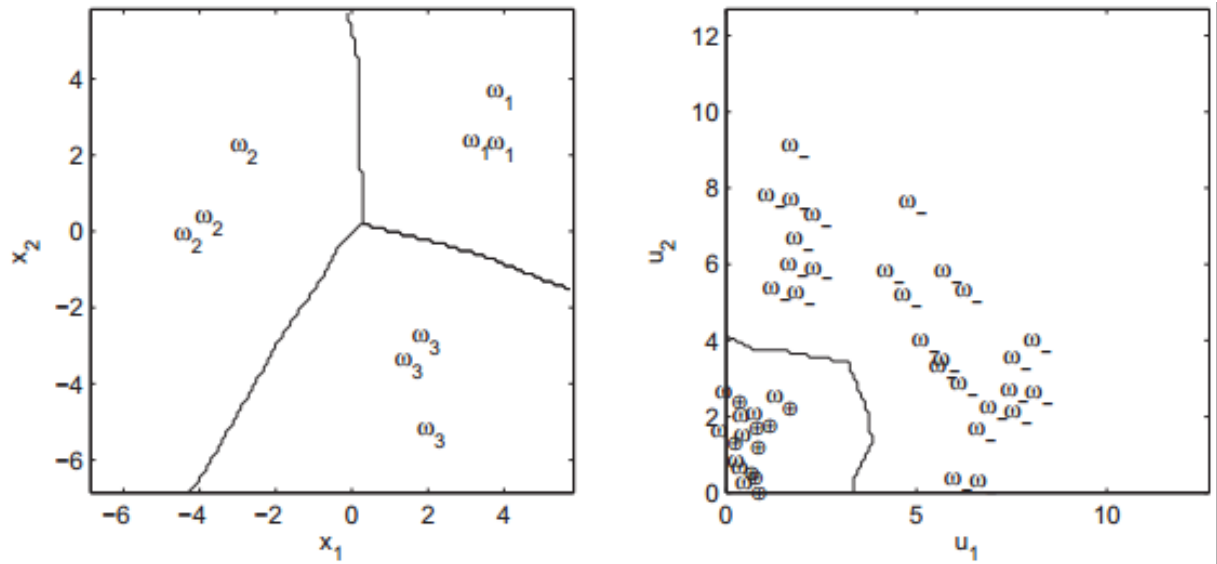
em que  $|\cdot|$  é o valor absoluto. Vale ressaltar que cada elemento de vetor  $u_r$  é igual à distância entre os elementos correspondentes dos vetores  $x_q$  e  $x_r$ , assim a dimensionalidade do vetor de distância e de característica é a mesma. No domínio da distância, indiferentemente do número de escritores, existem apenas duas classes: a intraclasse, denotada por  $\omega_{\oplus}$  (positiva) e a interclasse, denotada por  $\omega_{\ominus}$  (negativa). O vetor de distância  $u_r$  é atribuído ao rótulo  $v_r$  de acordo com 3.6:

$$v_r = \begin{cases} \omega_{\oplus}, & \text{se } y_q = y_r \\ \omega_{\ominus}, & \text{caso contrário} \end{cases} \quad (3.6)$$

Logo, documentos do mesmo escritor devem estar próximos uns dos outros no espaço de características, agrupando-se assim perto da origem no espaço de distância, enquanto que os documentos de diferentes escritores devem estar distantes uns dos outros e também da origem no espaço de características (Rivard, 2010).

Para ilustrar tem-se a Figura - 3.12 que representa três classes escritores diferentes,  $(\omega_1, \omega_2, \omega_3)$ , no espaço de características (à esquerda), projetado no espaço de distância (à direita) pela transformação de dissimilaridade para formar duas classes  $\omega_{\oplus}, \omega_{\ominus}$ . Os limites de decisão em ambos os espaços são inferidos pelo algoritmo do vizinho mais próximo. E a Figura - 3.13 representa a função de probabilidade para cada distribuição, tanto no espaço

de características (à esquerda) quanto no espaço de distância. No espaço de distância (à direita), pode-se observar que a intraclasse  $\omega_{\oplus}$  é agrupada perto da origem, enquanto a interclasse  $\omega_{\ominus}$  é distribuída distante da origem.



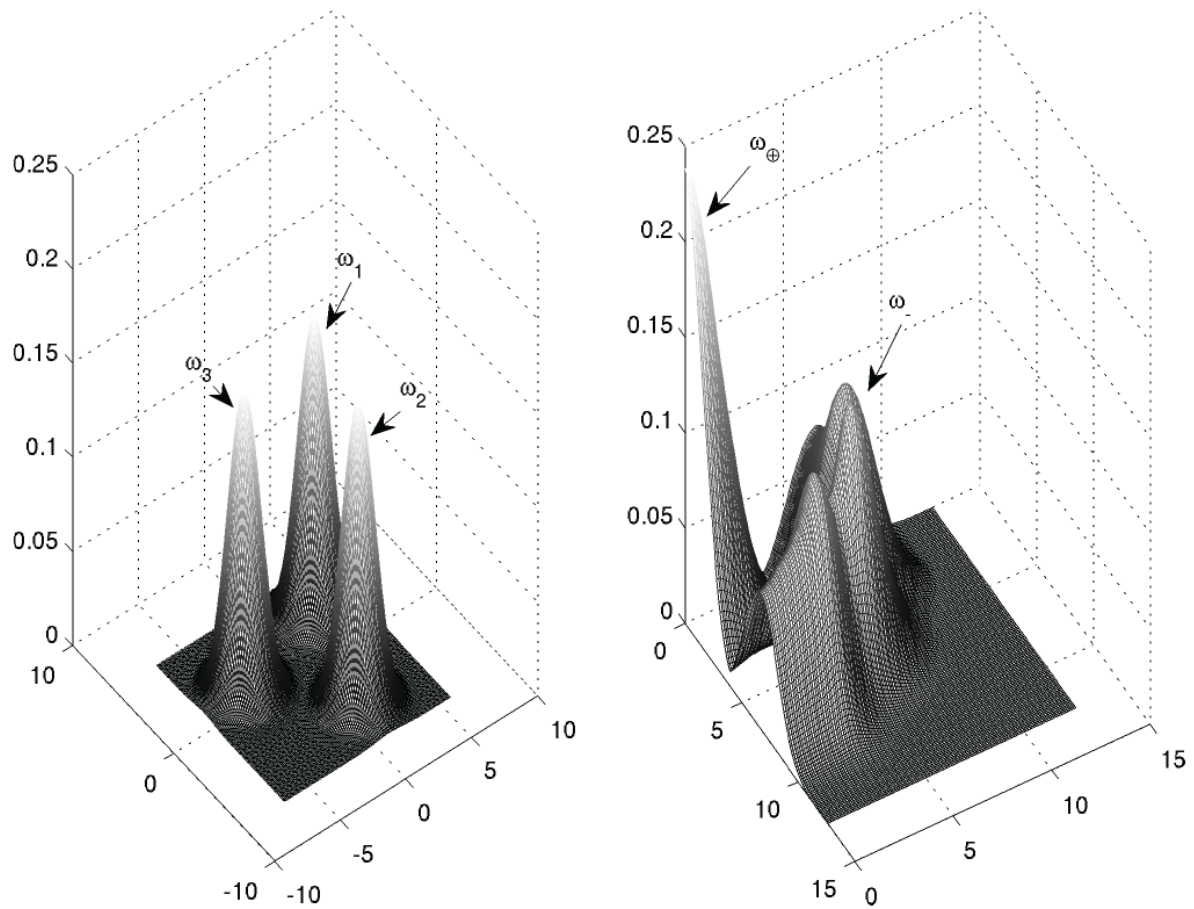
**Figura 3.12:** Características de três diferentes classes (à esquerda), transformação com a dissimilaridade de três para duas classes (à direita) (Rivard, 2010).

## 3.6 Combinação de Classificadores

Quando há mais de um classificador para classificar uma única amostra, pode-se utilizar algumas técnicas para combinar as saídas dos classificadores e chegar a uma decisão final. A combinação de classificadores pode ser realizada quando as saídas dos classificadores (predições) apresentam, para cada amostra, uma estimativa de probabilidade para cada classe existente no esquema de classificação. Para gerar os resultados de combinações deste trabalho foram utilizados três modelos de combinação de classificadores propostas em (Kittler et al., 1998). São elas a Regra da Soma, Regra do Produto e Regra do Máximo discutidas a seguir.

### 3.6.1 Regra da Soma

A regra consiste em realizar um somatório para cada classe. O somatório é feito por meio das probabilidades encontradas em cada classificação de uma amostra. Após feito todos os somatórios a amostra questionada recebe o rótulo da classe em que se encontra a maior probabilidade.



**Figura 3.13:** Distribuições no espaço de características (à esquerda) projetadas no espaço da distância pela transformação da dissimilaridade (à direita) (Rivard, 2010).

Tem-se que probabilidades *a posteriori* calculada por seu classificador não irão diferenciar muito das probabilidades *a priori*. O autor das regras considera essa suposição rígida mas pode ser satisfeita quando a informação é altamente ambígua, por exemplo, quando existe um alto nível de ruído. Assim, a regra de decisão é definida na equação 3.7.

$$\text{atribuir } Z \rightarrow w_j \text{ se} \\ (1-R)P(w_j) + \sum_{i=1}^R P(w_j|x_i) = \max_{k=1}^m [(1-R)P(w_k) + \sum_{i=1}^R P(w_k|x_i)] \quad (3.7)$$

### 3.6.2 Regra do Produto

Similar à regra da soma, com a diferença que a regra do produto ao invés de somatórios são calculados produtórios associados à saída de cada classificador. Assim, ela quantifica a probabilidade de dada uma hipótese ser combinada a probabilidade *a posteriori* gerada por cada classificador por meio de uma regra do produto. Sendo ela uma regra severa caso tenha uma classificação errada. Assim, a regra de decisão é definida na equação 3.8.

$$\begin{aligned} & \text{atribuir } Z \rightarrow w_j \text{ se} \\ & p^{-(R-1)} w_j \prod_{i=1}^R P(w_j|x_i) = \max_{k=1}^m p^{-(R-1)}(w_k) \prod_{i=1}^R P(w_k|x_i) \end{aligned} \quad (3.8)$$

Por meio das equações de soma e produto apresentadas respectivamente pelas equações 3.7 e 3.8, é possível definir outras regras de combinação.

