



USO DE REDES PIRAMIDAIS DE INIBIÇÃO LATERAL NA DETECÇÃO DE ANOMALIAS ATRAVÉS DA ANÁLISE DE SINAIS

Trabalho de Conclusão de Curso

Engenharia da Computação

Carlos Henrique Macedo dos Santos
Orientador: Prof. Carmelo José Albanez Bastos Filho
Co-orientador: Rodrigo de Paula Monteiro



**Universidade de Pernambuco
Escola Politécnica de Pernambuco
Graduação em Engenharia de Computação**

**CARLOS HENRIQUE MACEDO DOS
SANTOS**

**USO DE REDES PIRAMIDAIIS DE
INIBIÇÃO LATERAL NA DETECÇÃO DE
ANOMALIAS ATRAVÉS DA ANÁLISE
DE SINAIS**

Monografia apresentada como requisito parcial para obtenção do diploma de Bacharel em Engenharia de Computação pela Escola Politécnica de Pernambuco – Universidade de Pernambuco.

Recife, Julho 2019.

1/41

Autorização de publicação de PFC

Eu, **Carlos Henrique Macedo dos Santos** autor(a) do projeto de final de curso intitulado: **USO DE REDES PIRAMIDAIAS DE INIBIÇÃO LATERAL NA DETECÇÃO DE ANOMALIAS ATRAVÉS DA ANÁLISE DE SINAIS**; autorizo a publicação de seu conteúdo na internet nos portais da Escola Politécnica de Pernambuco e Universidade de Pernambuco.

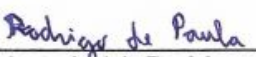
O conteúdo do projeto de final de curso é de responsabilidade do autor.



Carlos Henrique Macedo dos Santos



Orientador(a): **Carmelo Jose Albanéz Bastos Filho**



Coorientador(a): **Rodrigo de Paula Monteiro**



Prof. de TCC: **Daniel Augusto Ribeiro Chaves**

Data: 03/07/2019

MONOGRAFIA DE FINAL DE CURSO

Avaliação Final (para o presidente da banca)*

No dia 03/07/2019, às 16h, reuniu-se para deliberar sobre a defesa da monografia de conclusão de curso do(a) discente **CARLOS HENRIQUE MACEDO DOS SANTOS**, orientado(a) pelo(a) professor(a) **CARMELO JOSE ALBANEZ BASTOS FILHO**, sob título **USO DE REDES PIRAMIDAIAS DE INIBIÇÃO LATERAL NA DETECÇÃO DE ANOMALIAS ATRAVÉS DA ANÁLISE DE SINAIS**, a banca composta pelos professores:

ALEXANDRE MAGNO ANDRADE MACIEL (PRESIDENTE)
CARMELO JOSE ALBANEZ BASTOS FILHO (ORIENTADOR)

Após a apresentação da monografia e discussão entre os membros da Banca, a mesma foi considerada:

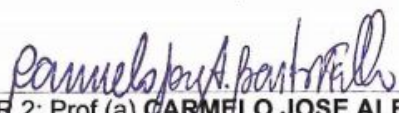
Aprovada Aprovada com Restrições* Reprovada

e foi-lhe atribuída nota: 9,0 (NOVE)

*(Obrigatório o preenchimento do campo abaixo com comentários para o autor)

O(A) discente terá 15 dias para entrega da versão final da monografia a contar da data deste documento.


AVALIADOR 1: Prof (a) **ALEXANDRE MAGNO ANDRADE MACIEL**


AVALIADOR 2: Prof (a) **CARMELO JOSE ALBANEZ BASTOS FILHO**


AVALIADOR 3: Prof (a)

* Este documento deverá ser encadernado juntamente com a monografia em versão final.

*Dedico essa monografia à todos que me apoiaram no caminho que trilhei até aqui
pois sem eles nada disso seria possível.*

Agradecimentos

Primeiramente, gostaria de agradecer à toda a equipe universidade pela oportunidade de receber uma formação de qualidade. Aos professores, que tiveram e dedicação em lecionar durante minha estadia na universidade, mesmo não sendo um trabalho fácil.

Ao meu orientador Carmelo José Albanez Filho, que me permitiu trabalhar neste projeto sendo acompanhado por ele e principalmente ao meu co-orientador Rodrigo de Paula Monteiro por sua inabalável paciência em explicar, re-explicar, contribuir e revisar alguns pontos do projeto.

À minha família e amigos que nunca deixaram de acreditar no meu potencial.

Resumo

Esta monografia visa apresentar a viabilidade do uso da rede neural piramidal de inibição lateral (LIPNet) para detecção de padrões em aplicações de análise de sinais temporais no mundo real com uma precisão razoável. Nesta pesquisa é evidenciado o poder de classificação desta rede para problemas mecânicos e o baixo poder computacional necessário para tal. A LIPNet é uma rede desenvolvida por Fernandes et al., que foi inicialmente usada para tratar de classificações binárias em imagens. Neste trabalho as imagens usadas como entrada da rede foram formadas a partir da transformação de sinais de imagem de espectrogramas. Então, o classificador retorna a probabilidade de uma imagem pertencer a uma das classes treinadas. Neste problema, foram avaliados dez tipos de sinais individualmente considerando uma base de dados de falhas em caixas de marcha. Foi obtida uma acurácia em torno de 85% para cem épocas na fase de treinamento. Estes, por sua vez, ocorreram ao usar configurações mais genéricas do modelo e têm uma possibilidade de melhora se os parâmetros da rede forem ajustados especificamente para cada classe.

Abstract

This project aims to present the feasibility of using Lateral Inhibition Pyramidal Neural Networks (LIPNet) for detecting patterns in real-world temporal signal analysis applications with reasonable accuracy. In this research it is evidenced this powerful classifier for mechanical problems and the low computational cost necessary for such. LIPNet is a network developed by Fernandes et al., which was initially used to deal with binary classifications in images. The images used as input of the network were formed from the transformation of signals into spectrogram images. Then, the classifier returns the probability of an image belongs to one of the trained classes. In this problem, we evaluated ten types of signals individually considering a database with failures in gearboxes. We obtained an accuracy of around 85% for one hundred epochs in the training phase. These results were obtained by using more generic configurations of the model, and improvement can be achieved with the adjustment of the network parameters for each class.

Sumário

Índice de Figuras	9
Índice de Tabelas	9
Índice de Equações	11
Tabela de Símbolos e Siglas	12
Capítulo 1	13
Introdução	13
Motivação	13
Objetivos	16
Objetivos Gerais	16
Objetivos Específicos	16
Estrutura da Monografia	17
Capítulo 2	19
Fundamentação Teórica	19
Redes neurais de aprendizagem profunda	19
Rede Piramidal (PyraNet)	19
Lateral Inhibition Pyramidal Neural Network (LIPNet)	20
Transformada de Fourier para espectrogramas	22
Capítulo 3	24
Metodologia	24
Aquisição de dados	24
Treinamento da rede LIPNet	27
Capítulo 4	31
Resultados	31
Capítulo 5	37
Considerações Finais	37
Conclusão	37
Trabalhos Futuros	37
Bibliografia	39

