

UNIVERSIDADE DE SÃO PAULO
ESCOLA DE ENGENHARIA DE SÃO CARLOS

JOÃO PEDRO FLORENTINO DE OLIVEIRA NASCIMENTO

Classificação automática de micrografias por métodos de aprendizado de máquina

São Carlos

2019

JOÃO PEDRO FLORENTINO DE OLIVEIRA NASCIMENTO

Classificação automática de micrografias por métodos de aprendizado de máquina

Monografia apresentada ao Curso de Engenharia de Materiais e Manufatura, da Escola de Engenharia de São Carlos da Universidade de São Paulo, como parte dos requisitos para obtenção do título de Engenheiro de Materiais.

Orientador: Profa. Dra. Lauralice de Campos Franceschini Canale

São Carlos

2019

AUTORIZO A REPRODUÇÃO TOTAL OU PARCIAL DESTA TRABALHO,
POR QUALQUER MEIO CONVENCIONAL OU ELETRÔNICO, PARA FINS
DE ESTUDO E PESQUISA, DESDE QUE CITADA A FONTE.

Ficha catalográfica elaborada pela Biblioteca Prof. Dr. Sérgio Rodrigues Fontes da
EESC/USP com os dados inseridos pelo(a) autor(a).

N244c Nascimento, João Pedro Florentino de Oliveira
Classificação automática de micrografias por
métodos de aprendizado de máquina / João Pedro
Florentino de Oliveira Nascimento; orientadora
Lauralice de Campos Franceschini Canale. São Carlos,
2019.

Monografia (Graduação em Engenharia de Materiais
e Manufatura) -- Escola de Engenharia de São Carlos da
Universidade de São Paulo, 2019.

1. Classificação de imagens. 2. Microestruturas.
3. Máquina de suporte vetorial. 4. Redes neurais. I.
Título.

Eduardo Graziosi Silva - CRB - 8/8907

FOLHA DE APROVAÇÃO

Candidato: João Pedro Florentino de Oliveira Nascimento
Título do TCC: Classificação automática de micrografias por métodos de aprendizado de máquina
Data de defesa: 14/11/2019

Comissão Julgadora	Resultado
Professor Titular Luralice de Campos Franceschini Canale (orientador)	<i>APROVADO</i>
Instituição: EESC - SMM	
Professor Doutor Roberto Ramon Mendonça	<i>APROVADO</i>
Instituição: IFSP - SC	
Pesquisador Rosa Simêncio Otero	<i>Aprovado</i>
Instituição: IQSC - Departamento de Química e Física Molecular	

Presidente da Banca: **Professor Titular Luralice de Campos Franceschini Canale**



(assinatura)

EPÍGRAFE

*“If we want machines to think, we need to
teach them to see.”*

Fei-Fei-Li (2015)

RESUMO

NASCIMENTO, J.P.F.O. **Classificação automática de micrografias por métodos de aprendizado de máquina.** 2019. 85p. Monografia (Trabalho de Conclusão de Curso) – Escola de Engenharia de São Carlos, Universidade de São Paulo, São Carlos, 2019.

A classificação de microestruturas ainda é feita, principalmente, de forma manual por especialistas com o auxílio de quadros metalográficos. Esse tipo de abordagem dá margens a incertezas devido ao alto grau de subjetividade envolvido. O desenvolvimento de métodos automáticos de classificação é um grande desafio pois as microestruturas são, em muitos casos, uma combinação de diferentes subestruturas complexas. Este trabalho pretende apresentar os principais métodos de classificação de imagens, bem como aplicá-los em três estudos de caso. A aplicação da máquina de suporte vetorial (SVM) mostrou-se de fácil implementação e interpretação, mas depende de métodos externos para a extração de características. Os modelos de redes neurais convolucionais (CNN) apresentaram boa capacidade de classificação apesar da queda de desempenho com a adição de novas classes. A aplicação do modelo *Mask R-CNN* mostrou-se eficaz mesmo na inferência de imagens com qualidade e ampliação diferentes das utilizadas em treinamento. Os resultados dão fortes indícios de que a qualidade de um classificador está atrelada não somente a quantidade de amostras de treinamento, mas também à qualidade e representatividade dessas amostras.

Palavras-chave: Classificação de imagens, Microestruturas, Máquina de suporte vetorial, Redes neurais.

ABSTRACT

NASCIMENTO, J.P.F.O. **Automatic classification of micrographs by machine learning methods.** 2019. 85p. Monografia (Trabalho de Conclusão de Curso) – Escola de Engenharia de São Carlos, Universidade de São Paulo, São Carlos, 2019.

The classification of microstructures is still mainly done manually by specialists with the help of metallographic charts. This type of approach gives rise to uncertainty due to the high degree of subjectivity involved. The development of automatic classification methods is a major challenge as microstructures are in many cases a combination of different complex substructures. This paper presents the main methods of image classification and applies them in three case studies. The application of support vector machine (SVM) proved to be easy to implement and interpret but depends on external methods for feature extraction. The convolutional neural network models (CNN) presented good classification capacity, despite the performance decrease with the addition of new classes. The application of the Mask R-CNN model proved to be effective even in the inference of images with different quality and magnification from those used in training. The results give strong indications that the quality of a classifier is linked not only to the quantity of training samples, but also to the quality and representativeness of these samples.

Keywords: Image classification, Microstructures, Support vector machine, Neural networks.

LISTA DE ILUSTRAÇÕES

Figura 1 - Exemplo da operação de convolução	25
Figura 2 – Micrografia de aço contendo perlita-esferoidita	26
Figura 3 - Histogramas de Intensidade da micrografia apresentada na Figura 2	26
Figura 4 - Imagem original x representação SIFT	27
Figura 5 - Imagem original x representação HOG	28
Figura 6– Micrografia rotulada por seu microconstituente predominante: “esferoidita”	29
Figura 7 - SVM de margens rígidas	32
Figura 8 - SVM de margens suaves	34
Figura 9 - SVM de margens suaves	35
Figura 10 - Arquitetura de um <i>Perceptron</i>	36
Figura 11 - Perceptron de camadas múltiplas	37
Figura 12 - Função sigmoide.....	38
Figura 13 - Função rectified linear unit.....	38
Figura 14 - Função tangente hiperbólica.....	39
Figura 15 - Representação do método do gradiente	40
Figura 16 - Operação de max pooling.....	42
Figura 17 - Classes de problemas em visão computacional	43
Figura 18 - UltraHigh Carbon Steel Micrograph Database.....	46
Figura 19 - Fotomicrografia 306 - Network.....	47
Figura 20 - Distribuição de micrografias por classe.....	48
Figura 21 - Distribuição de micrografias por ampliação.....	48
Figura 22 - Fotomicrografia sem o rodapé.....	49
Figura 23 - Ferro fundido nodular.....	50
Figura 24 - Ferro fundido nodular.....	50
Figura 25 - Regiões anotadas	51
Figura 26- Esferoidita e representação HOG	55
Figura 27 - Perlita e representação HOG	55
Figura 28 – Classe esferoidita e representação HOG	56
Figura 29 - Classe perlita e representação HOG	57
Figura 30 - Superfície de decisão SVM	59
Figura 31 - Superfície de decisão	60
Figura 32 - Gráficos de acurácia e custo, modelo 03_VGG_16	61
Figura 33 - Gráficos de acurácia e custo, modelo 04_VGG_16	61
Figura 34 - Gráficos de acurácia e custo, modelo 05_VGG_16	62
Figura 35- Magnificação por classe, esferoidita	65
Figura 36 - Magnificação por classe, martensita.....	65
Figura 37 - Magnificação por classe, network	66
Figura 38 - Magnificação por classe, perlita	66
Figura 39 - Magnificação por classe, perlita+esferoidita	67
Figura 40 - Fotomicrografias esferoidita.....	67
Figura 41- Fotomicrografias martensita	68
Figura 42 - Fotomicrografias network	68

Figura 43 - Fotomicrografias perlita	68
Figura 44 - Fotomicrografias perlita + esferoidita	69
Figura 45 - Fotomicrografias martensita.....	69
Figura 46 - Fotomicrografias martensita.....	70
Figura 47 - Fotomicrografia martensita.....	70
Figura 48 - Fotomicrografias martensita.....	70
Figura 49 - Fotomicrografias martensita	71
Figura 50 - Amostra de imagens do banco de treinamentos, martensita.....	72
Figura 51 - Fotomicrografias perlita, perlita + esferoidita	73
Figura 52 - Amostra de dados de treinamento. A: perlita, B: perlita + esferoidita	73
Figura 53 - Inferência Mask R-CNN.....	74
Figura 54 - Inferência Mask R-CNN.....	74
Figura 55 - Inferência Mask R-CNN.....	75
Figura 56 - Inferência Mask R-CNN.....	75
Figura 57 - Teste do modelo	76
Figura 58 - Teste do modelo	77
Figura 59 - Teste do modelo	77

LISTA DE TABELAS

Tabela 1 - Função Kernel e seus respectivos parâmetros.....	35
Tabela 2 - Organização do banco de dados	49
Tabela 3 - Modelos CNN	53
Tabela 4 - Parâmetros dos modelos.....	53
Tabela 5 - Relatório de classificação modelo 01, imagens seccionadas	56
Tabela 6- Relatório de classificação modelo 01, banco de teste	57
Tabela 7 - Relatório de classificação modelo 02, imagens seccionadas	58
Tabela 8 - Relatório de classificação do modelo 02, banco de teste	59
Tabela 9 - Relatório de classificação 03_VGG_16.....	63
Tabela 10 - Relatório de classificação 04_VGG_16.....	63
Tabela 11 - Relatório de classificação 05_VGG_16.....	64

LISTA DE ABREVIATURAS E SIGLAS

CNN	–	Redes neurais convolucionais
HOG	–	Histograma dos gradientes orientados
MLP	–	<i>Perceptron</i> de camadas múltiplas
PCA	–	Análise de componentes principais
RCNN	–	<i>Region-based convolutional neural network</i>
RELU	–	<i>Rectified linear unit</i>
RESNET	–	<i>Residual neural network</i>
RPN	–	<i>Regional proposal network</i>
SIFT	–	Transformada de características invariantes
SVM	–	Máquina de suporte vetorial
TANH	–	Tangente hiperbólica
VGG	–	<i>Visual Geometry Group</i>

SUMÁRIO

1.	INTRODUÇÃO	19
2	OBJETIVOS	21
3	REFERENCIAL TEÓRICO	23
3.1	Metalografia	23
3.2	Imagens	23
3.3	Segmentação	24
3.5	Classificação	28
3.6	Máquina de suporte vetorial.....	30
3.6.1	SVM de margens rígidas	30
3.6.2	SVM de margens suaves	32
3.7	Redes Neurais.....	35
3.7.1	Treinamento da rede.....	39
3.7.2	Redes Neurais Convolucionais.....	41
3.8	MASK R-CNN.....	43
4	MATERIAIS E MÉTODOS	45
4.1	Recursos computacionais	45
4.2	Preparo do banco de dados e funções auxiliares	46
4.2.1	<i>UltraHigh Carbon Steel Micrograph Database</i>	46
4.2.2	Banco de imagens de ferro fundido nodular - Welding	50
4.2.3	Funções auxiliares	51
4.3	Modelos SVM	52
4.3.1	SVM - HOG	52
4.3.2	SVM - HOG - PCA	52
4.4	Modelos de redes neurais convolucionais.....	52
4.5	Aplicação: Mask R-CNN	53
5	RESULTADOS E DISCUSSÕES	55
5.1	SVM	55
5.1.1	SVM - HOG	55
5.1.2	SVM - HOG - PCA	58
5.2	Redes Neurais Convolucionais.....	61
5.3	Aplicação: <i>Mask R-CNN</i>	74
6	CONCLUSÃO	79
7	TRABALHOS FUTUROS.....	81
	REFERÊNCIAS	83

1. INTRODUÇÃO

As propriedades dos materiais, tais como a dureza, resistência ao impacto, condutividade térmica e elétrica são determinadas por sua microestrutura. Por microestrutura, entende-se a natureza, quantidade e distribuição de elementos que formam o material (SEGAL, 2017). O estudo da microestrutura pode ser feito com o uso de fotomicrografias, *i.e.*, o registro de imagens obtidas a partir de técnicas de microscopia óptica ou eletrônica. A análise das micrografias é aplicada, entre outros, na previsão de propriedades, no projeto de novos materiais, na determinação da qualidade de um tratamento térmico e na averiguação do modo de fratura de materiais (CALLISTER, 2012).

Apesar do estudo de microestruturas ser difundido e bem conhecido, a classificação dos microconstituintes ainda é feita, principalmente, por especialistas com o auxílio de quadros metalográficos o que dá margem para incertezas devido ao grau de subjetividade. Quanto à classificação automática, as microestruturas são tipicamente definidas por meio de procedimentos padronizados em metalografia, com o uso de ataques químicos e análise do contraste das imagens, o que ainda é o estado da arte na ciência dos materiais na maioria dos institutos e na indústria. O emprego de métodos para classificação automática dessas estruturas é um grande desafio, pois as microestruturas são, em muitos casos, uma combinação de diferentes fases com subestruturas complexas (AZIMI et al., 2018).

A classificação é uma tarefa que tem como objetivo determinar um mapa que relacione determinadas características à rótulos, sendo que amostras com as mesmas características devem, obrigatoriamente, apresentar o mesmo rótulo. As técnicas de classificação automática de imagens podem ser divididas em duas classes: métodos supervisionados e não supervisionados. Os métodos supervisionados consideram classes definidas a partir de parâmetros previamente identificados. Esses parâmetros devem ser capazes de resumir as propriedades das amostras que as compõem. Nos métodos não supervisionados, todas as informações devem ser obtidas a partir das próprias imagens a serem rotuladas. As amostras que apresentarem propriedades semelhantes, e consequentemente o mesmo rótulo, fazem parte da mesma classe. A diferença entre os classificadores supervisionados e não supervisionado está no conhecimento do significado previamente associados aos rótulos (PEDRINI, SCHWARTZ, 2007).