### 3.6.3 Regra do Máximo

A regra do máximo possui uma ideia simples e direta: escolher a classe por meio da maior predição encontrada nas amostras classificadas. A regra tende a ser menos severa, pois basta que um dos classificador seja bom para ter um alto valor de probabilidade. Ela faz a aproximação por meio da regra da soma pelas máximas das probabilidades *a posteriori* e assumindo como iguais probabilidades *a priori*. Assim, a regra de decisão é definida na equação 3.9.

$$\begin{aligned} & \text{atribuir } Z \rightarrow w_j \text{ se} \\ & \max_{i=1}^R P(w_j|x_i) = \max_{k=1}^m \max_{i=1}^R P(w_k|x_i) \end{aligned} \quad (3.9)$$

## 3.7 Considerações

Neste capítulo, foram apresentadas as bases dados e as principais técnicas computacionais a serem empregadas neste trabalho. A fim de demonstrar os conceitos introdutórios para cada uma das seções do capítulo. Contudo, algumas abordagens não foram descritas nesse trabalho, pois, centenas de trabalhos os descrevem com riqueza de detalhes. Um exemplo disso é o classificador SVM proposto e detalhado no livro de (Vapnik, 1995) a ser empregado nos experimentos deste trabalho. Ainda, mais detalhes podem ser encontrados nas referências citadas neste trabalho.

---

# Método Proposto

---

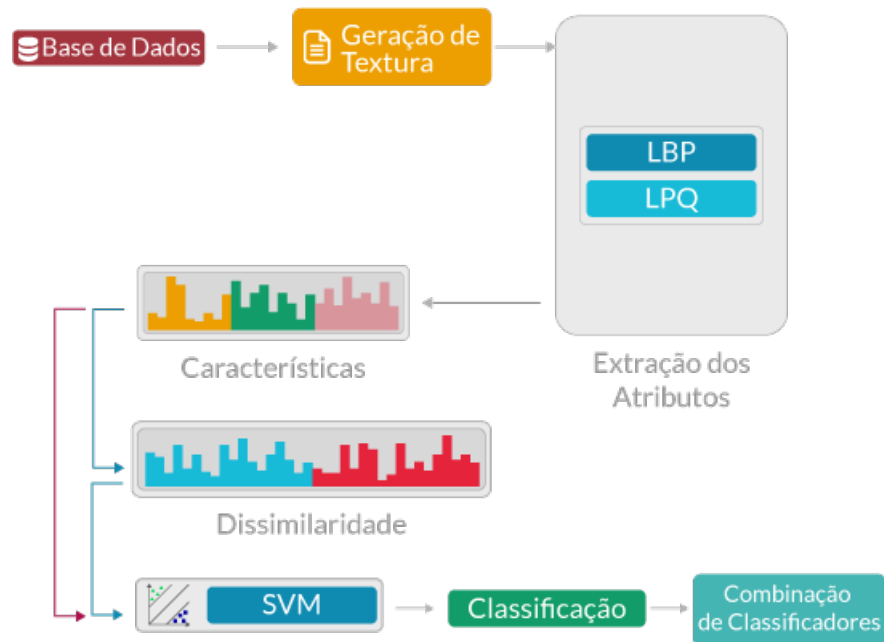
Neste capítulo, são apresentadas as etapas necessárias para o desenvolvimento do método proposto, o detalhamento das decisões tomadas e os métodos de classificação utilizados no desenvolvimento do presente trabalho. Ambas metodologias propostas são baseadas no Capítulo 3 da Fundamentação Teórica.

A Seção 4.1 apresenta o detalhamento de quais foram as etapas utilizadas para realização do método com a extração manual de características. A Seção 4.2 detalha a utilização de uma *Convolutional Neural Network* (CNN) para classificação e extração de características. Também é apresentado como é feita a divisão da base de dados e a arquitetura da rede.

## 4.1 Metodologia: Utilizando Descritores de Textura

Os experimentos utilizando essa abordagem seguiram as seguintes etapas para realizar a tarefa de classificação dos manuscritos: a partir da base de dados tem-se à geração de textura dos manuscritos, o zoneamento que consiste na divisão em nove blocos das imagens geradas, extração de características com os descritores, treinamento utilizando SVM para a classificação dos segmentos e a combinação dos classificadores com os blocos gerados. Ainda, em uma segunda abordagem também foi avaliado o cálculo da dissimilaridade (*vide* Seção 3.5) a partir das características extraídas, assim é feita uma outra classificação com os valores obtidos. A representação esquemática do método proposto para o experimento pode ser observado na Figura - 4.1.

Algumas particularidades do processo da Figura - 4.1 são descritas na sequência em subseções, nas quais são apresentadas as configurações e os padrões utilizados.



**Figura 4.1:** Representação esquemática do método proposto para os experimentos com descritores de textura.

### Base de Dados

As bases utilizadas nesta primeira abordagem foram a CVL e a LAMIS-MSHD. Em ambas bases foi aplicado o processo de geração de textura (*vide* Seção 3.2). Em seguida, foi feito o zoneamento das imagens dividindo cada amostra em nove blocos de tamanho  $256 \times 256$ .

### LBP

A criação dos vetores de características por meio LBP levou em consideração o ( $LBP_{8,2}^{u2}$ ), ou seja, considerou-se 8 pixels com uma distância de 2 pixels do pixel central, levando assim a criação de vetores de características compostos por 59 atributos.

### LPQ

Para a criação dos vetores de características a partir do LPQ se fez o uso de uma janela de dimensão  $7 \times 7$  e ao se percorrer todos os pixels da imagem com a máscara foi obtido um vetor composto por 256 atributos. A justificativa da escolha do tamanho da janela é devido ao fato do trabalho de Bertolini (2014) analisar em experimentos preliminares diversos tamanhos e apresentando o melhor desempenho com janelas de  $7 \times 7$ .

## 4.2 Metodologia: Utilizando *Deep Learning*

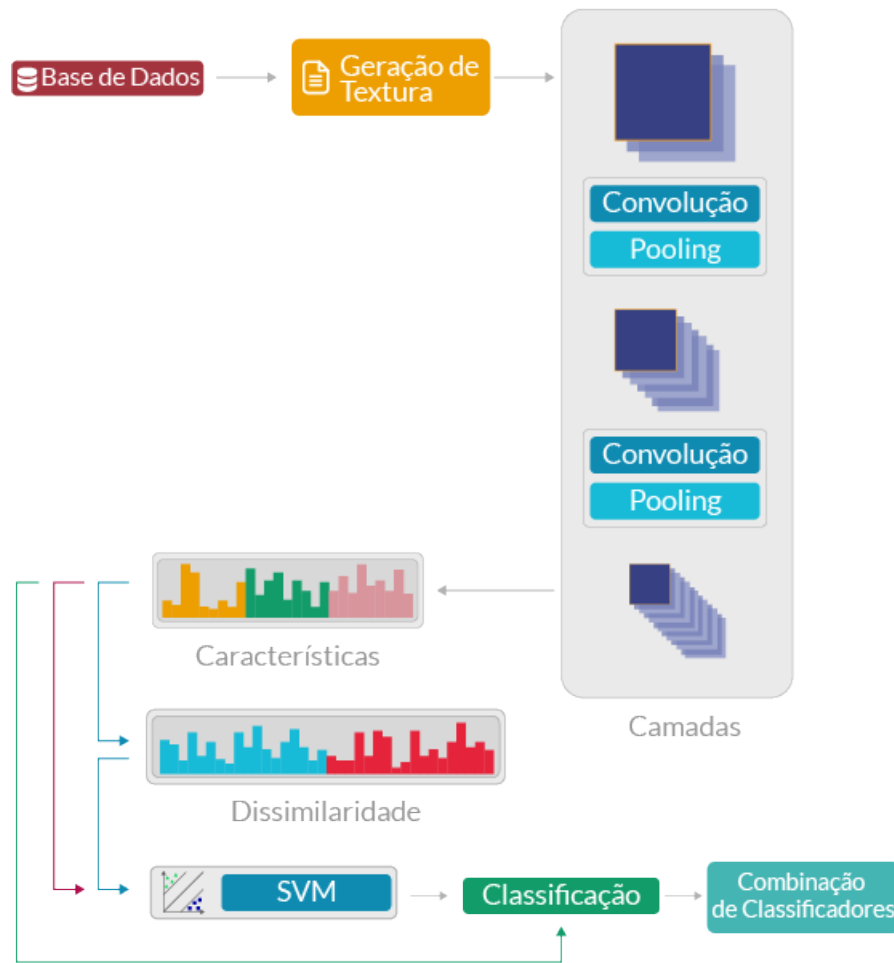
A representação do modelo para o desenvolvimento do trabalho pode ser observada na Figura - 4.2. O método consiste em utilizar *deep learning*, mais especificamente uma *Convolutional Neural Network* (CNN), assim serão abordadas três comparações utilizando CNN, descritas individualmente na sequência:

1. Por meio da base dados de manuscritos é aplicado o processo de geração de textura nas imagens, elas serão dadas como entrada em uma CNN. Nesta abordagem (flecha verde da Figura - 4.2), a própria CNN irá classificar as amostras e com os resultados gerados por ela será aplicada a combinação de classificadores;
2. Na segunda abordagem (flecha vermelha da Figura - 4.2) a CNN será utilizada como extrator de características. O processo inicial é similar ao da primeira abordagem, é gerada a textura da base de dados e a imagem é dada como entrada da CNN, mas desta vez, ela não irá até a classificação, será interrompida. Assim, os valores de ativação dos neurônios da rede (pesos) que estariam ligados na camada totalmente conectada são utilizados como características, em seguida é utilizado o *Support Vector Machine* (SVM) para a classificação e por fim, a combinação dos classificadores;
3. Na terceira e última abordagem (flechas em azul da Figura - 4.2) também será utilizada a CNN como extrator de características. O processo é similar a segunda abordagem, se diferenciando após o processo de extração de características, no resultado da extração será aplicado a dissimilaridade nas características, transformando as amostras em positivas e negativas e em seguida aplicando o SVM para suas classificações e efetuando, por fim, a combinação dos classificadores.