Índice de Figuras

Figura 01. Exemplo de realce nas características quando processado um sinal	14
Figura 02. Representação gráfica do modelo PyraNet.	20
Figura 03. Arquitetura da LIPNet.	21
Figura 04. Exemplo da base de dados Morph-II.	22
Figura 05. Esquema de extração de sinais na caixa de marcha.	25
Figura 06. Níveis de severidade na quebra dos dentes das engrenagens.	26
Figura 07. Exemplo de sinais de entrada apresentados a LIPNet.	26
Figura 08. Processo de redimensionamento de uma imagem da classe P1.	27
Figura 09. Representação visual da configuração 2 da LIPNet.	29
Figura 10. Representação visual das configurações 3 e 4 da LIPNet.	29

Índice de Tabelas

Tabela 1. Configuração dos parâmetros da LIPNet	30
Tabela 2. Acurácias obtidas para diferentes configurações da LIPNet	33
Tabela 3. Tempos de treinamento e classificação para cada configuração testada	34
Tabela 4. Resultados das simulações da LIPNet em comparação com a DCNN para imagens em escalas de cinza	35
Tabela 5. Resultados das simulações da LIPNet em comparação com a DCNN para imagens coloridas	36

Índice de Equações

Equação 1. Neurônios por camada no modelo LIPNet-PSO

28

Tabela de Símbolos e Siglas

1-D – Uma dimensão

2-D – Duas dimensões

CNN – *Convolutional Neural Network*

CPU – *Central Processing Unit*

DAE – *Deep Auto-encoder*

DBN – *Deep Belief Network*

DCNN – *Deep Convolutional Neural Network*

DFT – *Discrete Fourier Transform*

DRN – *Deep Residual Network*

FFT – *Fast Fourier Transform*

FT – *Fourier Transform*

LIPNet – *Lateral Inhibition Pyramidal Neural Network*

ML – *Machine Learning*

PSO – *Particle Swarm Optimization*

PyraNet – *Pyramidal Neural Network*

SPNN – *Structured Pyramidal Neural Network*

STFT – *Short Time Fourier Transform*

SVM – *Support Vector Machine*

Capítulo 1

Introdução

Este capítulo encontra-se dividido em três seções. Inicialmente expõe-se a motivação do trabalho com uma breve descrição do que será abordado. Após isso, são discutidos quais são os principais objetivos da pesquisa e, por fim, é apresentada como os demais capítulos são estruturados.

1.1 Motivação

Os sinais podem descrever uma grande variedade de fenômenos físicos. Embora possam ser representados de diferentes maneiras, a informação do sinal está sempre contida em algum tipo de variação (como velocidade ou frequência sobre tempo). Com isto, por exemplo, é possível modelar o comportamento da força e velocidade de um automóvel em função do tempo [10]. Também existem sinais que são gerados e transmitidos com informações codificadas, exemplificando aqueles usados em comunicações digitais para transferência de áudio ou vídeo.

Sinais podem estar compreendidos no domínio do tempo ou na frequência. Quando realizado estudos sobre um sinal, por vezes, é conveniente que este se encontre no domínio da frequência, pois, este domínio tem capacidade de fornecer mais informações sobre a composição de um sinal. Além disso, operações sobre este sinal são mais simples [10].

Estudos e análises dos sinais auxiliam o entendimento de um fenômeno quando observado o seu comportamento, que pode ser aleatório ou ter uma assinatura própria. Um exemplo de análise é disposto a Figura 01, que apresenta dois sinais em diferentes domínios.

A parte superior da Figura 01 estão dois sinais com suas amplitudes em função do tempo, apenas essas informações não tornam evidentes as diferenças entre A e B. Na parte inferior da figura estão os mesmos sinais processados em

função das frequências, as diferenças entre A e B tornam-se perceptíveis, além disso, os sinais apresentam padrões (assinaturas), como as linhas horizontais e suas repetições ao longo de seu espectro, a intensidade de apresentada por cada ponto e a forma que é composta as cores que preenchem a imagem.

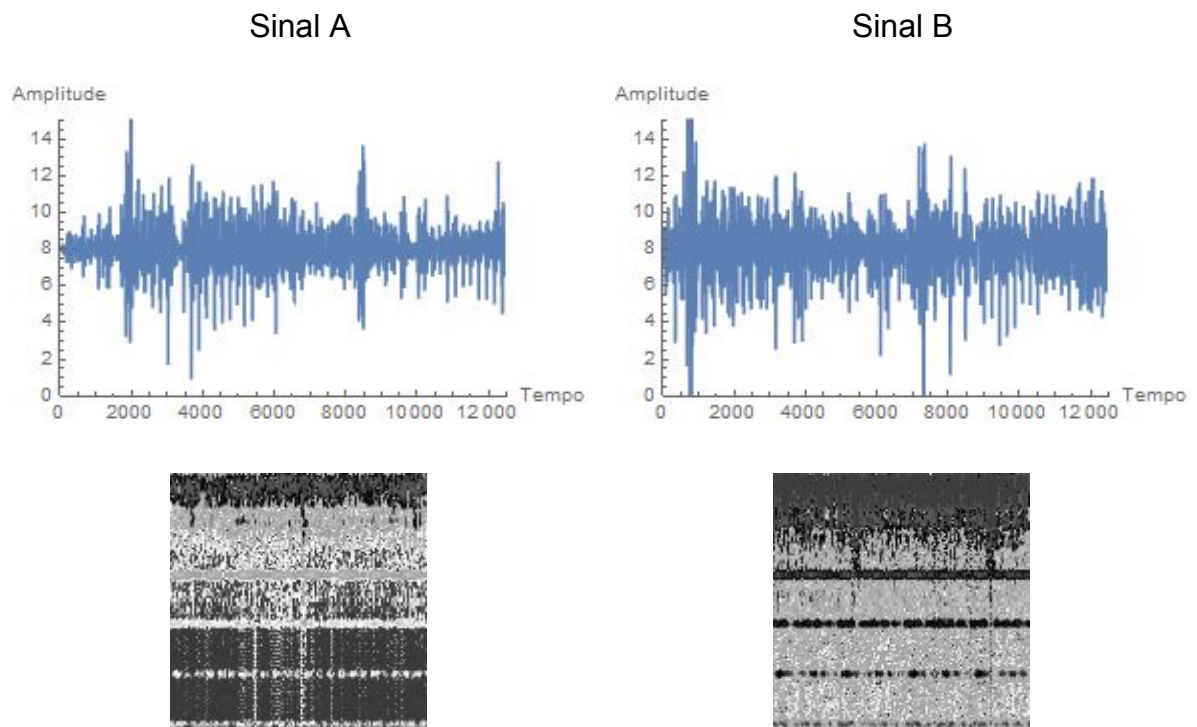


Figura 01. Exemplo de realce nas características quando processado um sinal

Em um sistema real, como, por exemplo, um motor, é possível a extração de sinais através de diversos componentes (peças) que compõem o sistema, estas extrações podem ser a partir de suas vibrações e emissões acústicas ou oscilatórios em decorrência da interação entre os componentes.

Neste maquinário espera-se que os sinais emitidos pelos componentes se comportem de forma previsível, e a isto atribui-se o nome de assinatura, podendo ser alterada quando há algum tipo de desgaste nestes componentes. Desta forma é aceitável inferir que qualquer assinatura que seja diferente da esperada pode consistir de uma falha no sistema, ou, quando analisado estes sinais, uma anomalia.

Anomalia são padrões em dados que não estão em conformidade com uma noção bem definida de comportamento normal. Uma aproximação direta na detecção é definir uma região que representa o comportamento normal e declarar qualquer observação nos dados que não pertencem a esta região normal como anomalia [1].