2 OBJETIVOS

O objetivo principal deste trabalho é apresentar algumas das principais técnicas de classificação de imagens. Os objetivos secundários são:

- Desenvolver um classificador binário utilizando o algoritmo máquina de suporte vetorial (SVM);
- Desenvolver um classificador multiclass utilizando redes neurais convolucionais (CNN);
- Aplicar o algoritmo *Mask R-CNN* na segmentação de fotomicrografias de ferro fundido nodular.

3 RERÊNCIAL TEÓRICO

3.1 Metalografia

Metalografia é um método investigativo da ciência dos materiais com o objetivo de examinar a microestrutura de materiais metálicos e descreve-la de forma quantitativa e qualitativa. Ela cobre o *gap* entre ciência e engenharia. A microestrutura é caracterizada pelo tamanho, forma, arranjo, tipo, distribuição e orientação das fases e dos defeitos. Essas características apresentam forte correlação com as propriedades dos materiais (KAY GEELS, 2007).

A obtenção de microestruturas que sejam representativas do material a ser analisado envolve uma série de etapas previamente definidas a fim de não alterar significativamente a amostra. É importante frisar que independente dos cuidados tomados o processo de preparação das amostras altera em algum grau a microestrutura das mesmas. As etapas necessárias são: o corte, embutimento, desbaste, acabamento superficial e ataque químico (ASM INTERNATIONAL, 2004, KAY GEELS, 2007).

As imagens podem ser analisadas por um especialista em um microscópio óptico ou eletrônico de varredura, com o auxílio dos atlas metalográficos, assistida por computador ou feita de forma automática.

As etapas envolvidas na classificação automática são: a aquisição, o processamento, o reconhecimento e a interpretação. A aquisição compreende registrar as fotomicrografias em um computador, isso é geralmente feito com câmeras acopladas nos microscópios. O processamento visa melhorar a qualidade da imagem, descrever as regiões e objetos de interesse e extrair suas características. Na etapa de reconhecimento, os objetos ou regiões a serem analisados são rotulados de acordo com o domínio do problema. A interpretação consiste em dar significado aos objetos reconhecidos (PEDRINI, SCHWARTZ, 2007).

3.2 Imagens

O processamento de imagens por um computador requer a definição de um modelo matemático que as definam adequadamente. O modelo adotado neste trabalho será o da função de intensidade luminosa onde para cada coordenada (x, y) há uma intensidade mapeada $f(x, y)$. A forma discreta da função é uma matriz $M \times N$ onde os elementos $f(x, y)$ são chamados de pixels (*picture elements*). Uma imagem é aproximada por uma matriz de

$M \times N$ elementos. As dimensões dos pixels, tanto ao longo do eixo x quanto do eixo y , estão relacionados com o espaçamento entre as amostras. Cada pixel é associado por uma escala de intensidade que varia de 0 a 255. Imagens coloridas possuem três canais que podem ser interpretados como matrizes agrupadas correspondentes as cores vermelho, azul e verde. As imagens em preto e branco apresentam apenas um canal. (PEDRINI, SCHWARTZ, 2007).

3.3 Segmentação

A etapa na qual uma imagem é subdividida em regiões ou objetos de interesse é chamada de segmentação. É uma das etapas mais difíceis no processamento e análise de imagens principalmente devido à complexidade envolvida no desenvolvimento de algoritmos de segmentação autônoma. A execução adequada dessa etapa é fundamental para o sucesso dos procedimentos de análise, especialmente quando os problemas envolvem um grande número de objetos a serem identificados. As técnicas de segmentação podem ser divididas em duas categorias: segmentação com base em descontinuidades e segmentação com base em similaridades. Na primeira classe, pressupõem-se que as fronteiras são suficientemente diferentes em relação ao fundo da imagem, permitindo a divisão com base nas descontinuidades locais. O principal exemplo é a segmentação baseada em detecção de bordas. A segunda classe compreende as técnicas utilizadas na divisão das imagens com base nos valores de intensidade e/ou nas suas propriedades (GONZALES, WOODS, 2010).

A segmentação por descontinuidades pode ser realizada através da detecção de pontos, de linhas e de bordas na imagem. Esses métodos envolvem percorrer a figura com o uso de máscaras de convolução, uma operação dada pela Equação 1:

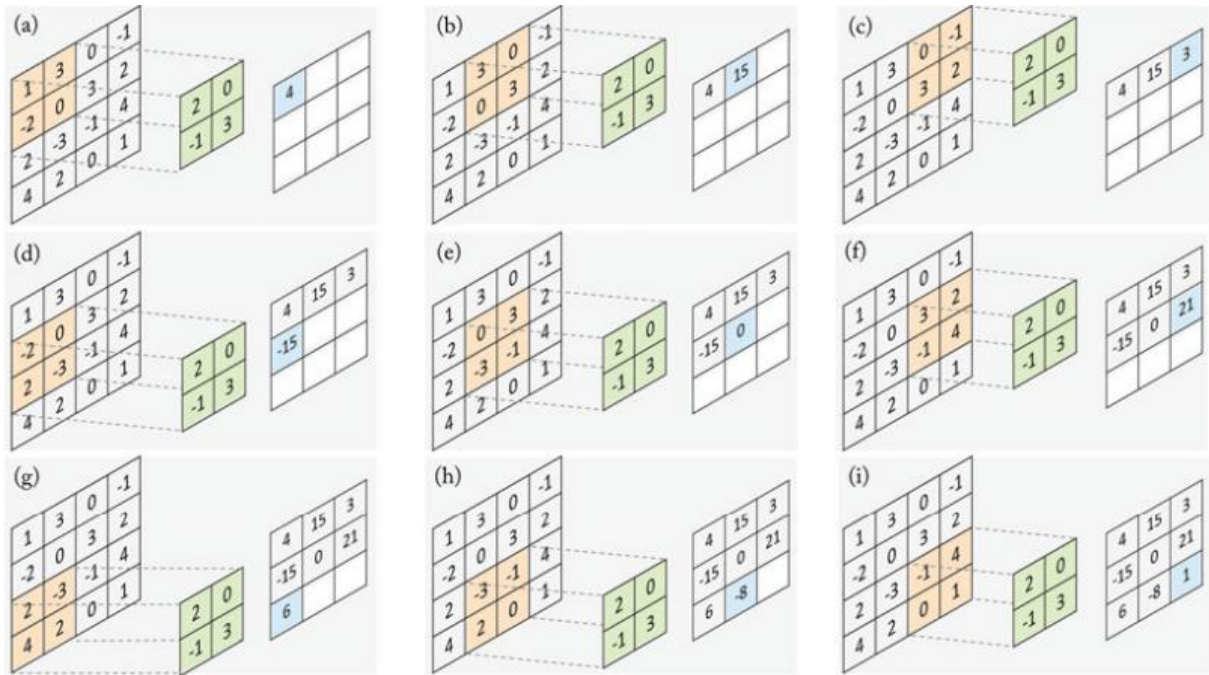
$$(f * g) = \int_{-\infty}^{\infty} f(\tau)g(t - \tau)d\tau \quad (1)$$

Ou na forma discreta:

$$(f * g) = \sum_{k=-\infty}^{\infty} f[k] \cdot g[t - k] \quad (2)$$

Intuitivamente, a convolução pode ser descrita como uma média ponderada da função $f(\tau)$ no momento t . A Figura 1 ilustra a operação.

Figura 1 - Exemplo da operação de convolução



Fonte: KHAN et al, 2018, p. 47

A segmentação baseada em valores de intensidade consiste no uso de técnicas de limiarização, *i.e.*, a classificação dos pixels com base em limiares de modo que os objetos e os fundo da imagem sejam separados em dois ou mais grupos. Seja T o limiar escolhido, uma imagem $g(x, y)$ limiarizada pode ser definida pela Equação 3:

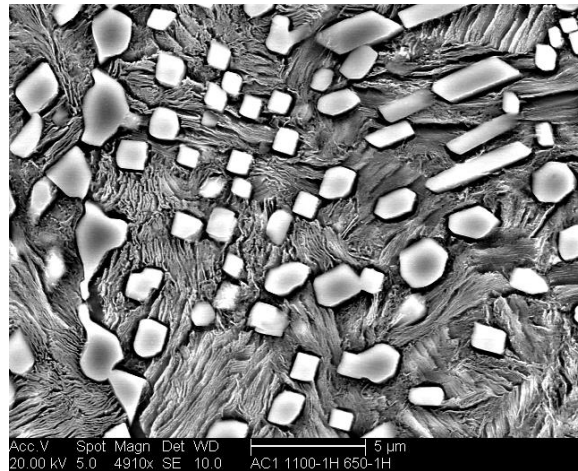
$$g(x, y) = \begin{cases} 0, & \text{se } f(x, y) \leq T \\ 1, & \text{se } f(x, y) > T \end{cases} \quad (3)$$

Nesse caso específico, a técnica é denominada de binarização pois a imagem possui apenas dois valores de intensidade. No caso de dois objetos e um fundo, a imagem segmentada é dada pela Equação 4:

$$g(x, y) = \begin{cases} a, & \text{se } f(x, y) > T_2 \\ b, & \text{se } T_1 < f(x, y) \leq T_2 \\ c, & \text{se } f(x, y) \leq T_1 \end{cases} \quad (4)$$

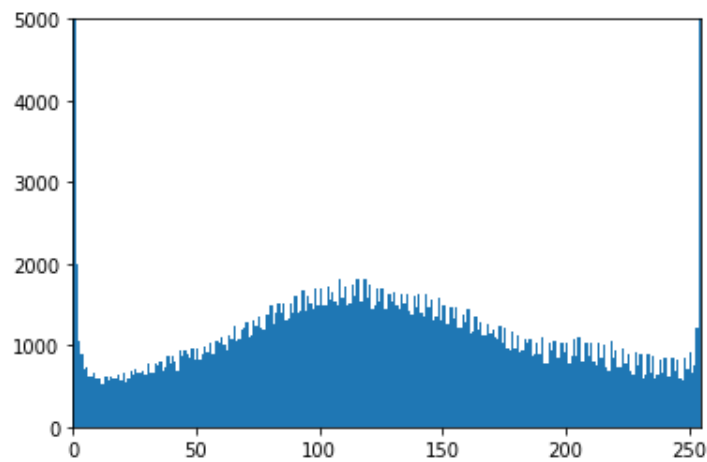
Onde a, b, c são valores de intensidade. Uma das formas de definir o valor de T é com a análise dos histogramas de intensidade, como exemplificado nas Figuras 2 e 3 (PEDRINI, SCHWARTZ, 2007).

Figura 2 – Micrografia de aço contendo perlita-esferoidita



Fonte: *UltraHigh Carbon Steel Micrograph Database*

Figura 3 - Histogramas de Intensidade da micrografia apresentada na Figura 2



Fonte: Elaborada pelo autor.

A segmentação pode ser aplicada para detecção de contornos de grãos, trincas, inclusões e na segmentação de fases (WOJNAR, 1999).

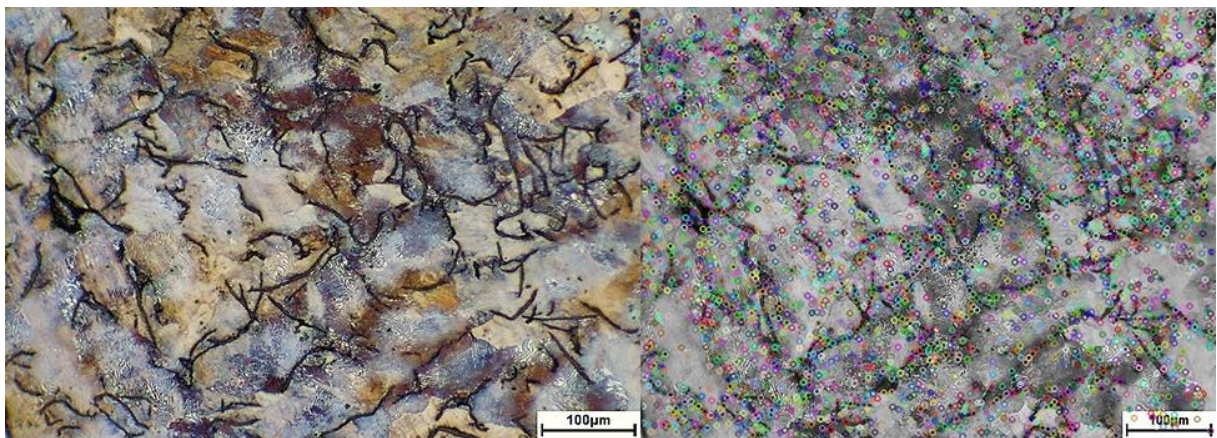
3.4 Extração de características

A extração de características é o estágio no qual é feita a conversão das regiões de interesse em atributos que resultem em informação quantitativa que seja útil na diferenciação das classes dos objetos. Os dados podem ser representados tanto como fronteiras quanto como

regiões completas. O primeiro tipo de representação é adequado quando o foco está nas características externas de forma, como vértices. A representação por regiões é utilizada quando o interesse está nas propriedades internas, como a textura. Dois métodos de extração merecem destaque: a transformada de características invariantes (SIFT) e o histograma de gradientes orientados (HOG) (GONZALES, WOODS, 2010).