### 4.2.1 Rede neural convolucional

Para chegar na arquitetura da rede utilizada neste trabalho foram avaliados vários hiperparâmetros. Com a configuração dos hiperparâmetros chegou-se na rede apresentada na Figura - 4.3 e Tabela - 4.1, a mesma rede foi utilizada em todos experimentos deste trabalho com CNN, porém vale notar que o tamanho da saída não foi descrito, pois ele muda conforme a base de dados, sendo que ele corresponde ao número de classes apresentadas em cada problema.





**Figura 4.2:** Representação esquemática do método proposto para os experimentos com CNN.

**Tabela 4.1:** Informações sobre a CNN utilizada.

Camada	Número de parâmetros treináveis	Dimensionalidade da saída
Entrada	0	$1 \times 32 \times 32$
1ª Convolução 2D	960	$96 \times 32 \times 32$
1º Max pooling 2D	0	$96 \times 15 \times 15$
2ª Convolução 2D	442.496	$128 \times 15 \times 15$
3ª Convolução 2D	147.584	$128 \times 15 \times 15$
4ª Convolução 2D	147.584	$128 \times 15 \times 15$
2º Max pooling 2D	0	$128 \times 7 \times 7$
1ª Totalmente Conectada	3.211.776	512
2ª Totalmente Conectada	262.656	512
Saída	-	Número de classes

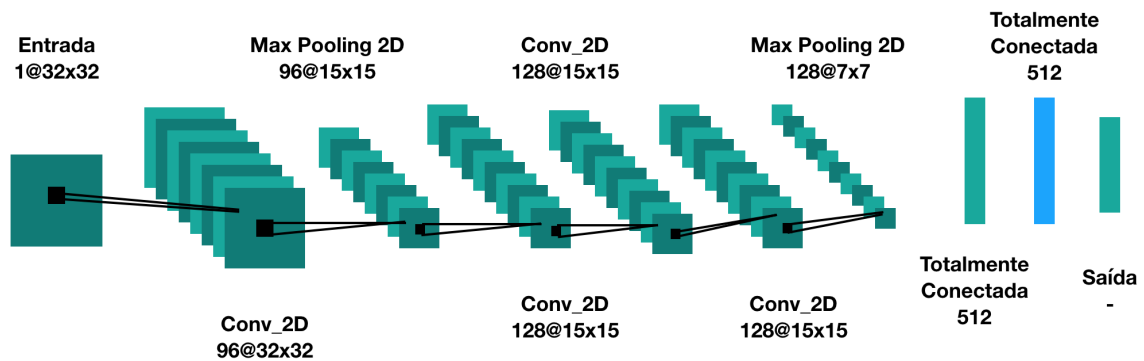


Figura 4.3: Representação da Rede Neural Convolutional utilizada.

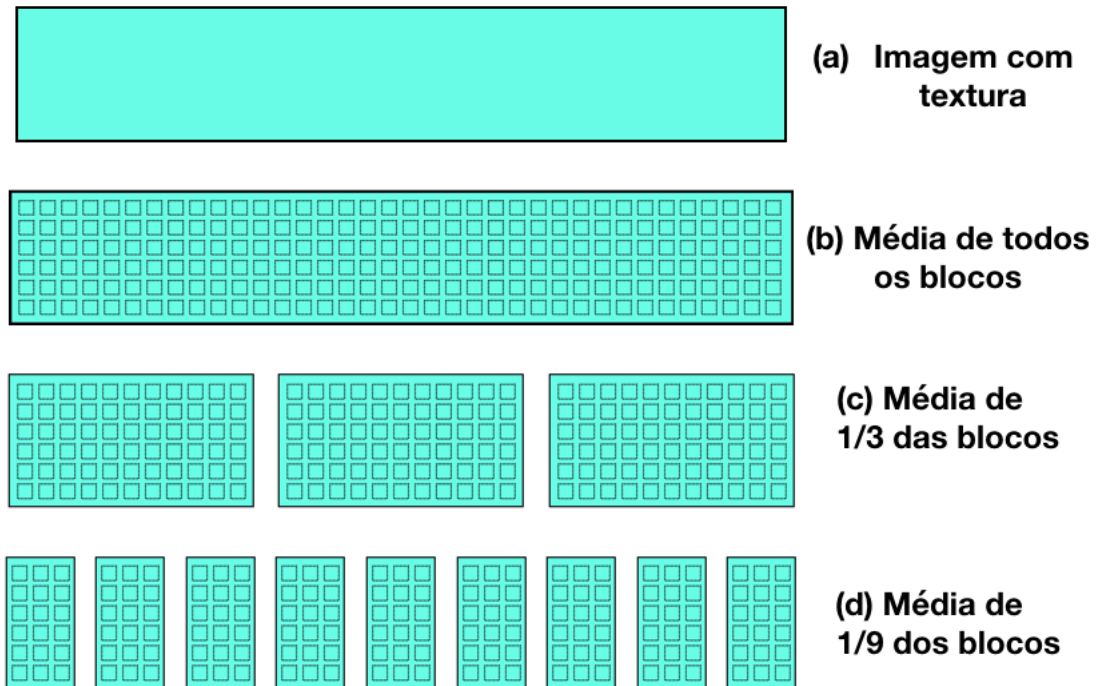
## 4.2.2 Divisão e Agrupamento das Bases de Dados

Após ter a CNN treinada é feita a extração de características, nesta etapa foram necessárias algumas definições para que fosse possível realizar a classificação utilizando essas características no SVM e também para realizar o cálculo da dissimilaridade para a identificação de escritores. A Figura - 4.4 ilustra como uma amostra de uma determinada base de dados foi utilizada.

Na Figura - 4.4 (a) é apresentado o método empregado a partir de uma amostra de textura de um determinado escritor na base de dados. Essa imagem é dividida em  $N$  blocos de tamanho  $32 \times 32$ , assim esses  $N$  blocos de cada amostra são dados como entrada na CNN, sendo possível extrair 512 características (tamanho da penúltima camada da CNN) de cada bloco. Por exemplo, com um  $N = 270$  cada amostra da base de dados se dividirá em 270 blocos com 512 características cada, o que implica em um tempo computacional muito elevado para o treinamento utilizando o SVM com *kernel* RBF e também para o cálculo da dissimilaridade, sendo assim foram adotadas as estratégias: Na Figura - 4.4 (b) média de todos os blocos, Na Figura - 4.4 (c) média de 1/3 dos blocos e Na Figura - 4.4 (d) média de 1/9 dos blocos.

A média de todos os blocos é feita com objetivo de reduzir os  $N$  blocos com 512 características para apenas uma com 512. Exemplo, então se o  $N = 270$  é feita a média das características de todos os blocos do mesmo escritor, transformando em apenas uma por amostra, viabilizando o tempo computacional da execução delas em um SVM.

A média de 1/3 dos blocos é feita com intuito de reduzir os  $N$  blocos em apenas três. Exemplo, com o  $N = 270$  a imagem é dividida em três partes contendo 90 blocos cada, assim é feita a média de cada uma dessas partes e são geradas apenas três blocos por amostra. O mesmo acontece com a média de 1/9 dos blocos com a diferença que os  $N$



**Figura 4.4:** Método divisão de blocos.

blocos são reduzidas para apenas nove. Por exemplo, é se novamente o  $N = 270$  a imagem é dividida em nove partes contendo 30 blocos cada, assim é feita a média de cada uma dessas partes e são geradas apenas nove blocos por amostra. Assim, esses dois casos: média de 1/3 e média de 1/9 serão utilizados para realizar a classificação utilizando o SVM e também para o cálculo da dissimilaridade que precisa de mais de uma amostra por escritor.

### 4.3 Considerações

Neste capítulo foram apresentadas as metodologias utilizando a abordagem tradicional e também com *deep learning* para a realização deste trabalho. No próximo capítulo são apresentados os experimentos realizados em ambos métodos.

---

# Resultados Experimentais

---

Este capítulo apresenta os experimentos realizados utilizando as características obtidas por meio de descritores de textura, descrito na Seção 5.1, os experimentos baseados na arquitetura de uma rede neural convolucional, descrito na Seção 5.2 e a discussão dos resultados na Seção 5.3.

Os experimentos foram conduzidos utilizando um CPU Intel I7 6700K 4.4GHz, 16Gb de RAM e NVIDIA GeForce GTX 1070ti. O processo de classificação utilizando o SVM foi feito com o software LIBSVM (Chang e Lin, 2011), os parâmetros utilizados foram obtidos pela aplicação de *grid search*. As redes neurais convolucionais de todos os experimentos foram implementadas em alto-nível com a biblioteca Keras (Chollet et al., 2015).