Com isto, podemos dizer que qualquer dado extraído no funcionamento correto do sistema é considerado normal, todos os demais são considerados dados anômalos. Existem várias técnicas que já foram aplicadas para detecção e classificação deste tipo de dado, como, por exemplo, algoritmos de clusterização, análises estatísticas, redes bayesianas, redes neurais artificiais entre outras técnicas de aprendizagem de máquina [1].

Com o advento de algoritmos de aprendizagem de máquina (*Machine learning algorithms* ou ML), os computadores ganharam algumas funções que antes não se imaginavam possíveis ser realizados por máquinas, como a capacidade de reconhecer objetos em uma imagem uma vez que lhe é dado exemplos similares [7].

Atualmente, são apresentadas na literatura algumas aproximações de ML que versam sobre o uso de redes neurais artificiais de aprendizagem profunda (ou *deep learning*) para inferir acerca de possíveis falhas em um sistema. Estas técnicas já se mostraram bastante eficientes em se tratando de classificação [8], clusterização [17], regressão [6], detecção de anomalias [18], entre outros.

Por tal motivo, é esperado que alguma destas técnicas de aprendizagem de máquina tenha a capacidade de decidir sobre a pertinência da assinatura de uma anomalia em um sinal para uma determinada categoria.

Ainda se encontra literatura que existem algumas aproximações já realizadas com o uso de diversas arquiteturas de redes para analisar o padrão dos sinais, como a CNN (*convolutional neural network*), DBN (*deep belief network*), DAE (*deep auto-encoder*) e a DRN (*deep residual network*) [18].

A depender da profundidade de uma rede neural, o treino e a classificação podem ser mais custosas, demandando um maior tempo de execução, tornando-as inviáveis para alguns problemas reais, onde, por vezes, é usado um *hardware*

limitado. Por isto, este trabalho objetiva adaptar uma rede piramidal como arquitetura alternativa às supracitadas, dispondo de seu processo de classificação mais simplificado, visando um baixo custo computacional enquanto almejando uma acurácia comparável às demais.

Dentre as arquiteturas existentes de redes piramidais, a selecionada para experimentação foi a LIPNet (*Lateral Inhibition Pyramidal Neural Network*), pois durante o treino ela oferece mecanismos internos para realce de características na imagem (*Inibição lateral e sobreposição de campos*), além disso a LIPNet estende os conceitos da PyraNet, podendo ter desempenho idêntico quando bem parametrizada.

1.2 Objetivos

1.2.1 Objetivos Gerais

Este projeto apresenta a viabilidade na detecção de anomalias em sinais emitidos na interação das peças de um maquinário através do uso de redes de aprendizagem profunda com o uso das arquiteturas hoje conhecidas de redes piramidais, que são bastante eficientes em comparação com modelos mais tradicionais [2][11]. A grande vantagem das redes piramidais é que elas oferecem baixos tempos de execução durante classificação e o ganho no uso delas, em comparação com outras aproximações mais custosas, ocorre em detrimento da acurácia [15].

Esta solução visa evidenciar esta redução no custo computacional (tempo de execução e complexidade computacional), enquanto o impacto na classificação se mantém acima do aceitável.

1.2.2 Objetivos Específicos

Como objetivos específicos, destacam-se:

- Adaptar a LIPNet, modelo de rede neural desenvolvido por Fernandes *et al.* [2] ao problema de classificação em sinais temporais emitidos por um sistema mecânico (caixa de marcha).
- Mostrar que neste modelo possui uma acurácia próxima a obtida por Monteiro [8] em sua aproximação com o uso da DCNN.

1.3 Estrutura da Monografia

Esta monografia foi dividida em 5 capítulos, sendo este destinado à introdução do tema, as motivações e a estruturação do documento.

Neste primeiro capítulo é apresentado um breve panorama sobre os conceitos que envolvem este projeto quanto a análises de sinais com o uso de *deep learning*. Também é discutido e justificado o uso de redes piramidais como possível alternativa para estes problemas.

No segundo capítulo são introduzidos os conceitos teóricos úteis para o entendimento do processo apresentado na metodologia que fundamentam a construção do projeto. São eles o pré-processamento dos dados e o funcionamento dos processos internos fundamentais da LIPNet.

No terceiro, é destrinchada a metodologia seguida durante o experimento. Neste capítulo será explanado como foram adquiridos os dados para a experimentação, bem como sua transformação e entrada na rede. Também é discutido e justificado o uso das configurações adotadas na LIPNet e nos testes paramétricos, e, por fim, os recursos da máquina utilizada no experimento.

No quarto capítulo estão presentes os resultados obtidos através da conclusão do processo de treinamento de cada simulação da LIPNet. Também são discutidas e apresentadas justificativas referentes aos resultados e comparações com demais arquiteturas.

Por fim, no quinto capítulo, encontram-se as conclusões do trabalho, neste são comentadas as possibilidades de alternativas e melhorias no processo de

classificação da LIPNet. Também são dadas sugestões de futuras pesquisas com esta arquitetura.

Capítulo 2

Fundamentação Teórica

Este capítulo tem por objetivo explicar detalhadamente os conceitos que serão trabalhados neste projeto. São apresentados inicialmente alguns conceitos essenciais para o entendimento da metodologia aplicada, trazendo uma breve introdução à PyraNet, que é a rede que inspirou a LIPNet. Em seguida será explicado o funcionamento da LIPNet e, por fim, a técnica utilizada para a transformação de um *array* de números em imagem que é usada na transformação dos dados extraídos para alimentar a entrada da rede.

2.1 Redes neurais de aprendizagem profunda

O conceito de redes neurais surge inicialmente em 1943 com McCulloch e Pitts, ambos acadêmicos que introduziram a idéia de modelar computacionalmente o comportamento de um neurônio [16]. As pesquisas continuaram desenvolvendo este conceito até o ponto em que surgiram as primeiras aplicações de redes neurais artificiais por Frank Rosenblatt em 1958 [16], onde haviam diversos neurônios interconectados e um algoritmo de treinamento capaz de, dada uma entrada, calibrar a intensidade dessa conexão [13]. Estes estudos originaram uma poderosa máquina de inferências, tendo aplicações que solucionam um amplo espectro de problemas. O conceito de rede de aprendizagem profunda refere-se a uma rede cuja complexidade de sua formação e configurações das conexões dos seus neurônios aumentam por quantidade de camadas, formas de treinar ou propagação de erros.

2.1.1 Rede Piramidal (PyraNet)

Rede piramidal é uma arquitetura de rede neural multi-níveis na qual os neurônios são ordenados de maneira hierárquica em uma estrutura em cascata. Nesta arquitetura existem camadas de neurônios 2-D para extração de

características e camadas 1-D para auxiliar na classificação. A primeira camada da rede (2-D) é conectada diretamente à imagem e sobre ela são postas as demais camadas 2-D. Ao fim da última camada 2-D está a primeira camada 1-D, que é seguida pelas demais camadas 1-D. Cada neurônio 2-D é conectado a uma região retangular específica (campo receptivo) da camada anterior. Neurônios 2-D fazem um somatório ponderado com as entradas do seu campo receptivo e, ao ser aplicada uma função de ativação não-linear, produzem a saída. Os neurônios 1-D são agrupados ao fim da rede de maneira que possam processar as características extraídas pelas camadas 2-D e suas saídas variam com o tipo de padrão inserido como entrada desta rede [11]. Esta arquitetura pode ser vista na Figura 02.