A transformada de características invariantes é um método de extração que tem como objetivo transformar a imagem em uma coleção de descritores invariantes a escala, rotação e translação. O algoritmo pode ser dividido em duas etapas: detecção de pontos de interesse e descrição da região ao redor dos pontos. Os pontos de interesse são aqueles nos quais há uma variação que excede um certo limiar. A representação das regiões é feita através da concatenação de múltiplos histogramas de orientação (LOWE, 2004). A Figura 4 apresenta o resultado da aplicação do algoritmo SIFT em uma imagem de ferro fundido cinzento onde os círculos coloridos correspondem aos pontos de interesse detectados.

Figura 4 - Imagem original x representação SIFT

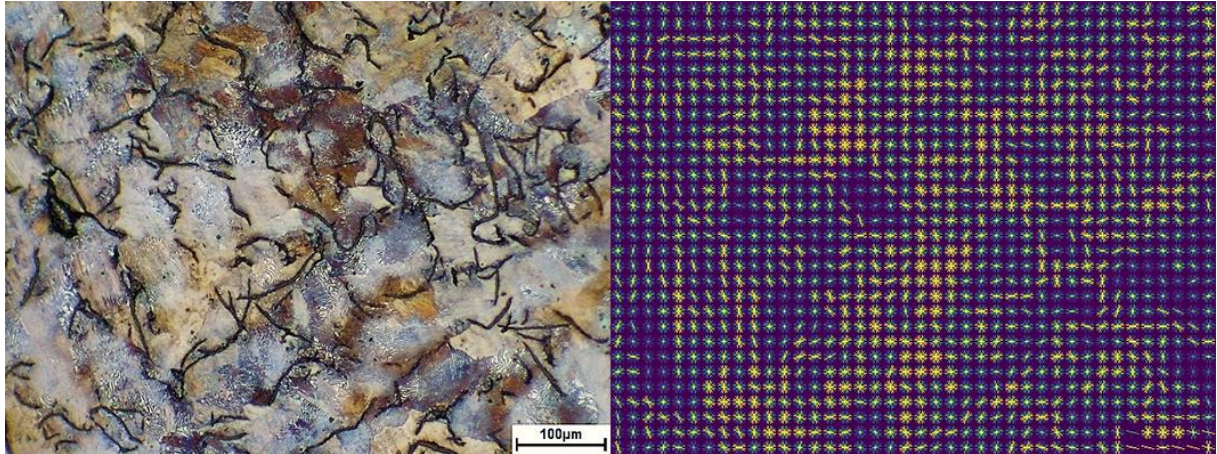


Fonte: Adaptada de Welding (2019)

O histograma de gradientes orientados é um método de extração de características que considera todas as localizações da imagem como pontos de interesse. O método assume que a aparência e o formato de objetos podem ser caracterizados pela distribuição do gradiente de intensidades locais ou pela direção das arestas, mesmo sem conhecimento preciso dos mesmos. Na prática, a imagem é dividida em pequenas regiões e, para cada região, um histograma dos gradientes orientados é gerado. A combinação normalizada dos histogramas

forma a representação (DALAL, TRIGGS, 2005). A Figura 5 apresenta o resultado da aplicação do HOG.

Figura 5 - Imagem original x representação HOG



Fonte: Adaptada de Welding (2019)

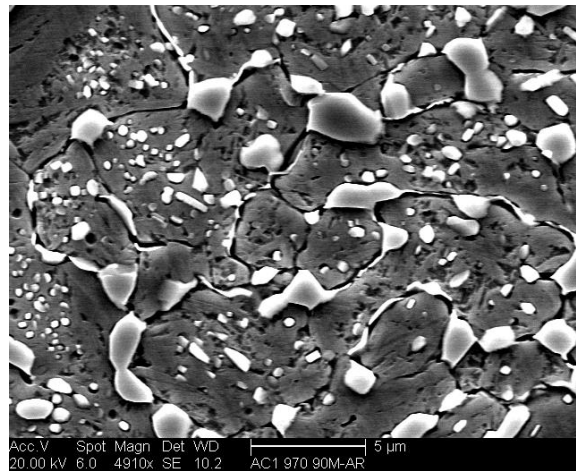
A extração de características também pode ser realizada pelo próprio classificador, como no caso das redes neurais convolucionais.

3.5 Classificação

A classificação pode ser definida como a tarefa de atribuir um rótulo a uma imagem a partir de um conjunto finito de categorias. É um dos principais problemas de visão computacional e possui uma série de aplicações.

Considere a tarefa de classificar a imagem da Figura 6 em uma classe de acordo com seu microconstituente principal: martensita, perlita ou esferoidita. O classificador deve associar à imagem selecionada um vetor de probabilidades referente as classes e rotula-la de acordo com a maior probabilidade. Voltando ao modelo definido na Seção 3.2 no qual uma imagem é modelada a partir de uma matriz e considerando que a Figura 6 possui dimensão $(645 \times 522 \times 1)$, observa-se que a imagem consiste em 336690 números. Se a mesma fosse colorida, esse número passaria para 1010070 $(645 \times 522 \times 3)$. A tarefa, então, é transformar 336690 números em um único rótulo (esferoidita).

Figura 6– Micrografia rotulada por seu microconstituente predominante: “esferoidita”



Fonte: *UltraHigh Carbon Steel Micrograph Database*

Considerando a simplicidade da tarefa de reconhecer uma imagem para um ser humano, é válido destacar as dificuldades envolvidas no desenvolvimento de um algoritmo que execute a mesma função, tais como:

- Variação de ponto de vista: um objeto pode estar orientado de diversas maneiras em relação a câmera;
- Variação de escala: objetos de uma mesma classe podem apresentar diferentes tamanhos (ex: diferentes ampliações no microscópio);
- Oclusão: as classes de interesse podem estar ocultas (ex: sobreposta a outro objeto);
- Condições de iluminação: Os níveis de pixel são extremamente sensíveis a variações de iluminação;
- Variação intraclasses: As classes de interesse podem ter variações abrangentes (ex: grafita do tipo A, B, C, tamanho 1, 2, 3).

O classificador deve receber dados de entrada (tais como imagens rotuladas ou características extraídas), aprender o que cada classe representa e ser capaz de prever os rótulos para imagens nunca vistas pelo modelo. A qualidade do classificador é avaliada pela comparação dos rótulos previstos com os rótulos reais (LI, F.F.; JOHNSON, J.; YEUNG, S, 2017).

Os modelos que apresentarem baixo índice de acerto no novo conjunto de dados quando comparado aos de treinamento, são chamados de modelos em estado de *overfitting*.

No caso de baixa taxa de acerto no conjunto de treinamento, o estado é conhecido como *underfitting*.

3.6 Máquina de suporte vetorial

A máquina de suporte vetorial (SVM) é um modelo de aprendizado supervisionado elaborado originalmente para a classificação linear de duas classes na qual a margem, *i.e.*, a distância mínima entre as classes, é maximizada. A SVM constrói o hiperplano na qual essa condição é satisfeita. Para o caso em que os dados não são linearmente separáveis, é realizada a transformação não linear dos dados em um espaço de características de maior dimensionalidade onde o hiperplano pode ser encontrado (BOSER, 1992) (RYCHETSKY, 2001).

3.6.1 SVM de margens rígidas

O primeiro modelo de SVM introduzido foi chamado de classificador de margem máxima. Ele é válido para dados que são linearmente separáveis no espaço das características. Apesar de ser pouco utilizável na prática, sua formulação é a base para a resolução de problemas mais complexos (CRISTIANINI, TAYLOR, 2014).

A construção deste classificador inicia com a premissa de que os dados são linearmente separáveis no espaço de características e que exista pelo menos uma combinação de parâmetros w e b para a qual a Equação 5 seja satisfeita.

$$f(x) = w \cdot x + b = 0 \quad (5)$$

A Equação 5 representa um hiperplano, onde $w \cdot x$ é o produto escalar entre os vetores w e x , w é o vetor normal ao hiperplano e $\frac{b}{||w||}$ é a distância do hiperplano em relação à origem. A Equação 6 define a função de classificação a partir do hiperplano.

$$h(x_i) = \begin{cases} +1, & \text{se } w \cdot x + b \geq 0 \\ -1, & \text{se } w \cdot x + b < 0 \end{cases} \quad (6)$$

O número de hiperplanos possíveis, a partir de (6), é infinito pois é possível multiplicar w e b por uma constante k . Portanto, define-se o hiperplano canônico como aquele que satisfaz a Equação 7:

$$|w \cdot x_i + b| = 1 \quad (7)$$

O que implica na Inequação 8:

$$\begin{cases} w \cdot x_i + b \geq +1, & \text{se } y_i = +1 \\ w \cdot x_i + b \leq -1, & \text{se } y_i = -1 \end{cases} \quad (8)$$

A Equação 9 define a margem em relação a fronteira.

$$M = \min_{(i=1\dots m)} y_i \left(\frac{w}{\|w\|} \cdot x + \frac{b}{\|w\|} \right) \quad (9)$$

Com isso, a seleção do hiperplano se torna um problema de otimização: encontrar os valores de w que maximizam a margem. A margem máxima ocorre ao encontrar o mínimo de $\|w\|$, como mostra a Equação 10:

$$\min_{w,b} \frac{1}{2} \|w\|^2 \quad (10)$$

Com a restrição dada pela Inequação 11:

$$y_i(w \cdot x_i + b) - 1 \geq 0, \text{ para todo } i = (1, \dots, n) \quad (11)$$

As restrições são impostas a fim de assegurar que não existam dados de treinamento entre as margens que separam as classes. Esse problema de otimização é quadrático e bem documentado (SCHÖLKOPF, SMOLA, 2002). A Equação 12 apresenta o resultado, onde α_i^* são os parâmetros de Lagrange, $\text{sgn}()$ é a função sinal, w é calculado pela Equação 13 e b^* por 14.

$$g(x) = \text{sgn}(f(x)) = \text{sgn}\left(\sum_{x_i \in SV} y_i \alpha_i^* x_i \cdot x + b^*\right) \quad (12)$$

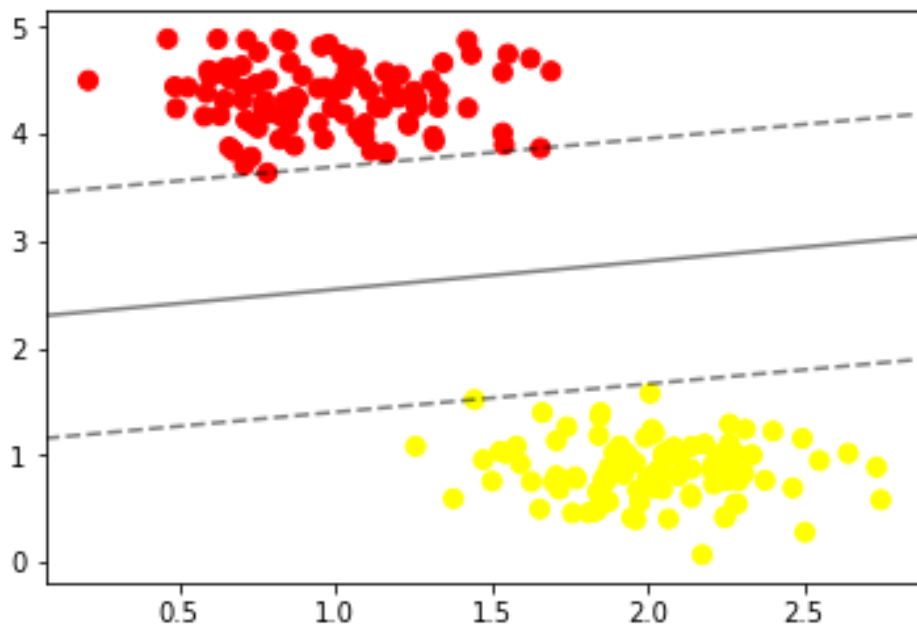
$$w = \sum_{i=1}^n \alpha_i y_i x_i \quad (13)$$

$$b^* = \frac{1}{n_{SV}} \sum_{x_j \in SV} \left(\frac{1}{y_j} - \sum_{x_i \in SV} \alpha_i^* y_i x_i \cdot x_j \right) \quad (14)$$

Os dados com $\alpha_i^* > 0$ são os vetores de suporte, pois são os únicos utilizados na determinação do hiperplano. A Equação 10 representa o hiperplano de máxima margem definido pela localização dos vetores de suporte (BISHOP,2006) (LORENA,2007).

A Figura 7 ilustra a SVM de margens rígidas e os vetores de suporte (em linha tracejada).

Figura 7 - SVM de margens rígidas



Fonte: Elaborada pelo autor

3.6.2 SVM de margens suaves

A SVM de margens rígidas foi definida para resultar na separação exata dos dados de treinamento, não permitindo a sobreposição de classes. Essa condição não é satisfeita para a maior parte dos casos reais, pois o conjunto de dados pode não ser linearmente separável por uma série de fatores tais como *outliers*, ruídos ou a própria natureza do problema.

Para resolver esses casos mais gerais, é necessário introduzir variáveis de folga (ξ) permitindo que alguns dados possam violar a restrição imposta, como mostra a Inequação 15. O classificador que aceita *outliers* é chamado de SVM com margens suaves (VAPNIK, 1995)

$$y_i (w \cdot x_i + b) \geq 1 - \xi_i, \xi_i \geq 0, \forall i = 1, \dots, n \quad (15)$$

Esse procedimento permite que alguns dados estejam entre os hiperplanos $H1$ e $H2$ e a ocorrência de erros de classificação. O problema de minimização apresentado na Equação 9 é reformulado na Equação 16:

$$\text{Minimizar}_{w,b,\xi} \frac{1}{2} \|w\|^2 + C(\sum_{i=1}^n \xi_i) \quad (16)$$

O termo C é um parâmetro de regularização que determina o *trade-off* entre alcançar um baixo nível de erros nos dados de treinamento e minimizar a norma dos pesos. Valores baixos de C resultam em margens suaves, enquanto valores mais altos em margens mais rígidas.