## 5.1 Abordagem Tradicional

Esta seção tem como objetivo apresentar os resultados obtidos empregando abordagens Escritor-Dependente e Escritor-Independente (uso de dissimilaridade) por meio de descritores de textura.

Foram aplicados os descritores de textura LBP e LPQ para extrair características das bases de dados. Os primeiros experimentos foram realizados utilizando a abordagem Um-Contra-Todos do SVM, a descrição está na Subseção 5.1.1. Em seguida, são demonstrados os próximos experimentos realizados utilizando dissimilaridade e seus resultados podem ser observados na Subseção 5.1.2. Em ambos experimentos são usadas as bases CVL com 115 escritores e LAMIS-MSHD com 100 escritores.

### 5.1.1 Um Contra Todos (SVM)

Existem duas estratégias populares quando se trabalha com o classificador SVM a “um contra um” e a “um contra todos”. Milgram et al. (2006) descrevem em seu trabalho as duas abordagens, os autores concluem que de acordo com a literatura não é uma tarefa fácil concluir qual delas é melhor. Esse trabalho adotou a abordagem “um contra todos” para os experimentos.

Após as características obtidas pelos descritores, elas foram utilizadas para treinamento e classificação no SVM. Assim, com as taxas obtidas pelo classificador foi utilizado a combinação de classificadores para obter a acurácia dos experimentos.

Com a base de dados CVL foram feitos cinco experimentos para esse modelo. O primeiro experimento foi feito utilizando apenas um subconjunto pequeno da base no treino, nele foram utilizadas todas as amostras do primeiro documento da base (carta-1), com manuscritos em inglês dos escritores, e o teste com apenas documentos em alemão (carta-5), o resultado no melhor caso utilizando a combinação de classificadores foi de 26,95%. Ao aumentar a base de treino, com mais cartas, os resultados melhoravam de forma significativa. Os melhores resultados obtidos foram quando a base de teste tinha documentos em inglês. A maior taxa obtida foi de 98,26%, quando inseridas os documentos em alemão na base de treino. A Tabela - 5.1 apresenta os resultados dos experimentos.

Já com a base de dados LAMIS-MSHD foram realizados quatro experimentos. Ao se treinar os manuscritos de mesmo alfabetos os resultados ficaram em 90,00% com documentos em francês e de 92,00% com documentos em árabe. Porém, por ser uma base *multi-script*, apresentou baixas taxas de acertos ao se realizar teste com documentos em alfabetos opostos ao treino. Quando o treino foi realizado com documentos em árabe e o teste em francês a taxa de acerto no melhor caso foi de 28,00% e o inverso apresentou o menor resultado de todos os experimentos que foi de 16,00%. Com isso pode ser observada a dificuldade em realizar classificações de bases *multi-script* com essa abordagem. Os resultados desse experimento são apresentados na Tabela - 5.2.

**Tabela 5.1:** Resultados Base CVL - Um Contra Todos

Treino	Teste	Taxa de Identificação					
		Regra da Soma		Regra do Produto		Regra do Máximo	
		LBP	LPQ	LBP	LPQ	LBP	LPQ
Inglês - Carta 1	Alemão - Carta 5	15,65%	20,86%	15,65%	20,00%	19,13%	<b>26,95%</b>
Inglês - Cartas 1 e 2	Alemão - Carta 5	81,73%	<b>87,82%</b>	82,60%	<b>87,82%</b>	80,86%	85,21%
Inglês - Cartas 1 e 2	Inglês - Carta 3	94,78%	<b>97,39%</b>	94,78%	96,52%	91,30%	95,65%
Inglês - Cartas 1,2 e 4 + Alemão - Carta 5	Inglês - Carta 3	96,52%	<b>98,26%</b>	97,39%	<b>98,26%</b>	96,52%	97,39%
Inglês - Cartas 1,2,3 e 4	Alemão - Carta 5	90,43%	90,43%	<b>91,30%</b>	90,43%	86,08%	88,69%

**Tabela 5.2:** Resultados Base LAMIS-MSHD - Um Contra Todos

Treino	Teste	Taxa de Identificação					
		Regra da Soma		Regra do Produto		Regra do Máximo	
		LBP	LPQ	LBP	LPQ	LBP	LPQ
Árabe - Carta 2,3,4,5,6	Árabe - Carta 7	<b>92,00%</b>	<b>92,00%</b>	90,00%	<b>92,00%</b>	89,00%	89,00%
Árabe - Carta 2,3,4,5,6,7	Francês - Carta 13	26,00%	19,00%	<b>28,00%</b>	19,00%	27,00%	18,00%
Francês - Carta 8,9,10,11,12	Francês - Carta 13	87,00%	<b>90,00%</b>	87,00%	<b>90,00%</b>	84,00%	83,00%
Francês - Carta 8,9,10,11,12,13	Árabe - Carta 7	<b>16,00%</b>	11,00%	<b>16,00%</b>	11,00%	<b>16,00%</b>	12,00%

## 5.1.2 Dissimilaridade

A mesma combinação de amostras, selecionadas para treino e teste, das bases de dados dos experimentos da Subseção 5.1.1, foram também utilizadas nestes experimentos. Com uma diferença no processo, após a obtenção das características foi aplicada a dissimilaridade.

Os experimentos com dissimilaridade realizados com a base CVL apresentaram resultados melhores. O primeiro experimento que anteriormente obteve-se acurácia de 26,95% foi para 98,26%. A Tabela - 5.3 apresenta os resultados dos experimentos na base. Para a base CVL que contém o alfabeto em latim, o descritor de textura LPQ apresentou melhores resultados por meio da regra da soma.

A base de dados LAMIS-MSHD também apresentou melhoras significativas nos resultados. No experimento anterior, a menor taxa obtida foi de 16,00% e o mesmo experimento com a aplicação da dissimilaridade teve como resultado 94,00%. Diferentemente da base CVL, a base LAMIS-MSHD é *multi-script* e ela apresentou melhores resultados por meio do uso do descritor de textura LBP. A Tabela - 5.4 apresenta os resultados obtidos. Bons resultados encontrados na base LAMIS-MSHD estão no trabalho de Bertolini et al. (2017), motivando os experimentos realizados nesse trabalho.

Os resultados deste trabalho com dissimilaridade (escritor-independente) apresentaram uma vantagem com relação a abordagem da Subseção 5.1.1 (escritor-dependente). Abordagens com escritor-dependente costumam produzir bons resultados. Porém, uma desvantagem dela é que para cada novo escritor, um novo modelo deve ser feito. Um fato importante é que esse modelo precisa de uma quantidade de dados significante para treinar um modelo confiável. Neste trabalho, o número de amostra utilizadas no aprendizado foi de 9 blocos por escritor, uma quantidade pequena.

**Tabela 5.3:** Resultados Base CVL - Dissimilaridade

Treino	Teste	Taxa de Identificação	
		Regra da Soma	
		LBP	LPQ
Inglês- Carta 1	Alemão - Carta 5	84,34%	<b>98,26%</b>
Inglês - Cartas 1 e 2	Alemão - Carta 5	84,34%	<b>98,26%</b>
Inglês - Cartas 1 e 2	Inglês - Carta 3	98,26%	<b>99,13%</b>
Inglês - Cartas 1,2 e 4 + Alemão - Carta 5	Inglês - Carta 3	98,26%	<b>99,13%</b>
Inglês - Cartas 1,2,3 e 4	Alemão- Carta 5	84,34%	<b>98,26%</b>

**Tabela 5.4:** Resultados Base LAMIS-MSHD - Dissimilaridade

Treino	Teste	Taxa de Identificação	
		Regra da Soma	
		LBP	LPQ
Árabe - Carta 2,3,4,5,6	Árabe - Carta 7	<b>99,00%</b>	98,00%
Árabe - Carta 2,3,4,5,6,7	Francês - Carta 13	<b>100,00%</b>	98,00%
Francês - Carta 8,9,10,11,12	Francês - Carta 13	<b>99,00%</b>	98,00%
Francês - Carta 8,9,10,11,12,13	Árabe - Carta 7	<b>94,00%</b>	91,00%



## 5.2 Rede Neural Convolutacional

Essa seção apresenta os resultados dos experimentos realizados para avaliar os hiperparâmetros da *Convolutional Neural Network* (CNN). A etapa consistiu em analisar dois principais comportamentos da CNN, o tamanho da camada de entrada e a dimensionalidade da penúltima camada, a totalmente conectada. Após avaliar e definir os hiperparâmetros foram feitos os experimentos com a base BFL e CVL descritos respectivamente nas Subseções 5.2.1 e 5.2.2.

A Tabela - 5.5 apresenta a rede utilizada no trabalho de Helal et al. (2017) no qual foram comparadas duas entradas para rede de tamanho  $64 \times 64$  e  $32 \times 32$ . A comparação foi realizada utilizando a base de dados CVL com 115 escritores escolhidos de forma aleatória, as amostras utilizadas passaram pelo processo de geração de textura e posteriormente utilizaram a CNN para classificação. O melhor resultado em todos os experimentos realizados foi a entrada da rede com o tamanho  $32 \times 32$ .