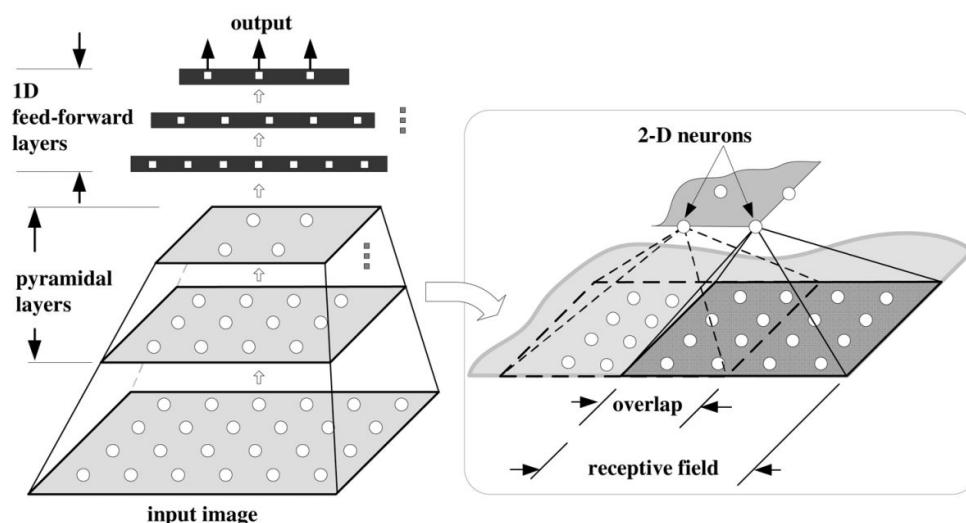


Figura 02. Representação gráfica do modelo PyraNet. Extraído de [11].

2.1.2 Lateral Inhibition Pyramidal Neural Network (LIPNet)

A LIPNet é uma arquitetura de redes piramidais de aprendizagem profunda introduzida em 2013 por Fernandes *et al.* [2] e é inspirada na PyraNet. Ela também possui os conceitos de camadas 1-D e camadas 2-D. No entanto, a LIPNet inclui a inibição lateral, mecanismo usado para controlar o processo de extração de características em uma dada imagem.

Inibição lateral, segundo Grigorescu *et al.* [5], conforme citado por Fernandes [2], é útil para suprimir contornos dentro de uma região que tenha a mesma textura,

então, ao aplicar esta técnica é possível não só dizer se uma informação é útil, mas também se ela prejudica o processo de classificação.

Enquanto a PyraNet possui parâmetros para tamanho de campo receptivo, tamanho de sobreposição entre campos e número de neurônios para cada camada, a LIPNet acrescenta a esta lista os parâmetros relacionados ao campo inibitório. São eles o tamanho deste campo e o peso da inibição.

A rede inteira conecta-se em cascata (ou seja, a saída de uma camada será a entrada da próxima). A primeira camada 2-D age como uma camada identidade, ou seja, para um neurônio fixado em uma certa coordenada desta camada, o seu valor de saída é o mesmo valor do pixel de mesma coordenada em uma dada imagem.

A primeira camada 1-D conecta-se a saída da última camada 2-D. Cada neurônio 2-D é conectado a um campo receptivo na camada anterior e é cercado por um campo inibitório da mesma camada. Estes neurônios também podem compartilhar informações dentro de seu campo receptivo [2]. A Figura 03 retrata a arquitetura da LIPNet, bem como as conexões que ocorrem na inibição lateral.

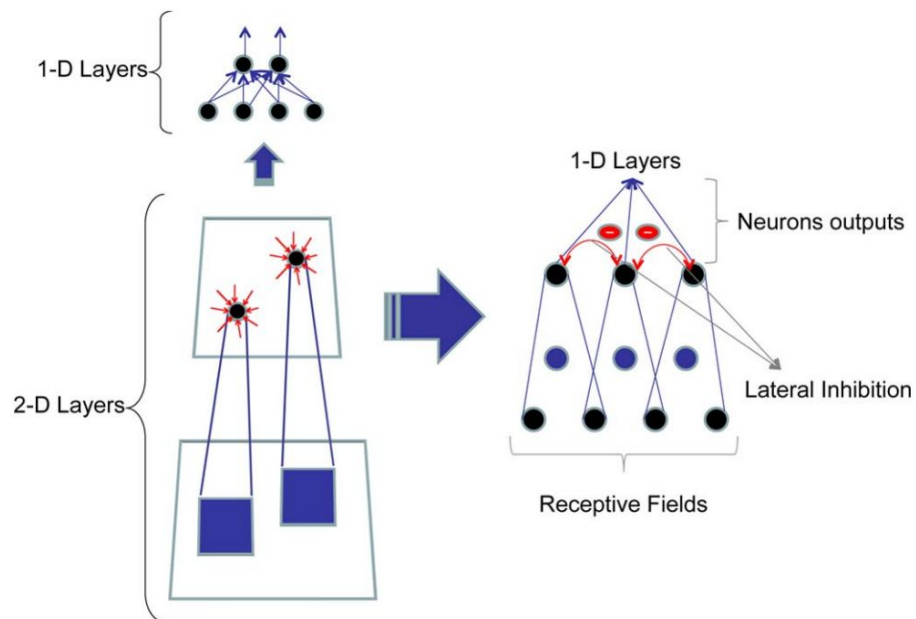


Figura 03. Arquitetura da LIPNet. A entrada da primeira camada 2-D é a imagem a ser classificada. Cada neurônio 2-D é conectado a um campo receptivo e é

cercado pelo campo inibitório da mesma camada. A primeira camada 1-D é a saída de um vetor rearranjado da última camada 2-D. Extraído de [2].

A LIPNet pode resolver os mesmos problemas que a PyraNet, isto porque a PyraNet é uma sub-categoria da LIPNet [2], isto significa que, se parametrizado a fim de eliminar a inibição lateral, a LIPNet comporta-se de maneira idêntica a PyraNet. Por isto a LIPNet, que já foi aplicada para resolver problemas como detecção de faces [2], e classificação em componentes sanguíneos [15], também deve ser capaz de solucionar os mesmos problemas que a PyraNet é capaz de solucionar. A Figura 04 ilustra o exemplo de entrada usado na LIPNet para identificação de gênero.



Figura 04. Base de dados Morph-II utilizada por Soares para treinar modelos de detecção de gênero em faces. Extraído de [15].

Outra das vantagens desta arquitetura está na quantidade de parâmetros necessárias para armazená-la em memória. Para todas as aplicações apresentadas por Soares [15], a LIPNet requereu de 5,85% a 27,48% do número de variáveis que a DCNN necessitaria para atingir níveis similares de classificação.

2.2 Transformada de Fourier para espectrogramas

Neste projeto todas as análises dos sinais foram feitas usando o domínio da frequência em função do tempo. O uso deste domínio é importante porque permite a visualização de componentes específicos de frequência relacionados ao aumento das falhas [8].