A Equação 17 apresenta o problema reformulado, onde os parâmetros α_i são multiplicadores de Lagrange.

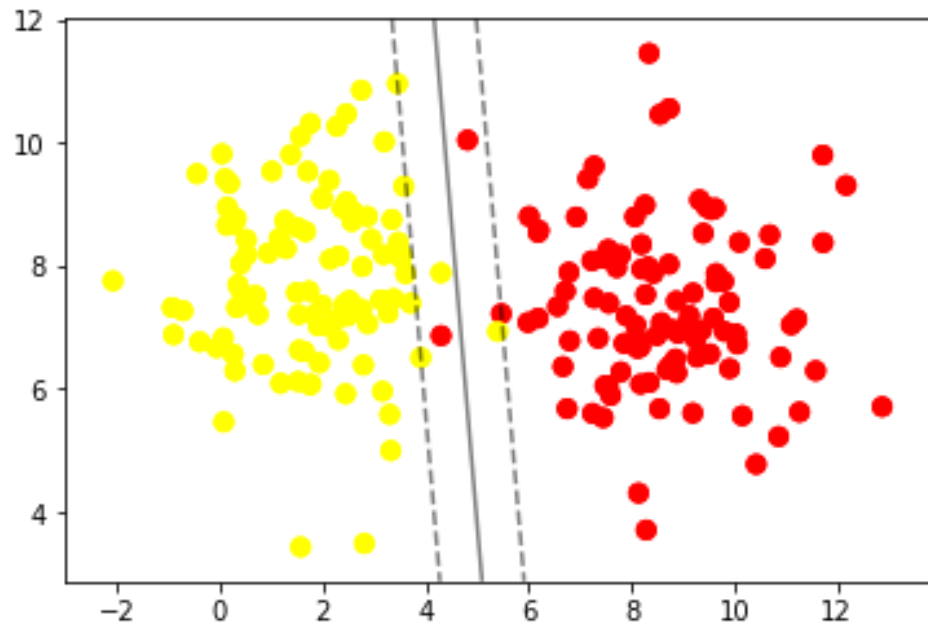
$$\text{Maximizar}_{\alpha} \sum_{i=1}^n \alpha_i - \frac{1}{2} \sum_{i,j=1}^n \alpha_i \alpha_j y_i y_j (x_i \cdot x_j) \quad (17)$$

Com as restrições dadas pelas equações 18 e 19:

$$\begin{cases} 0 \leq \alpha_i \leq C, \forall i = 1, \dots, n & (18) \\ \sum_{i=1}^n \alpha_i y_i = 0 & (19) \end{cases}$$

A Figura 8 ilustra a SVM de margens suaves.

Figura 8 - SVM de margens suaves



Fonte: Elaborada pelo autor

Um último cenário a ser considerado é a aplicação de SVM para problemas não lineares. Neste caso, além dos termos de folga (ξ_i), é necessário fazer o mapeamento do vetor de entrada para um espaço de característica com mais alta dimensionalidade. Esse procedimento é resultado do Teorema de Cover, no qual ele afirma que o espaço de entradas pode ser transformado em um espaço de características no qual há alta probabilidade de os dados serem linearmente separáveis. As condições necessárias do teorema são: transformação não linear e dimensão do espaço de características suficientemente alta (COVER, 1965).

O mapeamento é feito a partir da utilização de funções *kernel*, que são matrizes de convolução utilizadas na operação definida na Seção 3.3, equações 1 e 2. Essa função recebe dois pontos x_i e x_j e computa o produto escalar desses dados no espaço de características. A função *kernel* é dada pela Equação 20:

$$K(x_i, x_j) = \phi(x_i) \cdot \phi(x_j) \quad (20)$$

Os *kernels* mais utilizados são os polinomiais, os sigmoidais e os gaussianos. Cada *kernel* apresenta parâmetros que devem ser decididos pelo usuário. A Tabela 1 apresenta a função *kernel* e seus respectivos parâmetros (LORENA, 2007)

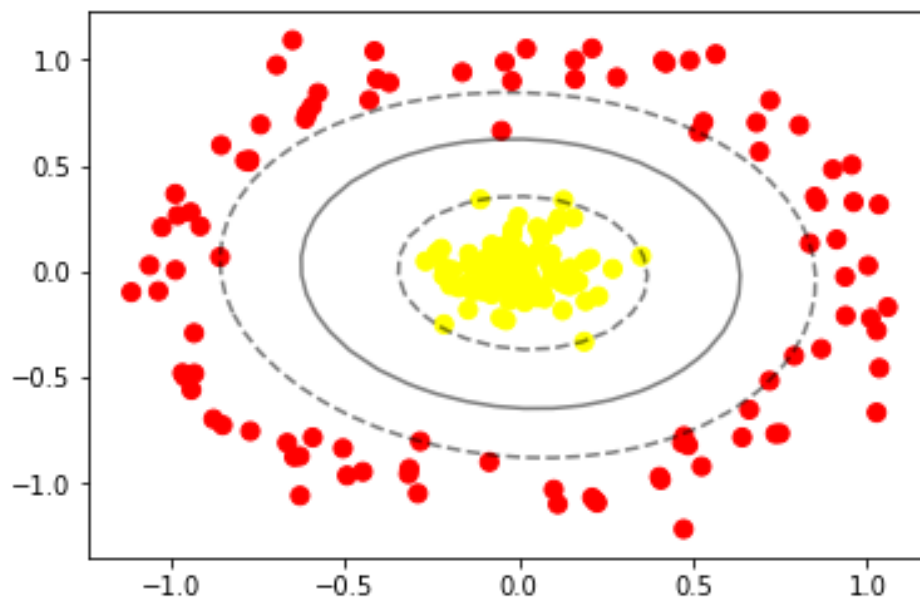
Tabela 1 - Função *Kernel* e seus respectivos parâmetros

Tipo de <i>Kernel</i>	Função $K(x_i, x_j)$	Parâmetros
Polinomial	$(\delta(x_i \cdot x_j) + \kappa)^d$	δ, κ e d
Gaussiano	$\exp(-\sigma \ x_i - x_j\ ^2)$	σ
Sigmoidal	$\tanh(\delta(x_i \cdot x_j) + \kappa)$	δ e κ

Fonte: LORENA, 2007, p.62

A Figura 9 ilustra a SVM de margens suaves com dados não separáveis linearmente.

Figura 9 - SVM de margens suaves



Fonte: Elaborada pelo autor

3.7 Redes Neurais

As redes neurais são modelos de aprendizado de máquina inspirados no funcionamento do cérebro humano: unidades de processamento interconectados formando uma rede. O objetivo da rede é aproximar alguma função $f(x)$ através do mapeamento $y = f(x, \theta)$, onde os valores dos parâmetros (θ) devem ser aprendidos pela rede e resultar na melhor aproximação (GOODFELLOW, 2016).

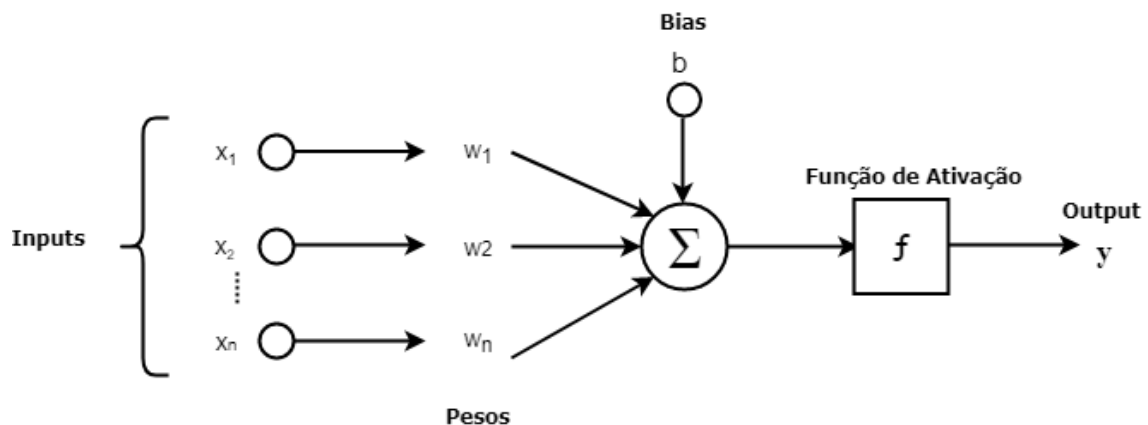
A unidade básica de processamento de informações de uma rede neural é chamada de *perceptron*. A Figura 10 apresenta o funcionamento básico dessa unidade, que é o bloco fundamental do design das famílias de redes neurais. Os elementos básicos do modelo envolvem os sinais (sinapses), os quais são caracterizados por pesos (w), uma combinação linear de w e uma função de ativação cujo objetivo é limitar a amplitude de saída do sinal (HAYKIN, 2009).

O modelo é matematicamente representado pela Equação 21:

$$\sum_i^n w_i x_i + b \quad (21)$$

Onde b representa o viés, termo que tem como objetivo deixar o modelo mais geral, e x os dados de entrada. O processo de cálculo é chamado de passo para frente (*forward pass*).

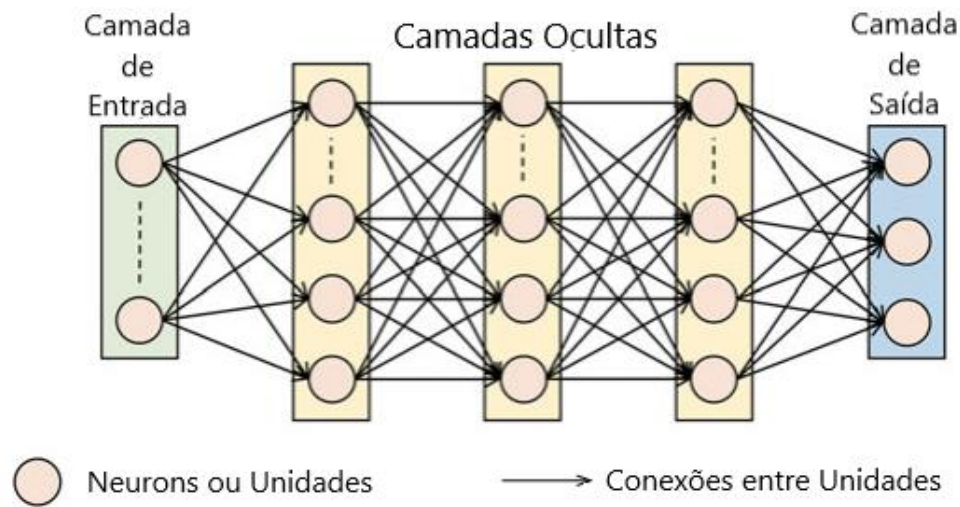
Figura 10 - Arquitetura de um *Perceptron*



Fonte: HAYKIN, 2009, p.11, adaptada.

O modelo deve aprender os valores de w , parâmetros do sistema, para que os resultados corretos sejam alcançados a partir dos dados de entrada (x). O modelo *perceptron* com camada única é limitado a aproximar funções lineares, pois o seu funcionamento é baseado em produtos escalares. Essa limitação é removida ao ser feita a adição de camadas intermediárias, chamadas de camadas escondidas. O modelo que utiliza múltiplas camadas é chamado de *perceptron* de camadas múltiplas (MLP) (ALPAYDIN, 2004).

A Figura 11 ilustra a arquitetura do MLP.

Figura 11 - *Perceptron* de camadas múltiplas

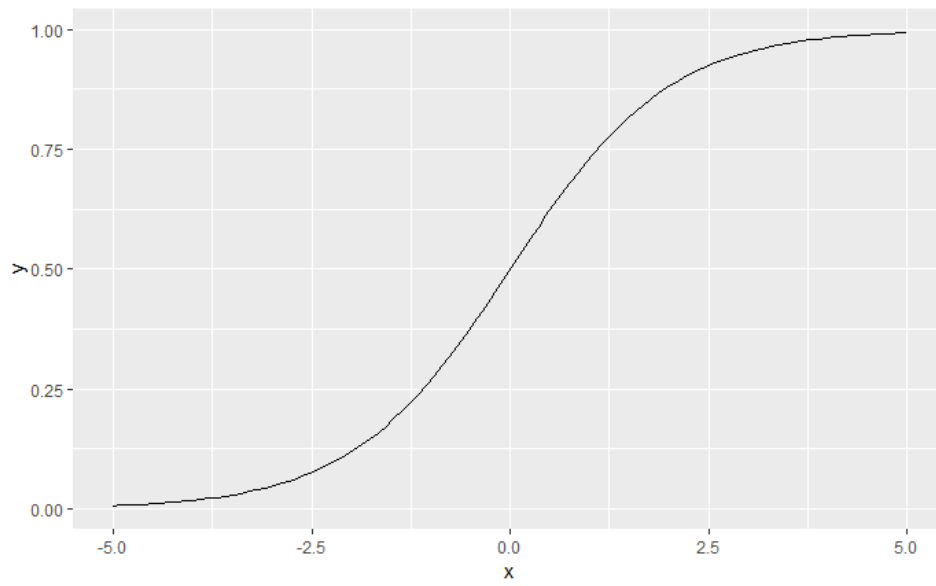
Fonte: KHAN et al, 2018, p.32

O modelo MLP apresenta as seguintes características: cada *neuron* da rede inclui uma função de ativação não linear e diferenciável, a rede contém uma ou mais camadas ocultas tanto da camada de entrada quanto da de saída e apresenta alto grau de conectividade. A função da camada escondida é de atuar na detecção de atributos, *i.e.*, durante o processo de treinamento, os *neurons* pertencentes a essas camadas descobrem, gradualmente, as características que melhor representam os dados. Isso é feito através das transformações não lineares associadas às funções de ativação dos sinais de entrada. As funções mais utilizadas nos modelos são a sigmoide (Figura 12 e Equação 22), *rectified linear unit* (Figura 13, Equação 23) e tangente hiperbólica (Figura 14, Equação 24).