**Tabela 5.5:** Comparação tamanho da entrada experimentos na base CVL.

Treino	Teste	Acurácia	
		Regra da Soma	
		Entrada $64 \times 64$	Entrada $32 \times 32$
Inglês - Doc. 1	Alemão - Doc. 5	33,14%	<b>51,21%</b>
Inglês - Doc. 1 e 2	Alemão - Doc. 5	51,30%	<b>57,39%</b>
Inglês - Doc. 1 e 2	Inglês - Doc. 3	63,17%	<b>81,44%</b>
Inglês - Doc. 1, 2 e 4 + Alemão - Doc. 5	Inglês - Doc. 3	77%	<b>92,46%</b>
Inglês - Doc. 1, 2, 3 e 4	Alemão - Doc. 5	70,43%	<b>79,32%</b>

O outro hiperparâmetro definido, foi o tamanho da penúltima camada totalmente conectada. Essa camada é utilizada para extração das características do modelo. Desse modo, foram testados os seguintes tamanhos para as dimensões da saída: 512, 1024, 2048 e 4096. Para essa validação foi utilizada a base de dados BFL com 100 escritores, para treino foram utilizados os documentos 1 - 2 e para teste o documento 3. Os resultados obtidos são apresentados na Tabela - 5.6.

A CNN utilizada teve como entrada imagens de tamanho  $32 \times 32$ . O experimento consistiu em extrair os melhores resultados da classificação utilizando a CNN e a utilização das características aprendidas, posteriormente extraídas e utilizadas como entrada para o classificador SVM. Os melhores resultados dos experimentos foram quando utilizada as dimensões de 512 e 4096 que obtiveram a mesma acurácia de 93% tanto com a CNN e quanto com o SVM. Assim, o valor escolhido para a camada foi de 512, pois outro fator

também foi levado em conta para essa escolha, o tempo computacional para classificação com a CNN, levando 34 segundos por época da rede com a dimensão 512 e cerca de 65 segundos para 4096, também por época de treino da rede.

**Tabela 5.6:** Comparação tamanho camada totalmente conectada.

Tamanho Camada Totalmente Conectada	Resultado CNN	Resultado SVM	Tempo por Época
512	93%	93%	34s
1024	91%	90%	38s
2048	91%	90%	47s
4096	93%	90%	65s

Portanto, os experimentos feitos em ambas bases BFL e CVL utilizaram os mesmos hiperparâmetros. Primeiro a geração de textura aplicada nas amostras, na sequência cada uma delas foi dividida em blocos, assim as imagens ficaram fragmentadas com tamanho de  $32 \times 32$ , tamanho da camada de entrada da rede. Assim, a CNN foi utilizada de duas formas, a primeira como classificador e a segunda como extrator de características.

Pensando no risco de *overfitting*, foi implementada a regularização *dropout* na rede. E definido o valor 50, como o número de épocas de treino em ambos experimentos, pois a partir desse valor, os resultados paravam de convergir, ou seja, aumentavam as taxas no treino e enquanto no teste as taxas se mantinham.

### 5.2.1 Experimentos utilizando a Base BFL

Os experimentos com a base BFL foram feitos com 315 escritores, cada escritor possui 3 amostras divididos em documentos 1, 2 e 3. O documento 1 foi utilizado na CNN para treino da rede e posteriormente seus pesos foram utilizados para a extração de características dos documentos 2 e 3.

Utilizando a CNN como um classificador foram realizados dois experimentos. O primeiro experimento foi com as amostras dos documentos 1 de cada escritor como entrada na rede para treino dela e os documentos 2 para teste. No segundo experimento também foi utilizado o documento 1 para treino, porém as amostras do documento 3 que foram utilizadas no teste. Ambas classificações obtiveram melhor resultado com a regra da soma. Foi realizada a média das classificações que teve sua taxa de acerto em 85,25%.

No segundo momento, a CNN é utilizada como extrator de características, foi utilizado documento 1 para treinar os pesos da rede, com a rede treinada são extraídos os pesos dos documentos 2 e 3. Por meio dessas características foram feitos três novos experimentos

utilizando o SVM e dois novos experimentos com a dissimilaridade. Nesta etapa foi utilizado o método de divisão dos blocos que está descrito na Subseção 4.2.2.

O primeiro experimento utilizando SVM consistiu em fazer a média de todas as amostras, o que em analogia equivale ao documento inteiro antes da divisão. Utilizando o documento 2 no treino e o documento 3 no teste e depois utilizando o documento 3 no treino e o documento 2 no teste. A média de ambas classificações foi de 88,57%.

O experimento com CNN e o primeiro utilizando as características no SVM pode ser observado na Tabela - 5.7.

**Tabela 5.7:** Resultados CNN e SVM média de todas as amostras.

<b>Classificador</b>	<b>Taxa de Identificação</b>	<b>Desvio Padrão</b>
CNN	85,25 %	1,06
SVM	88,57 %	0,46

O segundo experimento é feito utilizando média de 1/3 das amostras, o que em analogia equivale ao documento inteiro dividido em três partes. Primeiro utilizando o documento 2 no treino e o documento 3 no teste, e segundo utilizando o documento 3 no treino e o documento 2 no teste. A média de ambas classificações teve sua taxa de acerto em 88,19%.

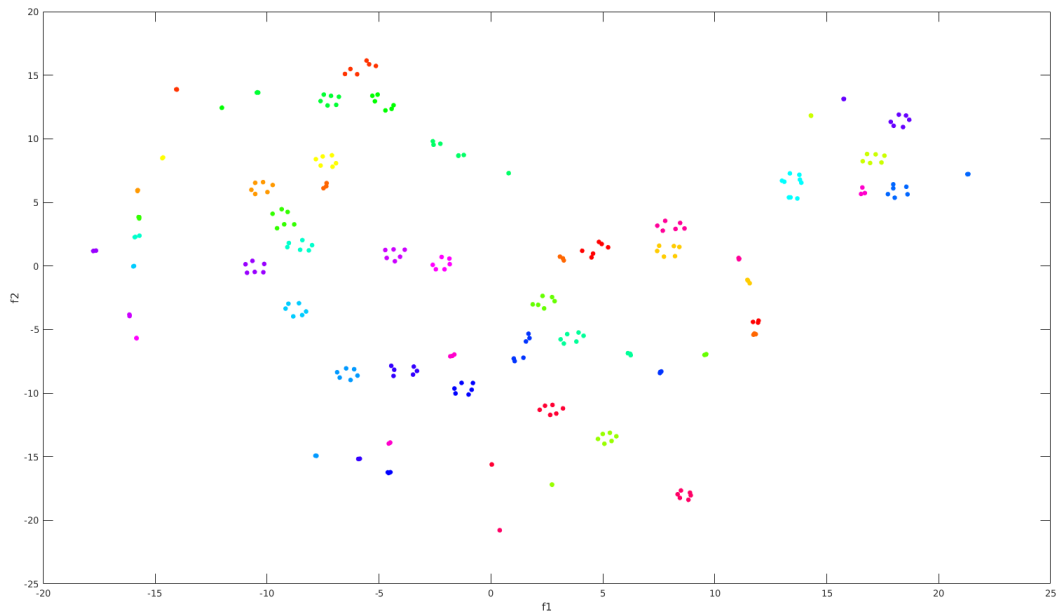
O terceiro experimento, foi feito a média de 1/9 das amostras o que em analogia equivale ao documento inteiro dividido em nove partes. Novamente com o documento 2 no treino e o documento 3 e depois utilizando o documento 3 no treino e o documento 2 no teste. A média de ambas classificações foi de 88,18%.

Os resultados do segundo e terceiro experimentos utilizando a média de 1/3 e 1/9 das amostras pode ser observado na Tabela - 5.8.

**Tabela 5.8:** Resultados SVM média de 1/3 e 1/9 das amostras.

<b>Classificador</b>	<b>Blocos</b>	<b>Taxa de Identificação</b>	<b>Desvio Padrão</b>
SVM	1/3	88,20 %	0,67
SVM	1/9	88,18 %	0,10

Com o intuito de demonstrar como ficaram as características dos escritores no espaço, foi utilizada a abordagem *t-Distributed Stochastic Neighbor Embedding* (t-SNE), descrita em (Maaten e Hinton, 2008). Foram inseridas amostras de apenas 30 escritores da base BFL, facilitando a visualização em um espaço bidimensional, pode-se observar na Figura - 5.1 como as amostras de cada escritor ficaram mais próximas e mais distantes das amostras de outros escritores.



**Figura 5.1:** Características de 30 escritores da base BFL separados por cores.

Os melhores resultados foram observados utilizando a dissimilaridade que utilizou a média de  $1/3$  das amostras e de  $1/9$  das amostras apresentados na Tabela - 5.9, em ambos casos a combinação dos classificadores melhorou os resultados, chegando em todos os casos a 100%.

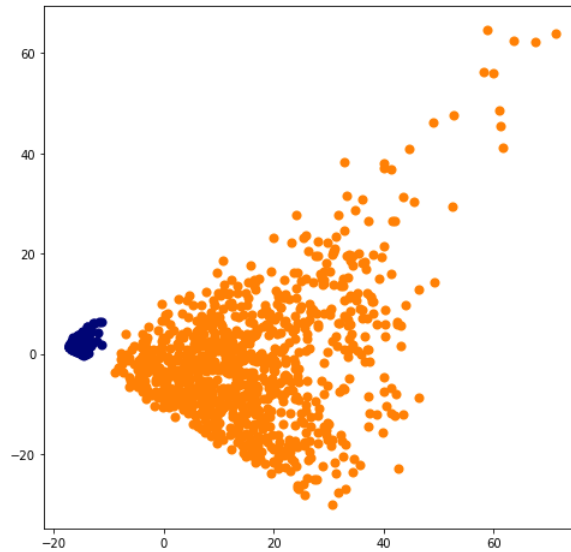
**Tabela 5.9:** Resultados dissimilaridade média de  $1/3$  e  $1/9$  das amostras.