Segundo Monteiro [8], a *Fourier Transform* (FT, ou Transformada de Fourier) é uma técnica de análise de sinais que informa a composição de frequências de um sinal e a contribuição de cada uma delas por sua magnitude (apud GAO, 2015 [3]), contribuindo para melhorias na filtragem de ruídos, reconhecimento de padrões e modulação de sinais. Também existem variantes, como a *Discrete Fourier Transform* (DFT, ou transformada de Fourier no domínio discreto) que agrega os conceitos da FT, aplicado para problemas no domínio discreto e a *Fast Fourier Transform* (FFT, ou transformada rápida de fourier), algoritmo que calcula a DFT de maneira mais eficiente (apud RAO, 2011 [12]).

A grande contribuição acerca da FFT é a redução da complexidade computacional de N^2 para $N \log N$ [4], reduzindo significativamente o tempo de execução deste algoritmo, provando-se útil na redução de custo computacional, que é o principal problema atacado por este projeto.

Ainda segundo Monteiro [8] a variante *Short Time Fourier Transform* (STFT) é comumente utilizada para análises de espectrogramas, pois gera uma representação da frequência e magnitude em função tempo (apud RAO [12]).

Uma vez que este projeto visa um menor custo computacional, as imagens que foram alimentadas à rede são geradas a partir do uso da STFT nas leituras realizadas a partir de um acelerômetro. Isto porque espera-se que esta solução seja utilizada em um sistema com baixo poder computacional.

Capítulo 3

Metodologia

Este capítulo tem por objetivo explicar todo o processo de aquisição da base de dados utilizada no projeto e como os conceitos apresentados previamente são aplicados.

3.1 Aquisição de dados

Neste experimento foi utilizado como gerador de sinais uma engrenagem em uma caixa de marcha de um automóvel com falhas introduzidas propositalmente em sua estrutura. A base obtida com a captura do funcionamento deste sistema contém diversos sinais de diversos tipos de problemas [8]. Espera-se que o modelo de rede piramidal treinando com esta base seja capaz de identificar qual falha está ocorrendo neste sistema.

De acordo com Monteiro [8], caixas de marcha podem apresentar diversas falhas, podendo estar relacionadas a condições de componentes mecânicos ou lubrificação. Uma das falhas que necessita de cuidados é a quebra dos dentes das engrenagens, que comprometem a operação do maquinário de forma significativa (apud NIE, 2013 [9]).

O arranjo do maquinário pode ser o visualizado no esquema da Figura 05. A caixa de marcha é composta por duas engrenagens (Z1 e Z2). Cada engrenagem possui um eixo independente. Também há um freio magnético (B) que está conectado ao eixo de saída e uma unidade de velocidade conectada ao motor [8].

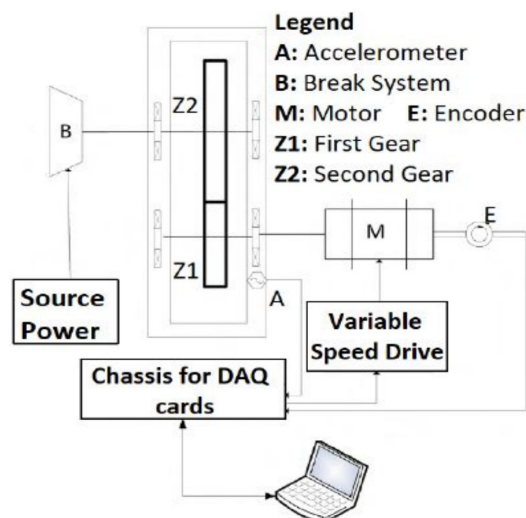


Figura 05. Esquema da extração de sinais na caixa de marcha. Extraído de [8].

Sobre esta caixa de marcha foi adicionado um acelerômetro unidirecional que tem a capacidade de extrair amostragens periódicas com frequência de 50 kHz. Além disso, foram feitas diferentes combinações de velocidades de rotação com diferentes tensões aplicadas sobre o freio magnético e diferentes níveis de desgaste dos dentes na engrenagem, onde cada amostra foi capturada num intervalo de 10 segundos, normalizada em um intervalo entre 0 e 1 e dividida em trechos de 0,25 segundos [8]. Ao todo, a base consiste de 18.000 sinais, onde existem 1.800 para cada nível de severidade no desgaste da engrenagem. Os níveis de severidade podem ser vistos na Figura 06.

Todas capturas foram feitas a partir da engrenagem Z1, e nela foram simulados dez níveis de desgaste. No primeiro teste não há desgaste, entretanto, nos outros 9 o grau de desgaste aumentou. A engrenagem Z2 não foi modificada [8].

Uma vez que as capturas da classe P1 são obtidas com a partir da engrenagem íntegra, todos os comportamento que não forem classificados como pertencentes a esta classe são considerados anômalos, as demais classes servem para indicar o grau de severidade da anomalia.



Figura 06. Aqui são apresentados os diferentes níveis de severidade na quebra dos dentes da engrenagem Z1. Extraído de [8].

Em seguida, para alimentar a rede, converteu-se os sinais unidimensionais extraídos para o domínio bidimensional. Para tal fez-se uso da STFT com janela de Hamming de 128 e sobreposição de 50%. Esta técnica foi necessária pois permite a visualização de componentes da frequência específicos ao aumento de falhas [8]. Após isso as imagens foram re-normalizadas para um intervalo de 0 a 255 e então foram geradas outras imagens em escalas de cinza para alimentar a LIPNet.

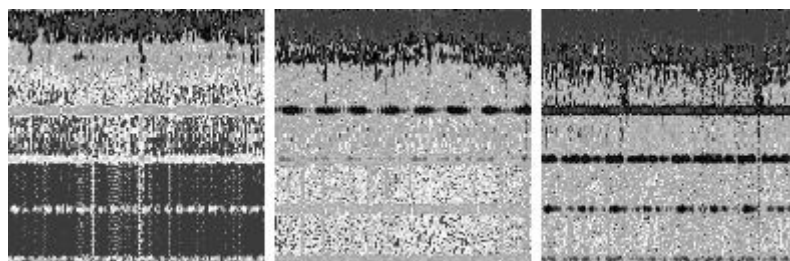


Figura 07. Exemplo de sinais de entrada da LIPNet. Na esquerda está a classe P1, ao centro a classe P5 ao lado direito a classe P10.

Uma vez que a LIPNet pode ser configurada durante o treinamento para apenas avaliar uma sub-região da imagem, foram testadas algumas configurações avaliando sub-regiões da imagem original (128x128) e alguns treinamentos avaliando a imagem completa e redimensionada.

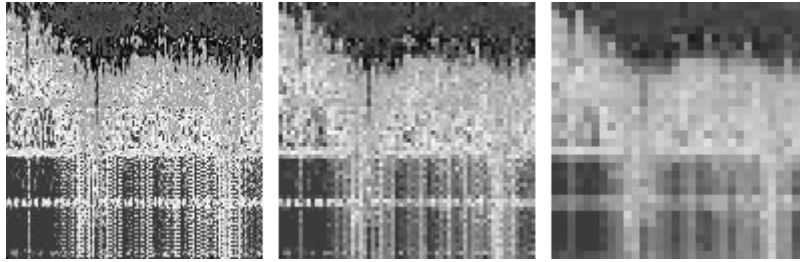


Figura 08. Processo de redimensionamento de uma imagem da classe P1. Da esquerda para a direita estão apresentadas as versões 128x128 (original), 64x64 e 32x32.

O redimensionamento foi feito dividindo a imagem original pela metade de seu tamanho, e os pixels que desapareceriam foram somados e divididos por 4, gerando assim duas bases (64x64 e 32x32).