Esses pontos são responsáveis pela versatilidade da rede e, ao mesmo tempo, pelas deficiências no conhecimento de como ela se comporta, pois, a presença de não linearidade e alta conectividade torna complicada a análise teórica do modelo (HAYKIN, 2009)

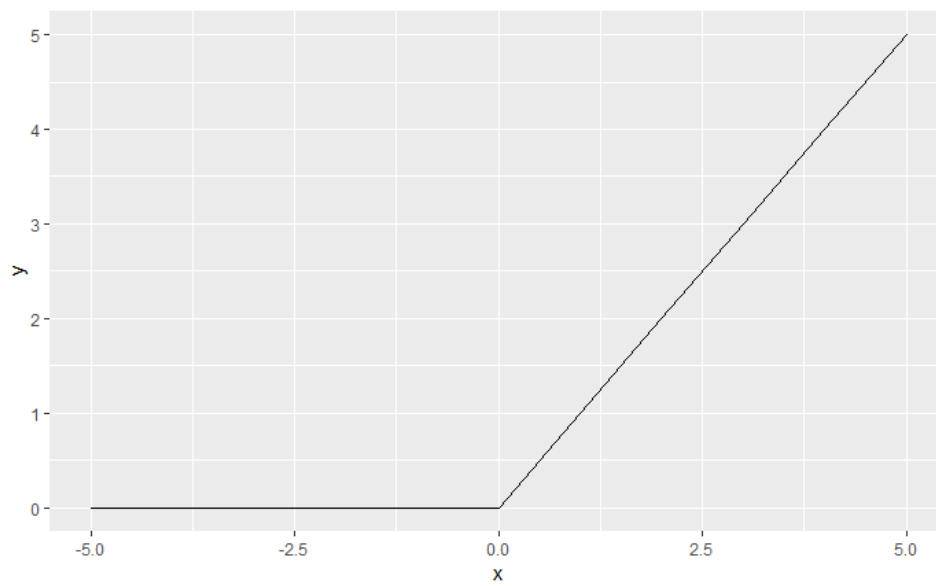
$$g(z) = \frac{1}{1 + e^{-z}} \quad (\text{sigmoide}) \quad (22)$$

Figura 12 - Função sigmoide



Fonte: Elaborada pelo autor

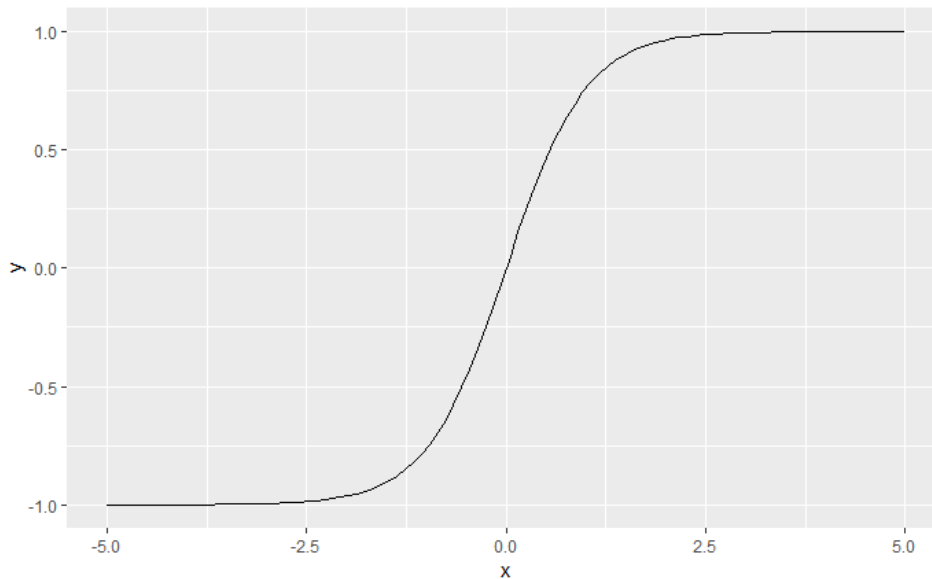
$$g(z) = \max(z, 0) \text{ (ReLU) } (23)$$

Figura 13 - Função *rectified linear unit*

Fonte: Elaborada pelo autor

$$g(z) = \frac{e^z - e^{-z}}{e^z + e^{-z}} \text{ (tanh) } (24)$$

Figura 14 - Função tangente hiperbólica



Fonte: Elaborada pelo autor

A quantidade de camadas, *neurons* por camada e como os mesmos são conectados são todos parâmetros da arquitetura da rede e devem ser escolhidos pelo usuário.

3.7.1 Treinamento da rede

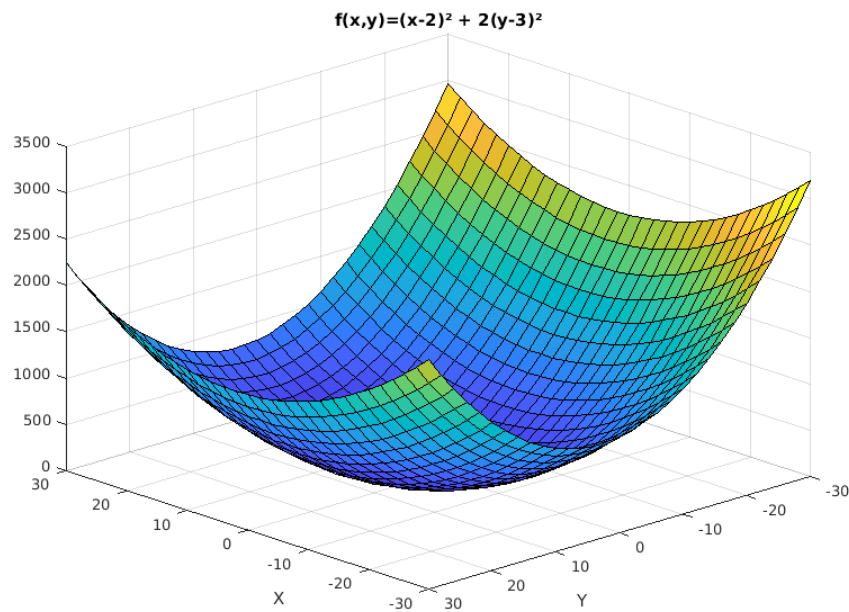
O treinamento da rede neural consiste em ajustar os pesos a fim de obter um classificador adequado. Isso é feito através da minimização da função de custo. Essa função compara, durante o treinamento, os resultados da rede com o rótulo verdadeiro, quanto maior a discrepância, maior o valor da função. Existem diversas funções de custo e seu uso depende do domínio do problema. A função mais simples é apresentada na Equação 25 onde y representa o resultado da rede, y' o rótulo verdadeiro e m o número de amostras.

$$L(y, \hat{y}) = \frac{1}{m} \sum_{i=1}^m (y_i - \hat{y}_i)^2 \quad (25)$$

O algoritmo mais utilizado no problema de minimização da função de custo é chamado de gradiente descendente. Intuitivamente, o algoritmo busca, a cada iteração, os

parâmetros da função que fornecem o menor valor de custo seguindo a inclinação da mesma. A Figura 15 ilustra o processo.

Figura 15 - Representação do método do gradiente



Fonte: neural-network.io (2019)

Matematicamente o processo é dado por:

1. Escolher um ponto de partida x_0
2. Fazer a atualização a cada iteração: $w^{k+1} = w^k - lr(\frac{\partial L}{\partial w^k})$
3. Repetir 2 até encontrar um ponto de mínimo

Onde w é o peso e lr a taxa de aprendizagem

As derivadas parciais são em relação aos pesos (w) pois os dados de entrada x e y são fixos durante o treinamento. A taxa de aprendizagem controla a velocidade na qual o algoritmo aprende, ou seja, a velocidade com a qual os pesos mudam. Valores excessivos desse parâmetro podem impedir a convergência do algoritmo.

O gradiente não é calculado de forma analítica, pois as arquiteturas modernas possuem milhões de *neurons* o que torna o problema muito complexo e computacionalmente caro. Para isso é empregado o algoritmo *backpropagation* (LE CUN, 1988).

O processo de treinamento pode ser descrito pelo seguinte algoritmo:

1. Inicializar os pesos com valores aleatórios
2. Executar o *forward pass*
3. Calcular os erros na camada de saída
4. Calcular os gradientes com respeito aos pesos
5. Aplicar o gradiente descendente para atualizar os pesos buscando o mínimo do custo
6. Repetir os passos 2-5 para todos os dados de treinamento

Um grande avanço das redes neurais em relação a outros métodos de aprendizado é o chamado aprendizado de ponta a ponta (*end-to-end learning*), onde as características intermediárias são aprendidas automaticamente através dos dados de entrada e saída. No exemplo da classificação de imagens, o modelo recebe a imagem e o rótulo como dados de treinamento e aprende as características intermediárias relevantes para a classificação por si só, ou seja, diferentemente das SVM, as redes neurais não necessitam de uma etapa externa de extração de características pois o próprio modelo cuida disso.

3.7.2 Redes Neurais Convolucionais

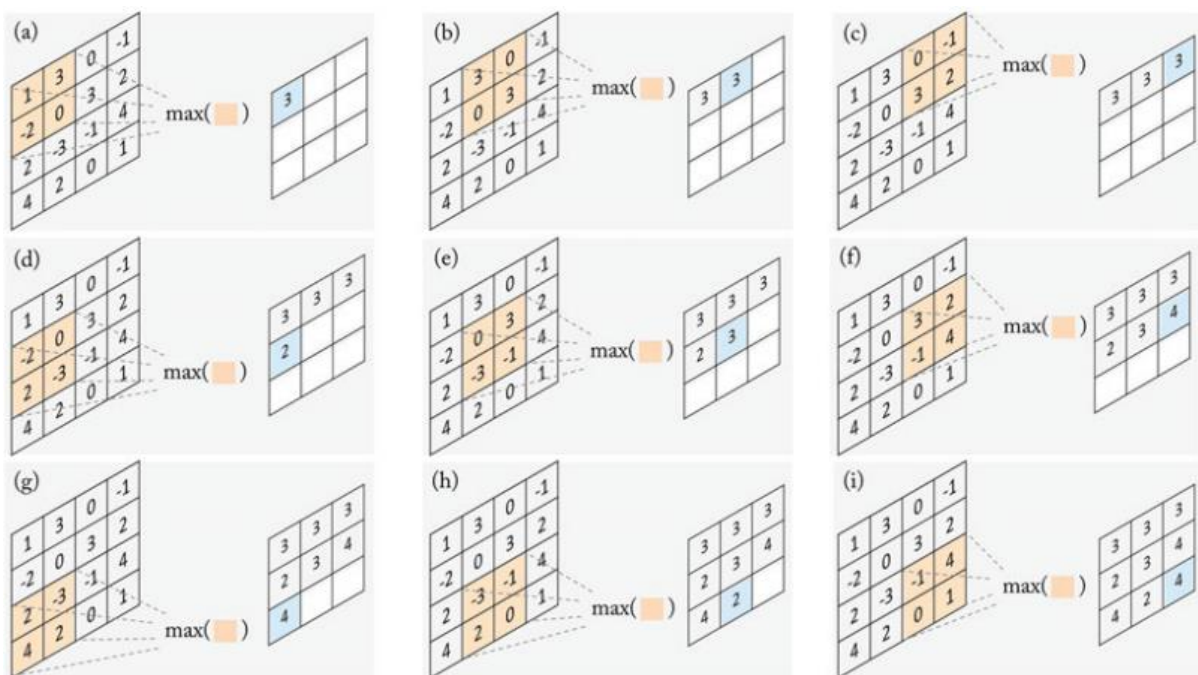
As redes neurais convolucionais (CNN) são uma classe de redes aplicadas em problemas nos quais os dados de entradas são matrizes. Elas operam de maneira similar as MLP, a principal diferença é que cada unidade em uma camada da CNN é um filtro de dimensão dois ou maior que é utilizado na convolução com os dados de entrada daquela camada. Assim, as suas camadas são arquitetadas de modo que as características sejam extraídas de maneira progressiva ao longo da rede: as primeiras camadas detectam elementos mais básicos como arestas e as finais identificam objetos mais complexos. As camadas básicas das CNNs são a camada de convolução, a camada de *pooling* e a camada totalmente conectada (KHAN et al., 2018).

A primeira e principal tipo de camada da rede é a de convolução. Nessa camada ocorre o produto escalar entre duas matrizes: a matriz de entrada e uma matriz com os parâmetros a serem aprendidos, chamada de *kernel* ou filtro. Como definido na Seção 3.3, a operação de convolução ocorre com o deslizamento do filtro sobre a imagem. O resultado da operação é uma nova matriz chamada de mapa de características, ou seja, o elemento responsável por detectar as características é o filtro. Cada tipo de filtro produz um mapa de características diferente capaz de identificar aspectos distintos, tais como curvas, arestas, etc. Vale destacar

que diferentemente dos métodos tradicionais de processamento de imagens, os filtros das CNNs são aprendidos diretamente pela rede durante o treinamento. Os parâmetros a serem especificados são o número de filtros e o seu tamanho. De uma maneira geral, quanto maior o número de filtros maior será o número de características extraídas. Após a etapa de convolução é feita a aplicação de alguma função não linear tal como a RELU (*Rectified linear unit*) (VENKATESAN, LI, 2018).