Classificador	Blocos	Taxa de Identificação	Desvio Padrão
SVM - Dissimilaridade	$1/3$	100 %	0
SVM- Dissimilaridade	$1/9$	100 %	0

Para demonstrar como as características das amostras ficaram após aplicado o cálculo da dissimilaridade, foi utilizado a abordagem *Principal Component Analysis* (PCA), descrito em (Jolliffe, 2011), no documento 3 da base. Assim, sendo possível visualizar no espaço bidimensional que as amostras positivas ficam agrupadas e isoladas das amostras negativas.

## 5.2.2 Experimentos utilizando a Base CVL

Para os experimentos com a base CVL foram utilizadas 309 classes (escritores), com 4 amostras em inglês de cada um deles, divididos em documentos 1, 2, 3 e 4. Os documentos



**Figura 5.2:** Características positivas (azul) e negativas (laranja) da base BFL.

1 e 2 foram utilizados na CNN para treino da rede e posteriormente seus pesos foram utilizados para a extração de características dos documentos 3 e 4.

Com a CNN como classificador foram realizados dois experimentos, ambos com as amostras dos documentos 1 e 2 no treino. O primeiro experimento teve o documento 3 no teste e no segundo o documento 4. Ambas classificações tiveram os melhores resultados com a utilização da regra da soma. Foi considerada a média das duas classificações como taxa de identificação que foi de 82,06%.

Para a classificação com o SVM foram utilizados os pesos da rede para extração de características dos documentos 3 e 4. Com as novas características extraídas foram realizados novos experimentos por meio do método de divisão de blocos descrito na Subseção 4.2.2. Os novos experimentos foram feitos com o documento 3 no treino e o 4 no teste e depois invertidos os documentos, assim coletando a média como taxa de identificação. O primeiro método aplicado foi a média de todos os blocos em um único bloco por escritor, a taxa de identificação foi de 90,29%.

Os resultados obtidos com a CNN e com as características com a média de todas as amostras estão na Tabela - 5.10 e também o desvio padrão desses experimentos.

**Tabela 5.10:** Resultados CNN e SVM média de todas as amostras.

Classificador	Taxa de Identificação	Desvio Padrão
CNN	82,06 %	4,32
SVM	90,29 %	2,74

A Tabela - 5.11 mostra os resultados utilizando a média de 1/3 e de 1/9 dos blocos. Para o segundo experimento utilizando o SVM com a média de 1/3 a taxa de identificação foi de 89,48 %. O terceiro experimento utilizando a média de 1/9 dos blocos foi que o teve a menor taxa de identificação que foi de 80,09 % e maior desvio padrão nos resultados da base CVL.

**Tabela 5.11:** Resultados SVM média de 1/3 e 1/9 das amostras.

Classificador	Blocos	Taxa de Identificação	Desvio Padrão
SVM	1/3	89,48 %	2,97
SVM	1/9	80,09 %	10,72

Os dois últimos experimentos realizados para a base CVL foram feitos por meio do cálculo da dissimilaridade para a média de 1/3 e de 1/9 dos blocos. Nesses experimentos também foram feitos com os documentos 3 no treino e 4 no teste e depois invertidos, assim também considerando a média com taxa de identificação. A Tabela - 5.12 demonstra os resultados para a média de 1/3 dos blocos que foi de 99,68 % e o melhor resultado para a base CVL que foi com 1/9 dos blocos que foi de 99,84 %.

**Tabela 5.12:** Resultados dissimilaridade média de 1/3 e 1/9 das amostras.

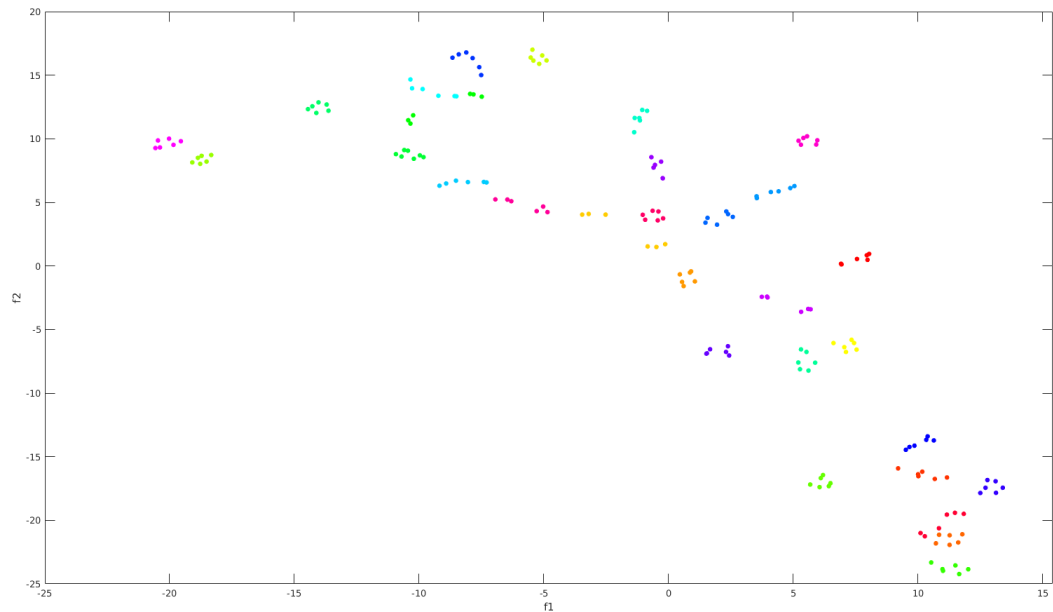
Classificador	Blocos	Taxa de Identificação	Desvio Padrão
SVM - Dissimilaridade	1/3	99,68 %	0,46
SVM - Dissimilaridade	1/9	99,84 %	0,26

O mesmo intuito de demonstrar como ficaram as características dos escritores no espaço feito com a base BFL utilizando t-SNE foi feito com a base CVL. Também foram utilizadas amostras de apenas 30 escritores da base CVL, pode-se observar na Figura - 5.3 o mesmo, como as amostras de cada escritor tenderam a ficar mais próximas e mais distantes das amostras de outros escritores.

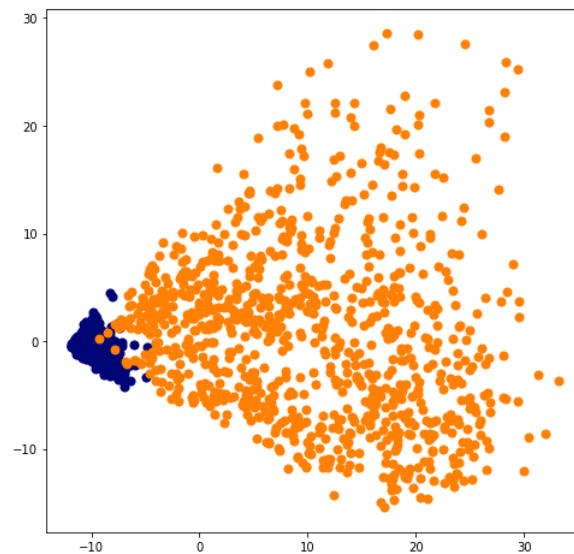
Com a dissimilaridade aplicada no documento 3 da base CVL, também foi utilizado o PCA para visualizar os dados no espaço bidimensional. Na Figura - 5.4 é possível observar que as amostras positivas ficam agrupadas e apenas algumas amostras negativas sobrepõem as positivas.

### 5.3 Discussão dos Resultados

O presente capítulo apresentou experimentos realizados com duas abordagens: a primeira utilizando a extração de características dos manuscritos de forma manual e a segunda o



**Figura 5.3:** Características de 30 escritores da base CVL separados por cores.



**Figura 5.4:** Características positivas (azul) e negativas (laranja) da base CVL.

uso de *deep learning* que “aprende” as características do problema de forma abstrata. Em ambos casos foram realizados experimentos com a dissimilaridade.

Na primeira abordagem, as características dos documentos foram obtidas por meio dos descritores de textura, mostrando-se eficazes nos experimentos nas duas bases de dados,

com a CVL foram selecionados de forma randômica 115 escritores e a LAMIS-MSHD, com 100 escritores, assim ambas bases ficaram com uma quantidade próxima de escritores.

Os resultados dos experimentos utilizando SVM apresentam taxas de 98,26% para a base CVL e de 92% para a base LAMIS-MSHD. Novos experimentos foram realizados com os mesmos conjuntos de treino e teste, porém com a aplicação da dissimilaridade, teve-se um aumento da acurácia, as taxas foram para 99,13% e para 99% para as bases CVL e LAMIS-MSHD respectivamente.

Já para a segunda abordagem utilizando CNN, foram realizados alguns experimentos preliminares com intuito de fazer uma melhor configuração dos hiperparâmetros da rede. Com a arquitetura definida, foram executados experimentos nas bases BFL com 315 escritores e CVL com 309 escritores, assim o número de escritores utilizados também foi bastante similar.

Os dois primeiros experimentos utilizaram a CNN para realizar as classificações em ambas bases de dados. Os pesos aprendidos por ela foram utilizados posteriormente para realizar experimentos com o SVM e com a dissimilaridade aplicada. Os melhores resultados dos experimentos foram utilizando a dissimilaridade, chegando a uma acurácia de 100% para a base BFL e de 99,84% na CVL.

Em ambas abordagens o uso da dissimilaridade apresentou melhoras significativas, indicando que o uso dela para a identificação de escritores é eficaz.



---

## Considerações Finais

---

As técnicas de *deep learning* se mostram cada dia mais acessíveis e eficazes, devido ao aumento do poder computacional das GPUs. Motivados por esse desempenho, ela foi utilizada nesta dissertação, para os experimentos de identificação de manuscritos. Mais especificamente, foi utilizada uma CNN para classificação e os pesos aprendidos na etapa de treinamento também foram utilizados para experimentos.