Esta maneira de redimensionar foi aplicada para evitar a perda de informação nos pontos que seriam excluídos caso apenas uma sub-região da imagem for avaliada. Além disso, uma diminuição do tamanho da imagem reduz significativamente tempo do processo de treinamento e classificação.

3.2 Treinamento da rede LIPNet

Com os dados prontos e normalizados se iniciou o processo de treinamento da LIPNet. No entanto, esta rede age como um classificador binário e, como são dez classes, algumas modificações no agrupamento das imagens foram feitas.

Primeiramente, foram separados de cada classe 75% dos dados para treinamento e 25% para validação. Como exemplos positivos apenas estavam presentes as imagens da classe em questão, já para os exemplos negativos, todas as outras nove. Após separar todas as imagens para dez classes, foi iniciada uma execução exploratória (configuração 1), ou seja, a quantidade de épocas foi extrapolada para averiguar sua quantidade ideal.

Na execução exploratória percebeu-se que apenas 100 épocas seriam suficientes para atingir acurácias superiores a 70%, então as demais parametrizações tiveram seu número máximo de épocas reduzido para 100.

Sobre os demais parâmetros, fez-se necessário explorá-los para averiguar a sua contribuição para a acurácia total. Para tal, foram executadas simulações e, dentre os parâmetros do modelo, a quantidade de neurônios por camada foi a mais influente, tornando este o foco da análise paramétrica.

A fim de reduzir a complexidade da análise, foi aplicada uma regra de construção na rede que define quantos neurônios serão utilizados por camada em função da sub-região da imagem analisada, da sobreposição e do tamanho do campo receptivo. Para isto, todos os experimentos consideraram os campos receptivos como quadrados. Esta regra também foi utilizada na variante LIPNet-PSO por Soares [14]. Esta regra é representada pela fórmula a seguir:

$$H^l = \frac{H^{l-1} - o^l}{r^l - o^l}, \quad (1)$$

onde H é a quantidade de neurônios na camada (altura e largura), que caso seja a primeira, assume o valor da sub-região da imagem. A variável o é a região de sobreposição e r é o campo receptivo.

Graças a esta simplificação, foi eliminada a responsabilidade de configurar um vetor de parâmetros do modelo. Os parâmetros de campo receptivo, sobreposição, campo inibitório e peso inibitório não possuem recomendações quanto os seus intervalos e valores, por isto se mantiveram constantes durante todas as simulações.

Iniciado o processo de testes usando a base original (128x128) onde o ponto a ser explorado seria o tamanho ideal de pixels a ser usados de entrada no modelo a fim de se obter uma acurácia ainda maior que a configuração 1, é gerada a configuração 2.

A configuração 2 contempla o primeiro quarto da imagem original, isto porque caso se desejasse analisar a imagem por completo, o tempo de treinamento seria exponencialmente maior. Além disso, assim como apontado pelos resultados obtidos na configuração 1, há de ser considerado que apenas uma região da imagem pode conter toda a informação necessária para a classificação adequada.

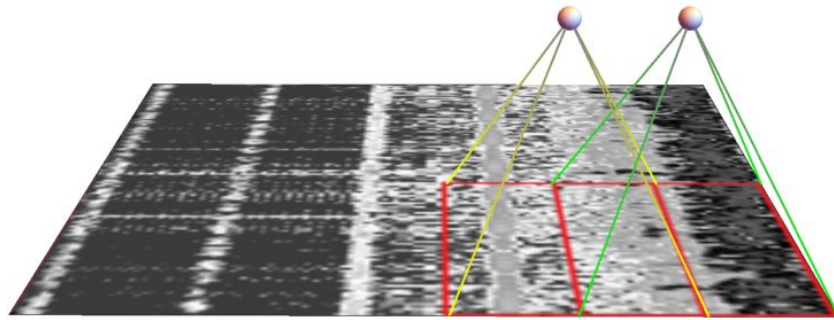


Figura 09. Representação visual da Configuração 2. As áreas vermelhas nas bases das pirâmides representam o campo receptivo da 2ª camada 2-D. Apenas um quarto da imagem é utilizada como entrada para a rede. A sobreposição dos campos receptivos é observada a partir da intersecção dos triângulos.

Devido ao problema do tempo de treinamento, foram gerados duas bases, cujo processo de criação é descrito no Capítulo 3. A aplicação dessas bases promoveu uma simplificação no processo de treinamento, surgindo as configurações 3 e 4. Na configuração 3, os parâmetros são idênticos em relação à configuração 2, com exceção da base utilizada como entrada, que contém mais informações condensadas em menos pontos. A configuração 4 usa uma base ainda menor.

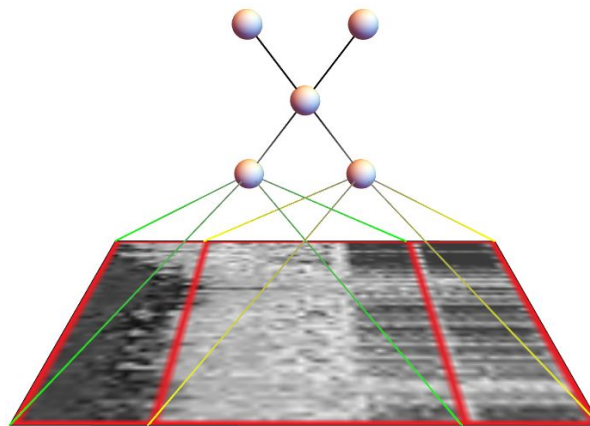


Figura 10. Representação da rede nas configurações 3 e 4. A imagem inteira é compreendida dentro do campo receptivo. Os vértices denotam as matrizes de neurônios por camadas. Os vértices superiores são os neurônios de saída da rede.

O algoritmo da LIPNet foi desenvolvido em Java. Esta implementação é a mesma usada por Fernandes *et al.* em [2]. No entanto, algumas alterações foram feitas quanto à captura de dados durante o processo de treinamento. Estas, por sua

vez, não interferem no seu comportamento. Os valores referentes às configurações adotadas durante o processo de análise paramétrica estão descritos na Tabela 1.

Todas as simulações foram feitas em um computador pessoal, sem uso de cálculos em GPU. A máquina em questão possui as seguintes configurações:

- Processador: Intel(R) Core™ i5-7300HQ, 4 núcleos 2.5 GHz
- Memória: 16GB DDR4 (2400 MHz)
- Sistema Operacional: Windows 10 Pro 64 bits (17763)

Base	128x128 (Original)		64x64	32x32
Parâmetro (Nome)	Configuração 1	Configuração 2	Configuração 3	Configuração 4
Sub-região da imagem analisada (x, y)	(0,0) a (19,19)	(0,0) a (63,63)	Imagem Completa	Imagem Completa
Qtd. de camadas 2D	2	2	2	2
Matriz de neurônios por camada (definidas pela eq. 1)	[6x6, 2x2]	[20x20, 6x6]	[20x20, 6x6]	[10x10, 3x3]
Campo Receptivo por camada	[4, 3]	[4, 3]	[4, 3]	[4, 3]
Sobreposição por camada	[1, 0]	[1, 0]	[1, 0]	[1, 0]
Campo inibitório por camada	[2, 0]	[2, 0]	[2, 0]	[2, 0]
Peso inibitório por camada	[1,6, 0]	[1,6, 0]	[1,6, 0]	[1,6, 0]
Qtd. de épocas por treino	500	100	100	100
Qtd. de simulações	30	30	30	30

Tabela 1. Tabela de configuração dos parâmetros da LIPNet.