A segunda camada é a de *pooling* e tem a função de reduzir a dimensão de cada mapa de características enquanto mantém as informações mais relevantes dela. Isso é feito através de técnicas de sumarização estatística nas vizinhanças de um elemento tais como o máximo, a média e a soma dos elementos. A operação utiliza um *kernel* de tamanho F e passo S . A redução dos mapas de características é progressiva e traz vantagens como a redução do número de parâmetros, o que ajuda a controlar o *overfitting*, e torna a rede invariante a pequenas transformações o que possibilita detecção de imagens independentemente da sua localização. A Figura 16 ilustra a operação de *max pooling*. (VENKATESAN, LI, 2018).

Figura 16 - Operação de *max pooling*



Fonte: (KHAN et al., 2018, p.53)

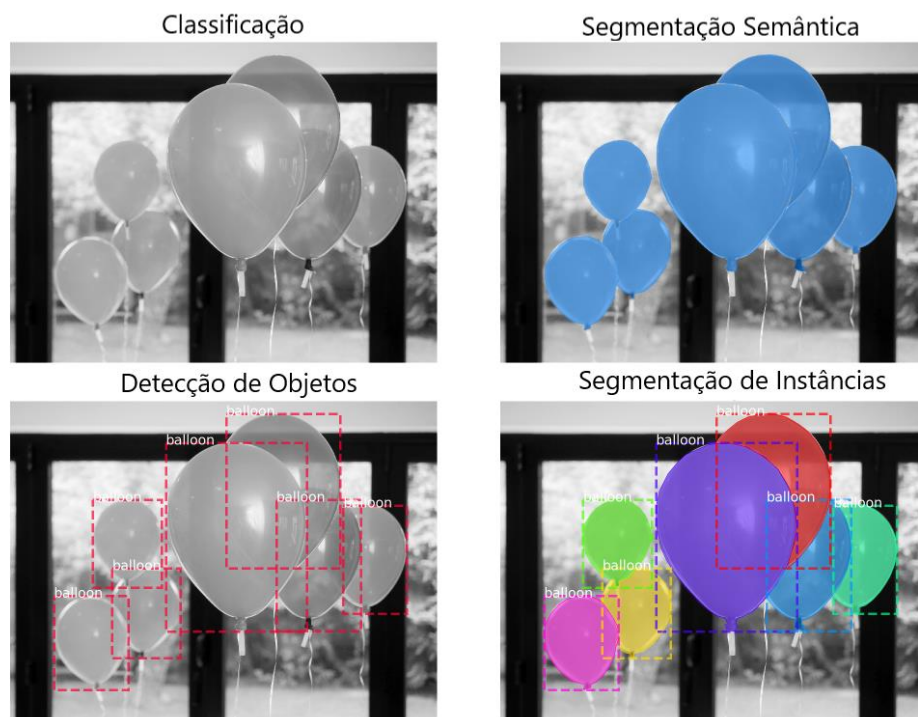
A terceira e última camada é a totalmente conectada. Essa é basicamente uma MLP acoplada a CNN. O objetivo dessa camada é receber as características extraídas nas camadas anteriores e aprender o significado delas para a resolução do problema (KHAN et al., 2018).

O processo de treinamento das CNNs é similar ao das MLP.

3.8 MASK R-CNN

A *Mask R-CNN* (*regional convolutional neural network*) é uma rede neural utilizada para resolver problemas de segmentação de instancia. Esse tipo de problema envolve detectar e delinear cada objeto de interesse presente na imagem analisada. Isso permite, por exemplo, contar objetos distintos como células, grãos, etc. A segmentação de instancias é um problema extremamente complexo pois envolve a classificação de cada pixel para um grupo desconhecido de instancias a priori. A Figura 17 ilustra a diferença entre classificação e segmentação de instancias.

Figura 17 - Classes de problemas em visão computacional



Fonte: engineering.matterport.com (2019)

Note que a segmentação de instancias envolve classificar corretamente o objeto, sua localização e quantidade dos mesmos na figura.

A *Mask R-CNN* funciona em dois estágios: no primeiro são geradas regiões de interesse, *i.e.*, as áreas com maior probabilidade de conter um objeto, enquanto no segundo

estágio são geradas as caixas delimitadoras e as máscaras de segmentação. Ambos estágios estão relacionados a uma CNN que serve como esqueleto, podendo ser a VGG (*Visual Geometry Group*), ResNet (*Residual Neural Network*), entre outras. Esse esqueleto tem a função de extrair as características das imagens e gerar mapas de características que servem como dados de entrada para as etapas seguintes. A etapa na qual é feita a busca por regiões de interesse é realizada por uma rede chamada de RPN (*Region Proposal Network*) através de objetos chamados de âncoras. A partir de cada âncora são gerados dois tipos de informações: uma classe (*background* ou *foreground*) e um refinamento da posição da caixa delimitadora (REN et al., 2015), (HE et al., 2018).

4 MATERIAIS E MÉTODOS

Foram realizados três estudos de caso com a finalidade de colocar em prática os conceitos apresentados no trabalho. O primeiro estudo consiste no desenvolvimento de um modelo binário para a classificação de aços cuja microestrutura predominante é esferoidita ou perlita. O algoritmo escolhido foi o SVM utilizando HOG como extrator de características.

O segundo estudo de caso foi o desenvolvimento de um classificador de fotomicrografias de aços de alto carbono utilizando até cinco classes: esferoidita, martensita, *network*, perlita e perlita + esferoidita. Três modelos foram treinados em ordem crescente de número de classes: três a cinco. A estratégia adotada foi a de transferência de aprendizado (*transfer learning*) e a rede base foi a VGG16 (SIMONYAN, ZISSERMAN, 2015)

A terceira aplicação foi o uso da rede *Mask R-CNN* na segmentação de instancias em fotomicrografias de ferro fundido nodular.

4.1 Recursos computacionais

Parte da programação foi realizada na plataforma Google Colab, um serviço gratuito de pesquisa na nuvem que permite o uso de equipamento adequado para tarefas computacionalmente complexas por até 12 horas diariamente. A linguagem de programação escolhida foi python (PYTHON CORE TEAM 2015), (ROSSUM, 1995), devido a quantidade de bibliotecas existentes, como *keras* (CHOLLET, 2015) e *tensorflow* (ABADI et al., 2015).

As Figuras 7, 8 e 9 foram produzidas utilizando a linguagem de programação python, com os pacotes *scikit-learn* (PEDREGOSA et al., 2011) e *matplotlib* (HUNTER, 2007)

As Figuras relativas as funções de ativação (Figuras 12, 13 e 14) foram elaboradas utilizando a linguagem de programação R (R CORE TEAM, 2016), com o pacote *ggplot2* (WICKHAM, 2016).

A aplicação da *Mask-RCNN* foi realizada com a implementação feita por Matterport Inc (ABDULLA, 2017), sendo que a ferramenta utilizada para anotar as Figuras foi a *VGG Image Annotation* (DUTTA, ZISSERMAN, 2019).

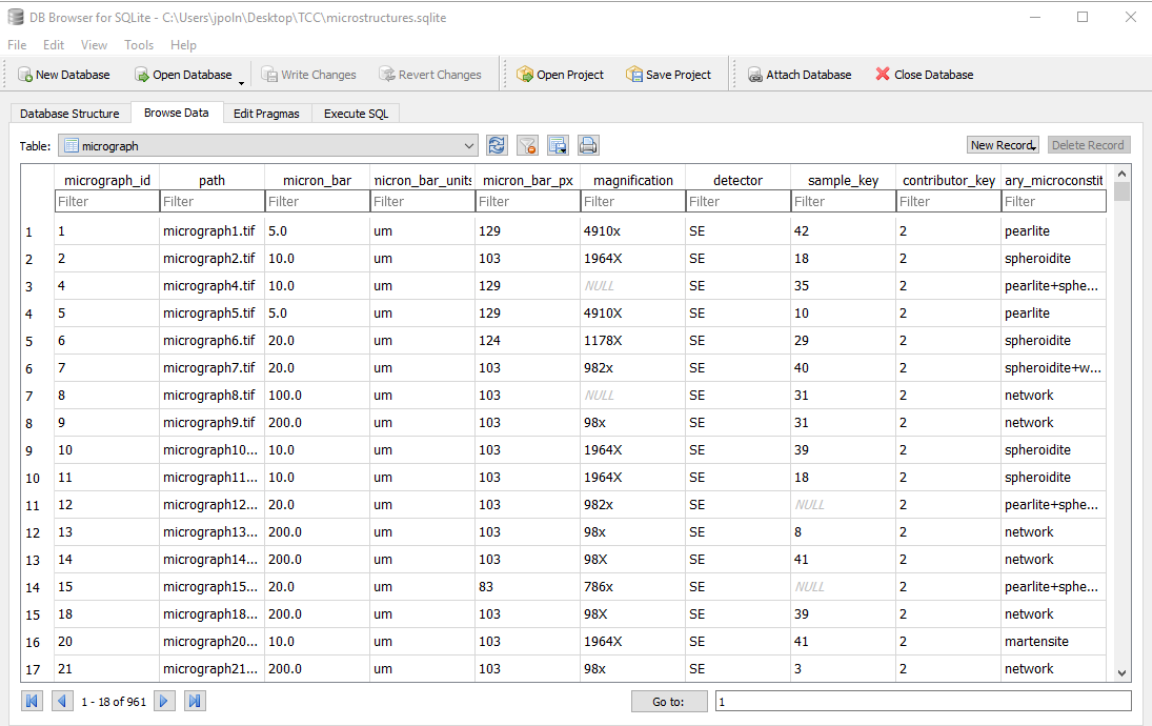
4.2 Preparo do banco de dados e funções auxiliares

4.2.1 *UltraHigh Carbon Steel Micrograph Database*

O banco de imagens utilizado para os primeiros dois estudos de caso foi o *UltraHigh Carbon Steel Micrograph Database* (DECOST et al., 2017). Ele é composto por 961 micrografias de dimensões 645x522x1 e rotuladas de acordo com o microconstituente primário: martensita, *network*, perlita, perlita + esferoidita, perlita + ferrita de widmanstatten, esferoidita e esferoidita + widmanstatten.

Foi feito o fichamento das imagens no banco com a ajuda dos softwares SQLite e Excel. A Figura 18 mostra parte do banco de dados original.

Figura 18 - *UltraHigh Carbon Steel Micrograph Database*

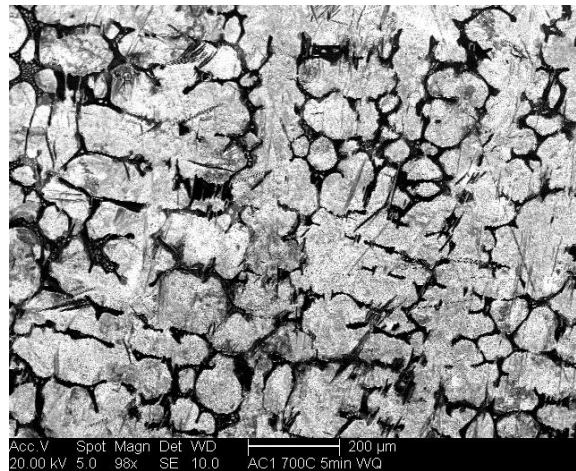


	micrograph_id	path	micron_bar	micron_bar_units	micron_bar_px	magnification	detector	sample_key	contributor_key	ary_microconstit
1	1	micrograph1.tif	5.0	um	129	4910x	SE	42	2	pearlite
2	2	micrograph2.tif	10.0	um	103	1964X	SE	18	2	spheroidite
3	4	micrograph4.tif	10.0	um	129	NULL	SE	35	2	pearlite+sphe...
4	5	micrograph5.tif	5.0	um	129	4910X	SE	10	2	pearlite
5	6	micrograph6.tif	20.0	um	124	1178X	SE	29	2	spheroidite
6	7	micrograph7.tif	20.0	um	103	982x	SE	40	2	spheroidite+w...
7	8	micrograph8.tif	100.0	um	103	NULL	SE	31	2	network
8	9	micrograph9.tif	200.0	um	103	98x	SE	31	2	network
9	10	micrograph10...	10.0	um	103	1964X	SE	39	2	spheroidite
10	11	micrograph11...	10.0	um	103	1964X	SE	18	2	spheroidite
11	12	micrograph12...	20.0	um	103	982x	SE	NULL	2	pearlite+sphe...
12	13	micrograph13...	200.0	um	103	98x	SE	8	2	network
13	14	micrograph14...	200.0	um	103	98X	SE	41	2	network
14	15	micrograph15...	20.0	um	83	786x	SE	NULL	2	pearlite+sphe...
15	18	micrograph18...	200.0	um	103	98X	SE	39	2	network
16	20	micrograph20...	10.0	um	103	1964X	SE	41	2	martensite
17	21	micrograph21...	200.0	um	103	98x	SE	3	2	network

Fonte: Elaborada pelo autor

Nota-se que o banco possui alguns dados faltantes. Os dados foram exportados para um arquivo em .xls e foi feita a correção dos mesmos da seguinte forma: as imagens correspondentes aos elementos com dados faltantes foram conferidas e as devidas correções foram em uma planilha excel. A Figura 19 apresenta as informações contidas nas fotomicrografias.