Antes de utilizar a CNN, foram realizados experimentos utilizando a abordagem tradicional de classificação, conhecida como extração manual ou engenharia de características. Os experimentos foram realizados por meio das características obtidas pelos descritores de textura LBP e o LPQ. Ambos descritores demonstraram resultados relevantes, principalmente quando se aplicou a dissimilaridade nas bases de dados. Os resultados chegaram próximos ao de trabalhos apresentados na literatura.

Visando melhorar os resultados, foram realizados os experimentos com CNN. Inicialmente, foi feita a configuração dos hiperparâmetros, definindo as dimensões das imagens de entrada como  $32 \times 32$  e o tamanho de 512 para a camada totalmente conectada. Com os ajustes feitos conseguiu-se definir uma arquitetura da rede com baixo *overfitting*, pensando nesse risco também foi utilizada a técnica de regularização *dropout*.

Assim, diversos experimentos foram realizados com a CNN. Os primeiros experimentos consistiram nela fazer a classificação das bases. Na sequência, por meio dos pesos aprendidos, foram extraídas características de outros documentos da mesma base do experimento em questão e eles foram aplicados no SVM. Por fim, com as características extraídas, também foi aplicado a dissimilaridade, nestes últimos experimentos obteve-se os melhores resultados desta dissertação.

No estado da arte descrito neste trabalho, alguns experimentos apresentados utilizavam as mesmas bases que os nossos experimentos. Os melhores resultados tiveram acurácia de 99,84% para a base de dados CVL contra 99,70% no trabalho de (Tang e Wu, 2016), cuja também foi utilizado CNN. Para a base BFL tivemos 100% de acerto contra 99,20% no trabalho de (Bertolini et al., 2013), o trabalho utilizava a abordagem tradicional de classificação. Assim, apontando que experimentos que não dependem de atributos extraídos por especialista (CNN) vem demonstrando bom resultados.

## 6.1 Trabalhos Futuros

Para trabalhos futuros, tem-se como objetivo avaliar outras bases de dados de manuscritos, tendo por finalidade verificar a eficácia do método de classificação utilizando CNN, aqui proposto. Fazer uma avaliação do impacto da utilização de palavras ou linhas dos documentos também seria possível. A arquitetura da rede também possui potencial para ser aprimorada. Outra proposta, seria a possibilidade de aplicar outros modelos de arquiteturas de CNN já pré-definidos, como as redes Inception (Szegedy et al., 2017), ImageNet (Krizhevsky et al., 2012), entre outras, visando realizar uma comparação direta com os resultados apresentados neste trabalho, contribuindo com o desenvolvimento contínuo do estado da arte.

## REFERÊNCIAS

---

- AHONEN, T.; HADID, A.; PIETIKAINEN, M. Face description with local binary patterns: Application to face recognition. *IEEE transactions on pattern analysis and machine intelligence*, v. 28, n. 12, p. 2037–2041, 2006.
- AL MÁADEED, S.; AYOUBY, W.; HASSAÏNE, A.; ALJAAM, J. M. Quwi: an arabic and english handwriting dataset for offline writer identification. In: *Frontiers in Handwriting Recognition (ICFHR), 2012 International Conference on*, IEEE, 2012, p. 746–751.
- ALMEIDA, P.; OLIVEIRA, L. S.; SILVA, E.; BRITTO, A.; KOERICH, A. Parking space detection using textural descriptors. In: *Systems, Man, and Cybernetics (SMC), 2013 IEEE International Conference on*, IEEE, 2013, p. 3603–3608.
- AMARAL, A. M. M. M. *Identificação de autoria de documentos manuscritos utilizando características grafométricas*. Tese de Doutorado, Pontifícia Universidade Católica do Paraná, 2014.
- AREL, I.; ROSE, D. C.; KARNOWSKI, T. P. Deep machine learning—a new frontier in artificial intelligence research [research frontier]. *IEEE Computational Intelligence Magazine*, v. 5, n. 4, p. 13–18, 2010.
- ARORA, H.; LEE, S.; SRIHARI, S. N.; CHA, S.-H. Individuality of handwriting. *Journal of forensic science*, v. 47, n. 4, p. 1–17, 2002.
- BARANOSKI, F. L.; OLIVEIRA, L. S.; JUSTINO, E. J. Writer identification based on forensic science approach. In: *Conferencia Latinoamericana de Informática (CLEI2007)*, 2007, p. 25–32.
- BENGIO, Y.; ET AL. Learning deep architectures for ai. *Foundations and trends® in Machine Learning*, v. 2, n. 1, p. 1–127, 2009.

BERTOLINI, D. Identificação e verificação de escritores usando características texturais e dissimilaridade. 2014. 117 f. Tese (Doutorado em Ciência da Computação) - Universidade Federal do Paraná, Curitiba, PR, BR, 2014.

BERTOLINI, D.; OLIVEIRA, L. S.; COSTA, Y. M.; HELAL, L. G. Knowledge transfer for writer identification. In: *Iberoamerican Congress on Pattern Recognition*, Springer, 2017, p. 102–110.

BERTOLINI, D.; OLIVEIRA, L. S.; JUSTINO, E.; SABOURIN, R. Texture-based descriptors for writer identification and verification. *Expert Systems with Applications*, v. 40, n. 6, p. 2069–2080, 2013.

BRADSKI, G. R.; YEO, B.-L.; YEUNG, M. M. Interface using pattern recognition and tracking. US Patent 6,363,160, 2002.

BRIAND, L. C.; BASILI, V. R.; THOMAS, W. M. A pattern recognition approach for software engineering data analysis. *IEEE transactions on software engineering*, v. 18, n. 11, p. 931–942, 1992.

CHA, S.-H.; SRIHARI, S. N. Writer identification: statistical analysis and dichotomizer. In: *Joint IAPR International Workshops on Statistical Techniques in Pattern Recognition (SPR) and Structural and Syntactic Pattern Recognition (SSPR)*, Springer, 2000, p. 123–132.

CHAHI, A.; RUICHEK, Y.; TOUAHNI, R.; ET AL. Block wise local binary count for off-line text-independent writer identification. *Expert Systems with Applications*, v. 93, p. 1–14, 2018.

CHANG, C.-C.; LIN, C.-J. Libsvm: a library for support vector machines. *ACM Transactions on Intelligent Systems and Technology (TIST)*, v. 2, n. 3, p. 27, 2011.

CHEN, S.; WANG, Y.; LIN, C.-T.; CAO, Z. Semi-supervised feature learning for off-line writer identifications. *arXiv preprint arXiv:1807.05490*, 2018.

CHOLLET, F.; ET AL. Keras. <https://keras.io>, 2015.

CHRISTLEIN, V.; BERNECKER, D.; HÖNIG, F.; MAIER, A.; ANGELOPOULOU, E. Writer identification using gmm supervectors and exemplar-svms. *Pattern Recognition*, v. 63, p. 258–267, 2017.

- COSTA, Y. M.; OLIVEIRA, L.; KOERICH, A. L.; GOUYON, F.; MARTINS, J. Music genre classification using lbp textural features. *Signal Processing*, v. 92, n. 11, p. 2723–2737, 2012.
- DENG, L.; YU, D.; ET AL. Deep learning: methods and applications. *Foundations and Trends® in Signal Processing*, v. 7, n. 3–4, p. 197–387, 2014.
- DING, X.; ZHANG, Y.; LIU, T.; DUAN, J. Deep learning for event-driven stock prediction. In: *Twenty-Fourth International Joint Conference on Artificial Intelligence*, 2015.
- DJEDDI, C.; GATTAL, A.; SOUCI-MESLATI, L.; SIDDIQI, I.; CHIBANI, Y.; EL ABED, H. Lamis-mshd: A multi-script offline handwriting database. In: *Frontiers in Handwriting Recognition (ICFHR), 2014 14th International Conference on*, IEEE, 2014, p. 93–97.
- DU, L.; YOU, X.; XU, H.; GAO, Z.; TANG, Y. Wavelet domain local binary pattern features for writer identification. In: *Pattern Recognition (ICPR), 2010 20th International Conference on*, IEEE, 2010, p. 3691–3694.
- FIEL, S.; SABLATNIG, R. Writer identification and writer retrieval using the fisher vector on visual vocabularies. In: *Document Analysis and Recognition (ICDAR), 2013 12th International Conference on*, IEEE, 2013, p. 545–549.
- FREITAS, C.; OLIVEIRA, L. S.; SABOURIN, R.; BORTOLOZZI, F. Brazilian forensic letter database. In: *11th International workshop on frontiers on handwriting recognition, Montreal, Canada*, 2008.
- GROSICKI, E.; CARRE, M.; BRODIN, J.-M.; GEOFFROIS, E. Rimes evaluation campaign for handwritten mail processing. In: *ICFHR 2008: 11th International Conference on Frontiers in Handwriting Recognition*, Concordia University, 2008, p. 1–6.
- HAFEMANN, L. G. An analysis of deep neural networks for texture classification. 2014. 89 f. Dissertação (Mestrado em Ciência da Computação) - Universidade Federal do Paraná, Curitiba, PR, BR, 2014.
- HANNAD, Y.; SIDDIQI, I.; EL KETTANI, M. E. Y. Writer identification using texture descriptors of handwritten fragments. *Expert Systems with Applications*, v. 47, p. 14–22, 2016.