Capítulo 4

Resultados

Durante a análise exploratória (configuração 1) todas as simulações apresentaram resultados nos quais a menor das acurácias foi 64,70%. Os algoritmos atingiram os máximos locais em poucas iterações. Após algumas épocas as curvas de sua acurácia pioraram ou demonstraram aumentos pouco expressivos, como na classe P10.

Os desvios padrões da configuração 1 também foram muito altos. A isto se atribui a falta de informação contida no primeiro bloco de pixels da imagem (20x20), não sendo suficiente para uma classificação adequada. O tempo de execução e classificação foram os menores encontrados durante as análises, pois a rede possui a menor quantidade de neurônios dentre as testadas.

Na configuração 2, todas as melhores acurácias foram encontradas, em média, nas últimas épocas, mesmo que a melhora por época não fosse expressiva. Agora, com o aumento nas informações de entrada, os desvios foram reduzidos. No geral não houve melhorias quanto a classificação, isto porque, assim como na configuração 1, a sub-área selecionada da imagem não possui informação o suficiente para contribuir com inferências corretas em todas as classes.

Nas configurações 3 e 4 o modelo obteve melhores acurácias, pois havia uma maior qualidade na informação da entrada na rede. Além disso, o tempo de treinamento, custo de memória e classificação são reduzidos, pois as imagens carregadas ocupam menos espaço.

A formação da rede na configuração 3 consiste de mais neurônios que a configuração 4, podendo ser muito mais influenciada por alterações nos parâmetros de campos receptivo e inibitório, sobreposição e peso inibitório. Devido a quantidade de neurônios ser a mesma da configuração 2, os tempos de treinamento se

mantiveram próximos, com exceção do período de pré-processamento nos dados. Dentre as configurações testadas, esta foi a que apresentou os melhores resultados.

Por fim, a configuração 4 foi a mais rápida em tempo de treinos e classificação, mas ainda se apresentando inferior à configuração 1. Entre as configurações é a de segundo menor custo computacional e a sua acurácia é similar a apresentada pela configuração 3.

Dentre as classes treinadas, a P1, P2 e P10 foram as mais fáceis de serem identificadas pelos classificadores, e as classes P4, P5, P7 e P9 as mais difíceis.

A Tabela 2 apresenta os resultados de acurácias obtidas nos testes. Os valores entre parêntesis indicam a média de épocas onde houve melhor acurácia de cada simulação, em seguida a acurácia encontrada e por fim o desvio padrão.

	Sub-Região		Imagem Redimensionada	
	128x128		64x64	32x32
	(Média de iterações) Acurácia (%)			
Classe	Configuração 1	Configuração 2	Configuração 3	Configuração 4
P1	(43)74,51±09,8	(100)89,57±01,8	(100)93,07±01,8	(100)99,92±00,1
P2	(42)86,95±05,8	(100)98,49±00,4	(100)85,63±06,6	(100)92,26±02,9
P3	(27)80,00±10,1	(100)71,50±08,7	(99)85,62±04,6	(100)82,27±05,2
P4	(25)68,68±21,0	(100)58,76±16,1	(100)71,20±09,7	(100)80,62±04,2
P5	(33)71,92±09,9	(100)53,15±13,0	(100)80,28±05,6	(100)75,96±10,3
P6	(51)69,58±07,7	(100)64,62±16,1	(100)88,82±01,1	(100)76,43±06,7
P7	(3)67,94±44,5	(100)73,97±02,7	(100)80,33±09,7	(99)77,42±09,6
P8	(2)69,34±46,2	(100)55,83±06,4	(98)99,90±00,1	(100)78,59±06,5
P9	(36)65,83±15,6	(100)66,77±07,7	(100)72,19±14,0	(100)86,75±01,3
P10	(395)64,70±11,8	(100)74,80±01,9	(97)92,37±00,8	(100)91,47±01,5
Média	71,94±06,9	70,75±01,5	84,94±00,9	84,17±08,1

Tabela 2. Acurácias e desvios padrões obtidos para diferentes configurações da LIPNet.

Outro fato observado foi quanto à redução da imagem. Para o problema testado e dentro das configurações aqui propostas, a queda da qualidade da informação de entrada devido a redução de suas dimensões não foi um dos fatores cruciais para queda do poder de classificação da rede.

Na Tabela 3 é apresentado o tempo de execução dos algoritmos bem como o tempo que a rede leva para classificar uma imagem.

	Configuração 1	Configuração 2	Configuração 3	Configuração 4
Classificação (milisegundos)	0,03	0,5	0,5	0,04
Tempo médio de treinamento por classe (segundos)	5224,92	16337,53	15127,26	3294,35

Tabela 3. Tempos de treinamento e classificação para cada configuração testada.

Estes tempos presentes na Tabela 3 podem estar sujeitos a variações a depender da utilização do CPU. Nos testes executados a máquina não estava executando mais processos do que os necessários ao sistema operacional, ou seja, pouco mais de 90% da CPU foi dedicada exclusivamente para o treinamento da rede. Quando comparado os resultados obtidos da LIPNet e a DCNN usada por Monteiro [8] fica claro que há diferença entre as classificações apontadas, sendo a DCNN superior dentre as configurações testadas pela LIPNet.

Para a LIPNet superar a DCNN neste problema seria necessário ajustes finos para melhorar a sua acurácia e, além disso, também há a possibilidade de explorar alternativas de atuação desde o pré-processamento dos dados até após as classificações. Monteiro [8], por exemplo, anexa à saída da DCNN um outro classificador (SVM) visando aprimorar seu poder de decisão. Esta mesma estratégia poderia ser aplicada aqui.

Destaca-se, no entanto, que houveram dois experimentos conduzidos por Monteiro [8] com a DCNN, em um deles foi usado imagens em escala de cinza e o outro imagens coloridas. Para as imagens em escala de cinza, os experimentos apontam que a acurácia média obtida é próxima a encontrada pela LIPNet, já no experimento conduzido com imagens coloridas, a DCNN foi capaz de superar a LIPNet, isto porque o algoritmo usado da LIPNet não foi projetado para lidar com imagens coloridas.

Avaliar imagens coloridas auxilia o processo de classificação. A depender da forma em que foi codificada, uma imagem pode ser composta de três a quatro canais capazes de armazenar informação, enquanto imagens em escalas de cinza contém apenas um. Por isto, não há maneira justa de comparar as diferentes arquiteturas quando ambas foram alimentadas com informações distintas.

A Tabela 4 é o comparativos entre as acurácias encontradas pela DCNN e LIPNet. A coluna DCNN + SVM apresenta as acurácias obtidas por Monteiro [8] quando utilizado uma SVM como camada final da rede.

	LIPNet Configuração 3	LIPNet Configuração 4	DCNN	DCNN + SVM
Base	Imagens em escalas de cinza			
Classe	Acurácia Média (%)			
P1	93,07±01,8	99,92±00,1	99,98±00,1	99,90±00,1
P2	85,63±06,6	92,26±02,9	98,35±01,3	95,00±00,6
P3	85,62±04,6	82,27±05,2	93,04±07,0	94,73±01,8
P4	71,20±09,7	80,62±04,2	81,58±12,4	86,64±03,8
P5	80,28±05,6	75,96±10,3	75,50±15,1	81,26±03,5
P6	88,82±01,1	76,43±06,7	69,13±19,5	79,60±04,1
P7	80,33±09,7	77,42±09,6	74,30±15,0	81,61±02,5
P8	99,90±00,1	78,59±06,5	82,70±07,2	86,16±01,8
P9	72,19±14,0	86,75±01,3	85,47±08,5	88,46±01,4
P10	92,37±00,8	91,47±01,5	89,81±09,6	94,70±02,9
Média	84,94±00,9	84,17±08.1	84.98±10.4	88.80±7.0

Tabela 4. Resultados das simulações da LIPNet em comparação com a DCNN ambas treinadas com imagens em escalas de cinza.