Figura 19 - Fotomicrografia 306 - Network

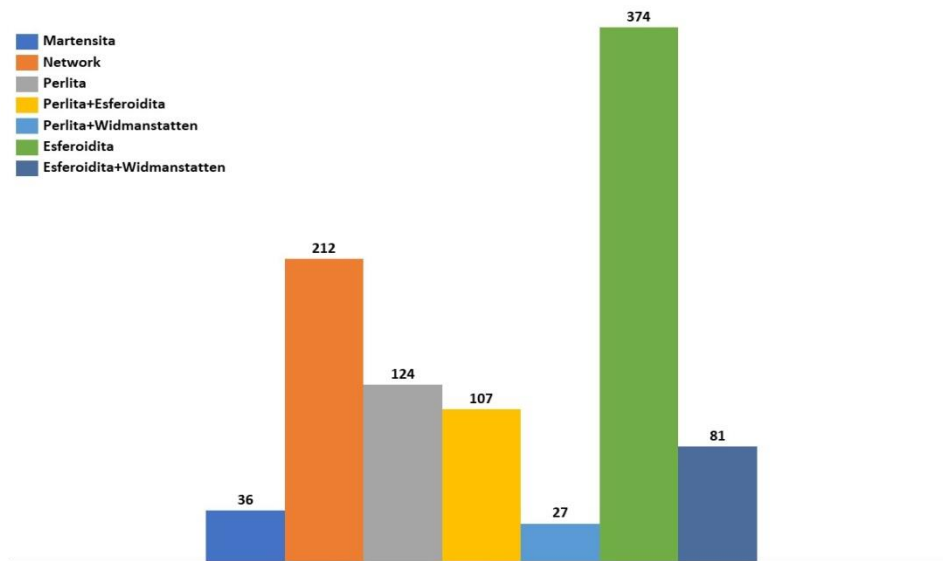


Fonte: *UltraHigh Carbon Steel Micrograph Database*

Após a correção dos dados, foi necessário criar um *script* para organizar as micrografias em pastas com o nome de suas classes correspondentes, pois as mesmas vieram em uma pasta única. Essa ação foi necessária para a adequação do formato de organização de dados aceito pelo *keras*. Com o uso do excel, o caminho das imagens foi segmentado de acordo com suas classes (perlita, martensita, etc). Em seguida, um *script* foi escrito para automatizar o processo de cópia para as pastas corretas. A função utilizada foi a *copy2*, da biblioteca *shutil*.

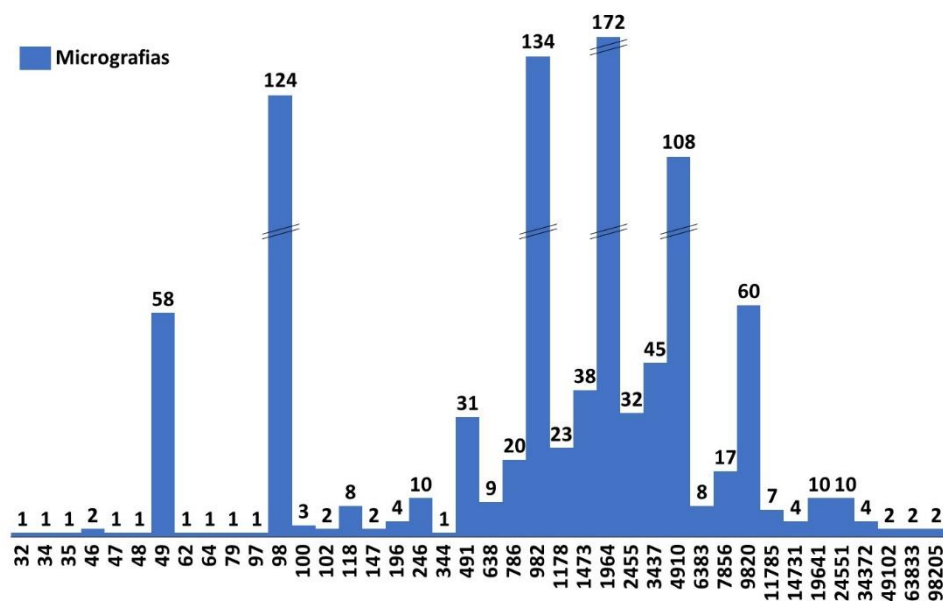
Com os arquivos nas respectivas pastas e os dados revisados, fora feito uma análise exploratória nos mesmos para melhor entender qual a composição das imagens contidas no banco de dados. Retomando as Figuras 18 e 19 nota-se que além das classes principais, cada fotomicrografia apresenta um valor de magnificação, tipo de detector, voltagem, entre outros. A Figura 20 apresenta a distribuição das imagens por suas classes e a Figura 21 a distribuição de imagens com relação a magnificação.

Figura 20 - Distribuição de micrografias por classe



Fonte: Elaborada pelo autor

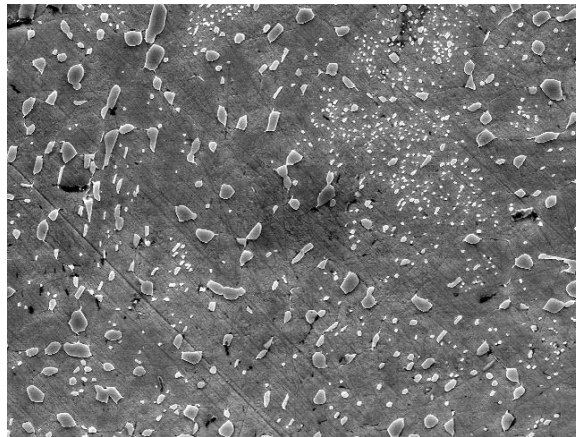
Figura 21 - Distribuição de micrografias por ampliação



Fonte: Elaborada pelo autor

O rodapé das fotomicrografias, contendo as informações descritas anteriormente, foi retirado com a ajuda de um *script*, a fim de deixar apenas a microestrutura registrada. O *script* funciona criando uma lista de imagens contidas nas pastas e, para cada imagem da lista, a sua entrada é atualizada com uma nova imagem de dimensões 645x485x1. A Figura 22 ilustra o resultado obtido.

Figura 22 - Fotomicrografia sem o rodapé



Fonte: *UltraHigh Carbon Steel Micrograph Database*, adaptada.

O último passo no preparo das imagens foi a separação das mesmas em conjunto de treinamento e teste. As imagens foram separadas numa razão 50:50 de modo aleatório. Devido ao baixo número de amostras de classes como martensita, optou-se por gerar dados de treinamento de forma sintética a fim de obter um conjunto de treinamento mais homogêneo. O método selecionado foi o de seccionar as imagens de treinamento em frações do original com o uso do pacote *image_slicer*. A Tabela 2 sumariza a organização do banco de dados.

Tabela 2 - Organização do banco de dados

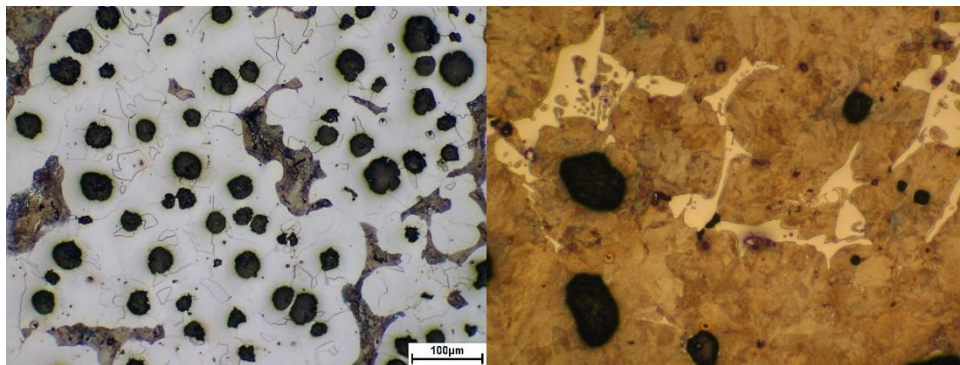
Constituinte primário	Quantidade	Teste	Treinamento	Sintéticas
Martensita	36	18	18	216
Network	212	106	106	954
Perlita	124	62	62	558
Perlita+Esferoidita	107	54	53	477
Perlita+Widmanstatten	27	14	13	208
Esferoidita	374	187	187	1683
Esferoidita+Widmanstatten	81	41	40	360

Fonte: Elaborada pelo autor

4.2.2 Banco de imagens de ferro fundido nodular - Welding

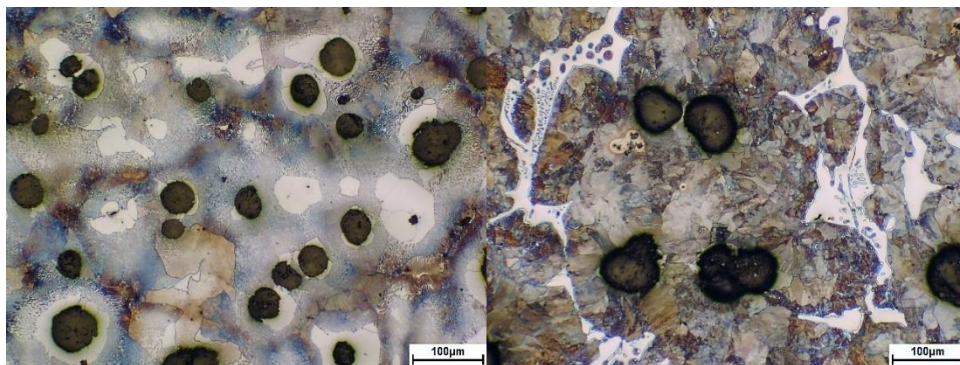
Esse banco de imagens foi construído a partir de fotomicrografias gentilmente cedidas pela empresa Welding. Ele é formado por trinta micrografias de ferro fundido nodular com dimensões 640x480x3 na ampliação 100x. As grafitas são de diferentes tamanhos, formatos e quantidade. As Figuras 23 e 24 apresentam alguns exemplos.

Figura 23 - Ferro fundido nodular



Fonte: Adaptada de Welding (2019)

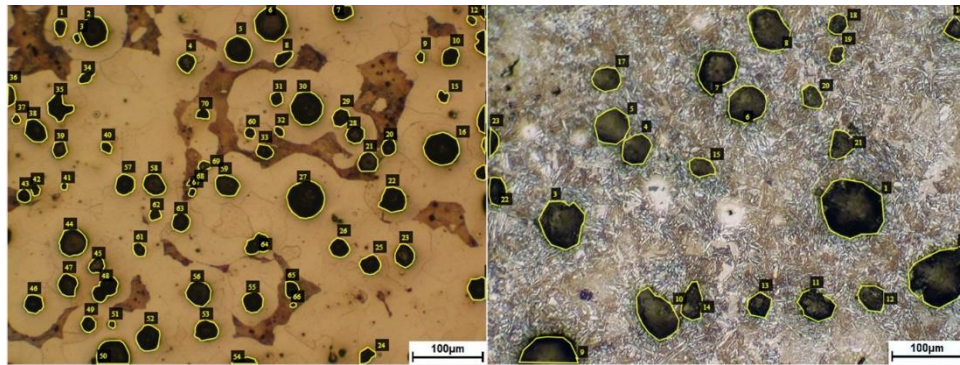
Figura 24 - Ferro fundido nodular



Fonte: Adaptada de Welding (2019)

O preparo das imagens para o banco de treinamento da *Mask R-CNN* envolve a anotação de cada grafita com o uso da ferramenta *VGG Image Annotation*. A Figura 25 apresenta exemplos de imagens anotadas.

Figura 25 - Regiões anotadas



Fonte: Elaborada pelo autor

Vinte e duas imagens foram separadas para treinamento e oito para validação.

4.2.3 Funções auxiliares

Foram definidas duas funções para possibilitar a manipulação das imagens utilizadas no treinamento e inferência do primeiro e segundo estudo de caso. O funcionamento delas é descrito a seguir:

- Define-se um vetor de categorias
- Para cada categoria no vetor de categorias:
 - É criada uma variável contendo o caminho para o diretório correto
 - Para cada pasta, é atribuído o nome da mesma a um rótulo de classe e:
 - É feita a iteração sobre uma amostra aleatória de tamanho N
 - As imagens, ao serem carregadas, são redimensionadas para 224x244x3
 - O valor dos pixels é normalizado para um intervalo entre $0 < x < 1$
 - A imagem e sua classificação são anexadas a uma lista

A lista obtida pela função de entrada de dados tem tamanho N e as imagens contidas nelas tem forma (224,224,3). Em seguida, as imagens são extraídas da lista e alocadas em um vetor de entradas X . O vetor é remodelado para $(N, 224, 224, 3)$. Os rótulos são convertidos para um vetor de dados categóricos e chamado de y . O mesmo é feito para os dados de teste e as informações são gravadas nos vetores X_{test} e y_{test} .

4.3 Modelos SVM

O algoritmo utilizado para a extração de características foi o HOG, pacote *scikit-learn*, e o *kernel* escolhido foi o gaussiano. Dois classificadores foram treinados: no primeiro modelo foram utilizadas as características obtidas diretamente do HOG como vetor de entrada e no segundo foi feita a redução de dimensionalidade das características através da aplicação da análise de componentes principais (PCA) (SHLENS, 2005).

4.3.1 SVM - HOG

A primeira etapa para o treinamento do modelo envolve a organização dos dados em treinamento e teste. Para isso, utilizou-se a função de apoio definida na Seção 4.2.3 com número de amostras igual a 800 sendo metade para a classe perlita e a outra metade para esferoidita. Em seguida, foi feita a extração de características de cada classe de estudo. O modelo foi treinado com 80% das imagens e validado nas demais 20%, sendo que o *kernel* escolhido foi o gaussiano.