- HANUSIAK, R. K.; OLIVEIRA, L. S.; JUSTINO, E.; SABOURIN, R. Writer verification using texture-based features. *International Journal on Document Analysis and Recognition (IJ DAR)*, v. 15, n. 3, p. 213–226, 2012.
- HELAL, L. G.; MALDONADO, Y.; DA COSTA, G.; GONÇALVES, D. B.; FELIPE, G. Z. Off-line writer identification using handcrafted features versus convnets. In: *2017 36th International Conference of the Chilean Computer Science Society (SCCC)*, IEEE, 2017.
- IDICULA, S. M. A survey on writer identification schemes. *Writer*, v. 25, n. 12, p. 15, 2011.
- JAIN, A. K.; ROSS, A.; PANKANTI, S. Biometrics: a tool for information security. *IEEE transactions on information forensics and security*, v. 1, n. 2, p. 125–143, 2006.
- JOLLIFFE, I. Principal component analysis. In: *International encyclopedia of statistical science*, Springer, p. 1094–1096, 2011.
- JUSTINO, E. A análise de documentos questionados. *Monografia para concurso de professor titular, Pontifícia Universidade Católica do Paraná, 74p*, 2002.
- KAHNG, M.; ANDREWS, P. Y.; KALRO, A.; CHAU, D. H. P. A ctiv is: Visual exploration of industry-scale deep neural network models. *IEEE transactions on visualization and computer graphics*, v. 24, n. 1, p. 88–97, 2018.
- KHAN, F. A.; TAHIR, M. A.; KHELIFI, F.; BOURIDANE, A. Offline text independent writer identification using ensemble of multi-scale local ternary pattern histograms. In: *Visual Information Processing (EUVIP), 2016 6th European Workshop on*, IEEE, 2016, p. 1–6.
- KHAN, F. A.; TAHIR, M. A.; KHELIFI, F.; BOURIDANE, A.; ALMOTAERYI, R. Robust off-line text independent writer identification using bagged discrete cosine transform features. *Expert Systems with Applications*, v. 71, p. 404–415, 2017.
- KITTLER, J.; HATEF, M.; DUIN, R. P.; MATAS, J. On combining classifiers. *IEEE transactions on pattern analysis and machine intelligence*, v. 20, n. 3, p. 226–239, 1998.
- KLEBER, F.; FIEL, S.; DIEM, M.; SABLATNIG, R. Cvl-database: An off-line database for writer retrieval, writer identification and word spotting. In: *Document Analysis and Recognition (ICDAR), 2013 12th International Conference on*, IEEE, 2013, p. 560–564.

- KRIZHEVSKY, A.; SUTSKEVER, I.; HINTON, G. E. Imagenet classification with deep convolutional neural networks. In: *Advances in neural information processing systems*, 2012, p. 1097–1105.
- LECUN, Y.; BENGIO, Y.; HINTON, G. Deep learning. *Nature*, v. 521, n. 7553, p. 436–444, 2015.
- LECUN, Y.; BOTTOU, L.; BENGIO, Y.; HAFFNER, P. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, v. 86, n. 11, p. 2278–2324, 1998.
- LI, F.-F.; KARPATHY, A.; JOHNSON, J. Cs231n: Convolutional neural networks for visual recognition. Online, acessado em Abril de 2017, 2015.  
Disponível em <http://cs231n.github.io/convolutional-networks/>
- LI, W.; CHEN, H. Identifying top sellers in underground economy using deep learning-based sentiment analysis. In: *2014 IEEE Joint Intelligence and Security Informatics Conference*, IEEE, 2014, p. 64–67.
- MAATEN, L. v. D.; HINTON, G. Visualizing data using t-sne. *Journal of machine learning research*, v. 9, n. Nov, p. 2579–2605, 2008.
- MÄENPÄÄ, T. *The local binary pattern approach to texture analysis: extensions and applications*. Oulun yliopisto Oulu, 2003.
- MAHMOUD, S. A.; AHMAD, I.; ALSHAYEB, M.; AL-KHATIB, W. G.; PARVEZ, M. T.; FINK, G. A.; MÄRGNER, V.; EL ABED, H. Khatt: Arabic offline handwritten text database. In: *Frontiers in Handwriting Recognition (ICFHR), 2012 International Conference on*, IEEE, 2012, p. 449–454.
- MANGASARIAN, O. L.; SETIONO, R.; WOLBERG, W. Pattern recognition via linear programming: Theory and application to medical diagnosis. *Large-scale numerical optimization*, p. 22–31, 1990.
- MARTI, U.-V.; BUNKE, H. The iam-database: an english sentence database for offline handwriting recognition. *International Journal on Document Analysis and Recognition*, v. 5, n. 1, p. 39–46, 2002.
- MARTIN, K. D.; KIM, Y. E. Musical instrument identification: A pattern-recognition approach. *The Journal of the Acoustical Society of America*, v. 104, n. 3, p. 1768–1768, 1998.

- MILGRAM, J.; CHERIET, M.; SABOURIN, R. “one against one” or “one against all”: Which one is better for handwriting recognition with svms? In: *tenth international workshop on Frontiers in handwriting recognition*, SuviSoft, 2006.
- MOHAMMADI, M.; MOGHADDAM, M. E.; SAADAT, S. A multi-language writer identification method based on image mining and genetic algorithm techniques. *Soft Computing*, p. 1–15, 2018.
- NEWELL, A. J.; GRIFFIN, L. D. Writer identification using oriented basic image features and the delta encoding. *Pattern Recognition*, v. 47, n. 6, p. 2255–2265, 2014.
- NGUYEN, H. T.; NGUYEN, C. T.; INO, T.; INDURKHYA, B.; NAKAGAWA, M. Text-independent writer identification using convolutional neural network. *Pattern Recognition Letters*, 2018.
- OJALA, T.; PIETIKÄINEN, M.; HARWOOD, D. A comparative study of texture measures with classification based on featured distributions. *Pattern recognition*, v. 29, n. 1, p. 51–59, 1996.
- OJALA, T.; PIETIKAINEN, M.; MAENPAA, T. Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *IEEE Transactions on pattern analysis and machine intelligence*, v. 24, n. 7, p. 971–987, 2002.
- OJANSIVU, V.; HEIKKILÄ, J. Blur insensitive texture classification using local phase quantization. In: *International conference on image and signal processing*, Springer, 2008, p. 236–243.
- OTSU, N. A threshold selection method from gray-level histograms. *Automatica*, v. 11, n. 285-296, p. 23–27, 1975.
- PARKHI, O. M.; VEDALDI, A.; ZISSERMAN, A.; ET AL. Deep face recognition. In: *bmvc*, 2015, p. 6.
- PAVELEC, D.; JUSTINO, E.; BATISTA, L. V.; OLIVEIRA, L. S. Author identification using writer-dependent and writer-independent strategies. In: *Proceedings of the 2008 ACM symposium on Applied computing*, ACM, 2008, p. 414–418.
- PECHWITZ, M.; MADDOURI, S. S.; MÄRGNER, V.; ELLOUZE, N.; AMIRI, H.; ET AL. Ifn/enit-database of handwritten arabic words. In: *Proc. of CIFED*, Citeseer, 2002, p. 127–136.



- PEDRINI, H.; SCHWARTZ, W. R. *Análise de imagens digitais: princípios, algoritmos e aplicações*. Thomson Learning, 2008.
- PEKALSKA, E.; DUIN, R. P. Dissimilarity representations allow for building good classifiers. *Pattern Recognition Letters*, v. 23, n. 8, p. 943–956, 2002.
- PEREZ, L.; WANG, J. The effectiveness of data augmentation in image classification using deep learning. *arXiv preprint arXiv:1712.04621*, 2017.
- RIVARD, D. *Multi-feature approach for writer-independent offline signature verification*. Tese de Doutorado, École de technologie supérieure, 2010.
- SCHERER, D.; MÜLLER, A.; BEHNKE, S. Evaluation of pooling operations in convolutional architectures for object recognition. In: *International Conference on Artificial Neural Networks*, Springer, 2010, p. 92–101.
- SERMANET, P.; LECUN, Y. Traffic sign recognition with multi-scale convolutional networks. In: *Neural Networks (IJCNN), The 2011 International Joint Conference on*, IEEE, 2011, p. 2809–2813.
- SINGH, P.; ROY, P. P.; RAMAN, B. Writer identification using texture features: A comparative study. *Computers & Electrical Engineering*, v. 71, p. 1–12, 2018.
- SZEGEDY, C.; IOFFE, S.; VANHOUCKE, V.; ALEMI, A. A. Inception-v4, inception-resnet and the impact of residual connections on learning. In: *AAAI*, 2017, p. 12.
- TANG, Y.; WU, X. Text-independent writer identification via cnn features and joint bayesian. In: *Frontiers in Handwriting Recognition (ICFHR), 2016 15th International Conference on*, IEEE, 2016, p. 566–571.
- TRIPPI, R. R.; TURBAN, E. *Neural networks in finance and investing: Using artificial intelligence to improve real world performance*. McGraw-Hill, Inc., 1992.
- VAPNIK, V. *The nature of statistical learning theory*. Springer science & business media, 1995.
- WONG, T. Y.; BRESSLER, N. M. Artificial intelligence with deep learning technology looks into diabetic retinopathy screening. *Jama*, v. 316, n. 22, p. 2366–2367, 2016.

XING, L.; QIAO, Y. Deepwriter: A multi-stream deep cnn for text-independent writer identification. In: *Frontiers in Handwriting Recognition (ICFHR), 2016 15th International Conference on*, IEEE, 2016, p. 584–589.

YANG, W.; JIN, L.; LIU, M. Deepwriterid: An end-to-end online text-independent writer identification system. *IEEE Intelligent Systems*, v. 31, n. 2, p. 45–53, 2016.

YILMAZ, M. B.; YANIKOGLU, B.; TIRKAZ, C.; KHOLMATOV, A. Offline signature verification using classifier combination of hog and lbp features. In: *Biometrics (IJCB), 2011 International Joint Conference on*, IEEE, 2011, p. 1–7.