Na Tabela 4 é possível observar que, dentro das configurações testadas, a configuração 3 se sobressaiu dentre as demais para as classes P6 e P8. Também é constatado que a média das simulações foi próxima da DCNN pura e ainda que os desvios padrões para cada classe foi menor na LIPNet. A DCNN munida do classificador extra (SVM) superou a LIPNet nas demais classes.

Capítulo 5

Considerações Finais

Este capítulo tem por objetivo concluir o desenvolvimento da monografia e sugerir possíveis trabalhos futuros acerca do tema.

5.1 Conclusão

Assim como definido na introdução, este trabalho foi idealizado com o intuito de trazer maneiras mais eficientes de tratar com problema de classificação em análise de sinais. A acurácia da LIPNet atendeu bem as expectativas e teve um bom percentual de acerto em se tratando das classificações de anomalias, porém, o processo ainda há de ser otimizado para superar a DCNN. A desvantagem do uso da LIPNet é a quantidade de parâmetros utilizados pela rede. Ao executar testes exaustivos nesta arquitetura alterando suas configurações, pode-se melhorar as acurácias encontradas.

Dentre os testes realizados pode-se perceber que há potencial para melhora dos resultados ao ajustar os parâmetros adequadamente. A depender da classe testada, a rede foi capaz de classificar muito bem, atingindo acurácias que apresentaram variações entre 71,2% e 99,9%.

Assim como proposto originalmente, o ganho no tempo de classificação é notório. Ainda podendo ser mais otimizado se aplicadas simplificações na dimensionalidade das imagens de entrada da rede.

5.2 Trabalhos Futuros

Durante o desenvolvimento deste projeto foram consideradas diversas alternativas que poderiam melhorar o desempenho do processo e dos resultados,

abrindo várias possibilidades de novas pesquisas. Para trabalhos futuros apontam-se as seguintes sugestões:

- Considerar algoritmos que evidenciam os pontos de maior entropia da imagem para serem utilizadas como campo receptivo. Ideia similar pode ser encontrada na SPNN, rede inspirada na LIPNet [14]. Desta forma não serão necessários os cálculos de pontos irrelevantes da imagem, melhorando o tempo de execução.
- Realizar a busca dos parâmetros da LIPNet com hiper-heurísticas, fazendo com que os máximos encontrados melhorem a adequação da rede para o problema. Aproximação similar foi realizada por Soares [15] com o algoritmo de enxames PSO e provou-se capaz de aumentar a acurácia do modelo.
- Redefinir a estrutura do algoritmo da LIPNet para uma classificação n-ária, reduzindo assim a quantidade de treinos necessário para todas as classes.
- Re-projetar a forma que os pontos na imagem são avaliadas a fim de extrair características de múltiplos canais, permitindo que a rede seja treinada com imagens coloridas.
- Utilizar os coeficientes obtidos nas saídas de cada classificação para cada classe como entrada para uma outra rede, atuando como camada de decisão. Trabalho similar a este foi realizado por Monteiro *et al.* [8].

Bibliografia

- [1] CHANDOLA, Varun; BANERJEE, Arindam; KUMAR, Vipin. Anomaly detection: A survey. **ACM computing surveys (CSUR)**, v. 41, n. 3, p. 15, 2009.
- [2] FERNANDES, Bruno José Torres; CAVALCANTI, George DC; REN, Tsang Ing. Lateral inhibition pyramidal neural network for image classification. **IEEE transactions on cybernetics**, v. 43, n. 6, p. 2082-2092, 2013.
- [3] GAO, Zhiwei; CECATI, Carlo; DING, Steven X. A survey of fault diagnosis and fault-tolerant techniques—Part I: Fault diagnosis with model-based and signal-based approaches. **IEEE Transactions on Industrial Electronics**, v. 62, n. 6, p. 3757-3767, 2015.
- [4] GONZALEZ, Rafael C.; WOODS, Richard E. **Digital Image Processing**. Terceira Edição. 2008.
- [5] GRIGORESCU, Cosmin; PETKOV, Nicolai; WESTENBERG, Michel A. Contour detection based on nonclassical receptive field inhibition. **IEEE Transactions on image processing**, v. 12, n. 7, p. 729-739, 2003.
- [6] LATHUILIÈRE, Stéphane et al. A comprehensive analysis of deep regression. **IEEE transactions on pattern analysis and machine intelligence**, 2019.
- [7] LECUN, Yann; BENGIO, Yoshua; HINTON, Geoffrey. Deep learning. **nature**, v. 521, n. 7553, p. 436, 2015.
- [8] MONTEIRO, Rodrigo P. *et al.* Using a Support Vector Machine Based Decision Stage to Improve the Fault Diagnosis on Gearboxes. **Computational Intelligence and Neuroscience**, v. 2019, 2019.
- [9] NIE, Mengyan; WANG, Ling. Review of condition monitoring and fault diagnosis technologies for wind turbine gearbox. **Procedia Cirp**, v. 11, p. 287-290, 2013.
- [10] OPPENHEIM, Alan V.; WILLSKY, Alan S.; NAWAB, S. Signals and Systems (Prentice-Hall signal processing series). 1996.

- [11] PHUNG, Son Lam; BOUZERDOUM, Abdesselam. A pyramidal neural network for visual pattern recognition. **IEEE transactions on neural networks**, v. 18, n. 2, p. 329-343, 2007.
- [12] RAO, Kamisetty Ramamohan; KIM, Do Nyeon; HWANG, Jae Jeong. **Fast Fourier transform-algorithms and applications**. Springer Science & Business Media, 2011.
- [13] ROSENBLATT, Frank. The perceptron: a probabilistic model for information storage and organization in the brain. **Psychological review**, v. 65, n. 6, p. 386, 1958.
- [14] SOARES, Alessandra M.; FERNANDES, Bruno JT; BASTOS-FILHO, Carmelo JA. Lateral inhibition pyramidal neural networks designed by particle swarm optimization. In: **International Conference on Artificial Neural Networks**. Springer, Cham, 2014. p. 667-674.
- [15] SOARES, Alessandra M.; FERNANDES, Bruno JT; BASTOS-FILHO, Carmelo JA. Structured pyramidal neural networks. 2018.
- [16] VALENÇA, Mêuser JS. Fundamentos das redes neurais: exemplos em Java. **Recife: Livro Rápido**, 2007.
- [17] XIE, Junyuan; GIRSHICK, Ross; FARHADI, Ali. Unsupervised deep embedding for clustering analysis. In: **International conference on machine learning**. 2016. p. 478-487.
- [18] ZHAO, Minghang et al. Deep residual networks with dynamically weighted wavelet coefficients for fault diagnosis of planetary gearboxes. **IEEE Transactions on Industrial Electronics**, v. 65, n. 5, p. 4290-4300, 2017.