4.3.2 SVM - HOG - PCA

Os passos necessários para o treinamento do segundo modelo são praticamente os mesmos da Seção 4.3.1. A diferença está na aplicação da técnica PCA após a extração das características. Essa técnica procura reduzir a dimensionalidade das características mantendo as informações mais relevantes.

4.4 Modelos de redes neurais convolucionais

A arquitetura escolhida foi a VGG16, uma rede neural convolucional composta por dezesseis camadas de convolução com filtros de 3x3, cinco camadas de *max pooling* de 2x2 e três camadas totalmente conectadas, envolvendo um total de 144 milhões de parâmetros (SIMONYAN, ZISSERMAN, 2015). Considerando a pequena quantidade de dados de treinamento e o esforço computacional necessário para treinar essa rede do zero, foi utilizada a estratégia de transferência de aprendizado. Foram desenvolvidos três classificadores com ordem crescente de número de classes, a fim de testar o impacto de diferentes números de

classes na acurácia do classificador. A Tabela 3 apresenta cada modelo e as classes nas quais fora treinado.

Tabela 3 - Modelos CNN

Modelo	Classes
03_VGG_16	Esferoidita, Martensita, Perlita
04_VGG_16	Esferoidita, Martensita, Perlita, Network
05_VGG_16	Esferoidita, Martensita, Network, Perlita, Perlita+Esferoidita

Fonte: Elaborada pelo autor

Os parâmetros utilizados foram: formato de entrada 224x224x3, pesos provenientes da *imagenet* e sem a camada do topo. As camadas foram congeladas (para preservar o valor dos pesos já treinados) e os dados de saída foram achatados em um longo vetor de características. A camada totalmente conectada foi definida com três *neurons*. A Tabela 4 apresenta o número de parâmetros de cada modelo.

Tabela 4 - Parâmetros dos modelos

Modelo	Parâmetros não treináveis	Parâmetros treináveis
03_VGG_16	14714688	75267
04_VGG_16	14714688	100356
05_VGG_16	14714688	125445

Fonte: Elaborada pelo autor

4.5 Aplicação: Mask R-CNN

O modelo utiliza a arquitetura *Resnet* 101 (HE, K.; ZHANG, X.; REN, S.; SUN, J.; 2015) como extrator de características e os pesos iniciais são provenientes de um modelo treinado no banco de dados MSCOCO com 20 *epochs* (número de vezes que todo o vetor de treinamento é utilizado para atualizar os pesos) e sem alterar os hiper parâmetros implementados.

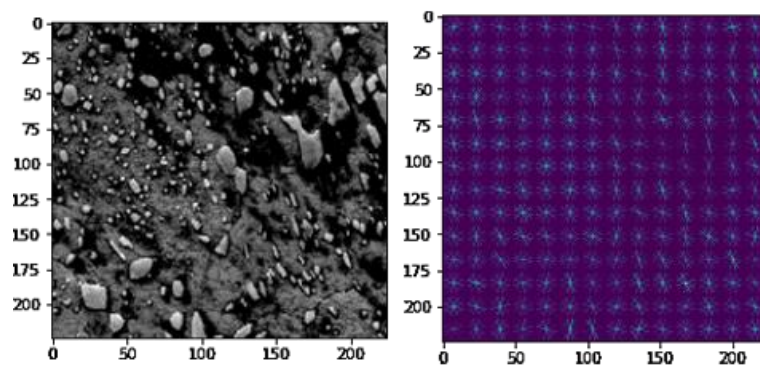
5 RESULTADOS E DISCUSSÕES

5.1 SVM

5.1.1 SVM - HOG

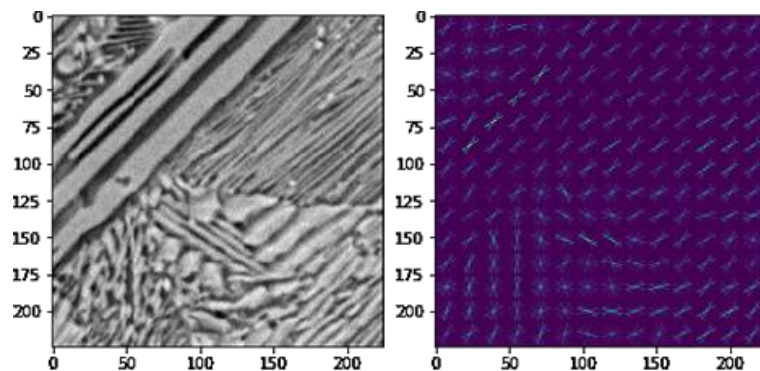
As Figuras 26 e 27 apresentam o resultado da extração de características, via HOG, em uma amostra de cada classe das imagens seccionadas do banco de treinamentos.

Figura 26- Esferoidita e representação HOG



Fonte: Elaborada pelo autor

Figura 27 - Perlita e representação HOG



Fonte: Elaborada pelo autor

O Quadro 1 e a Tabela 5 apresentam a matriz de confusão e o relatório de classificação do modelo.

Quadro 1 – Matriz de confusão modelo 01, imagens seccionadas

SVM_01		Previsto	
		Esferoidita	Perlita
Classe real	Esferoidita	80	6
	Perlita	24	50

Fonte: Elaborada pelo autor

Tabela 5 - Relatório de classificação modelo 01, imagens seccionadas

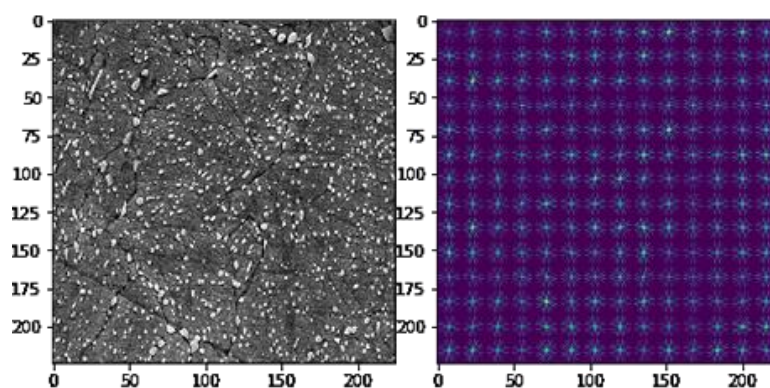
SVM_01	Precisão	Revocação	Medida F1	Nº de amostras
Esferoidita	0.77	0.93	0.84	86
Perlita	0.89	0.68	0.77	74
Acurácia			0.81	160
Média	0.83	0.80	0.81	160
Média ponderada	0.83	0.81	0.81	160

Fonte: Elaborada pelo autor

Vale destacar que os resultados do Quadro 1 e Tabela 5 são referentes as imagens seccionadas.

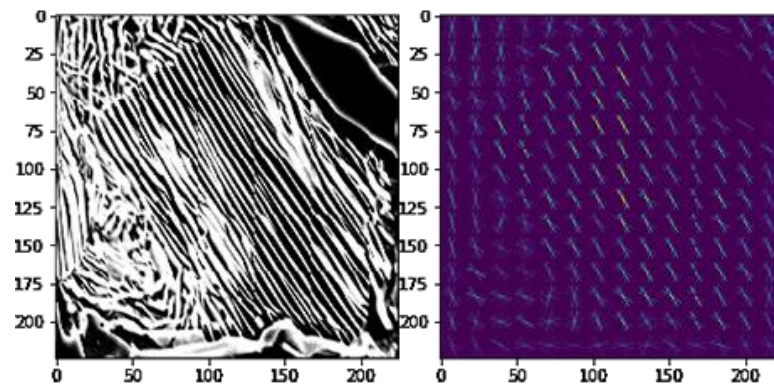
Em seguida, o modelo foi testado nas fotomicrografias, não seccionadas, previamente separadas para validação do classificador. As Figuras 28 e 29 apresentam uma amostra das imagens de validação bem como sua representação HOG.

Figura 28 – Classe esferoidita e representação HOG



Fonte: Elaborada pelo autor

Figura 29 - Classe perlita e representação HOG



Fonte: Elaborada pelo autor

O Quadro 2 e a Tabela 6 apresentam os resultados obtidos na inferência das micrografias.

Quadro 2 – Matriz de confusão modelo 01, banco de teste

SVM_01		Previsto	
		Esferoidita	Perlita
Classe real	Esferoidita	186	1
	Perlita	28	34

Fonte: Elaborada pelo autor

Tabela 6- Relatório de classificação modelo 01, banco de teste

SVM_01	Precisão	Revocação	Medida F1	Nº de amostras
Esferoidita	0.87	0.99	0.93	187
Perlita	0.97	0.55	0.70	62
Acurácia			0.88	249
Média	0.92	0.77	0.81	249
Média ponderada	0.89	0.88	0.87	249

Fonte: Elaborada pelo autor

As Tabelas 5 e 6 apresentam as métricas precisões, revocação, medida F1 e acurácia. A precisão avalia a capacidade do modelo em evitar falsos positivos. Com isso, uma precisão de 97% na classe perlita mostra que dos 35 elementos classificados como perlita, 97% foram

corretamente segmentados. A revocação é uma medida que avalia quão boa foi a segmentação entre as classes. Uma revocação de 99% para a esferoidita mostra que dos 187 elementos classificados como esferoidita, 99% foram corretamente selecionados. A acurácia corresponde aos pontos corretamente selecionados durante o teste. Um valor de 86% da acurácia indica que, das 249 imagens, 220(186 esferoiditas e 34 perlitas) foram corretamente classificadas. A medida F1 é a média harmônica entre precisão e revocação, *i.e.*, mostra o quão confiável é a acurácia obtida.

Comparando o modelo durante o treinamento e teste, nota-se uma maior precisão e revocação para a classe esferoidita, enquanto a classe perlita apresenta uma maior precisão e menor revocação. Essas diferenças podem indicar que as características extraídas descrevem melhor a classe esferoidita.

5.1.2 SVM – HOG - PCA

O Quadro 3 e Tabela 7 apresentam a matriz de confusão e o relatório de classificação das imagens seccionadas.

Quadro 3 – Matriz de confusão modelo 02, imagens seccionadas

SVM_02		Previsto	
		Esferoidita	Perlita
Classe real	Esferoidita	72	2
	Perlita	21	65

Fonte: Elaborada pelo autor

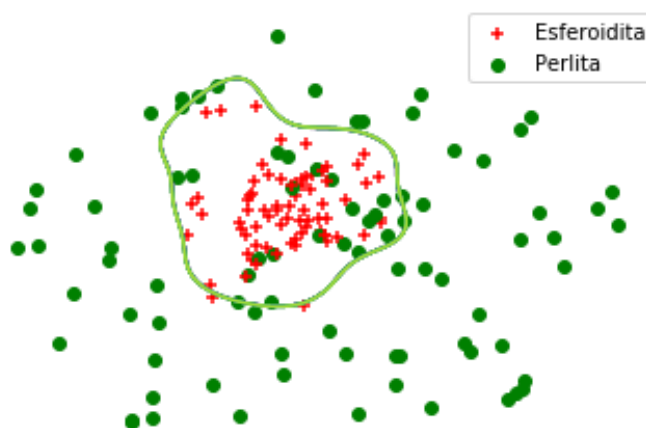
Tabela 7 - Relatório de classificação modelo 02, imagens seccionadas

SVM_02	Precisão	Revocação	Medida F1	Nº de amostras
Esferoidita	0.77	0.97	0.86	74
Perlita	0.97	0.76	0.85	86
Acurácia			0.86	160
Média	0.87	0.86	0.86	160
Média ponderada	0.88	0.86	0.86	160

Fonte: Elaborada pelo autor

A redução da dimensionalidade das características permite a visualização das classes assim como o a superfície de separação entre as mesmas em um gráfico de dispersão. A Figura 30 ilustra a superfície de decisão da SVM.

Figura 30 - Superfície de decisão SVM



Fonte: Elaborada pelo autor.

O Quadro 4 e Tabela 8 apresentam os resultados obtidos na inferência das micrografias.

Quadro 4 – Matriz de confusão do modelo 02, banco de teste

SVM_02		Previsto	
		Esferoidita	Perlita
Classe real	Esferoidita	185	2
	Perlita	30	32

Fonte: Elaborada pelo autor

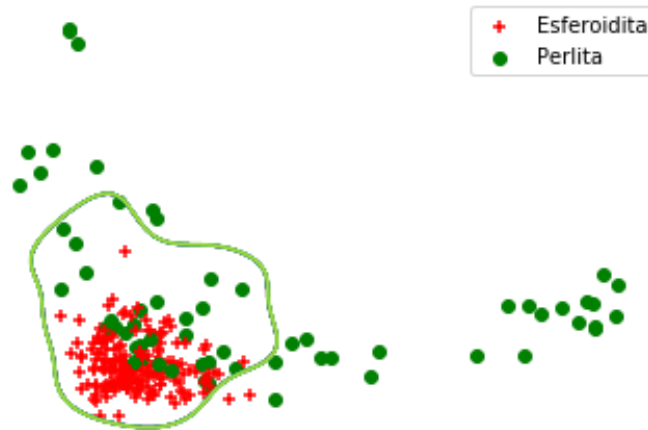
Tabela 8 - Relatório de classificação do modelo 02, banco de teste

SVM_02	Precisão	Revocação	Medida F1	Nº de amostras
Esferoidita	0.86	0.99	0.92	187
Perlita	0.94	0.52	0.67	62
Acurácia			0.87	249
Média	0.90	0.75	0.79	249
Média ponderada	0.88	0.87	0.86	249

Fonte: Elaborada pelo autor

A Figura 31 representa a superfície de decisão nas micrografias do banco de testes.

Figura 31 - Superfície de decisão



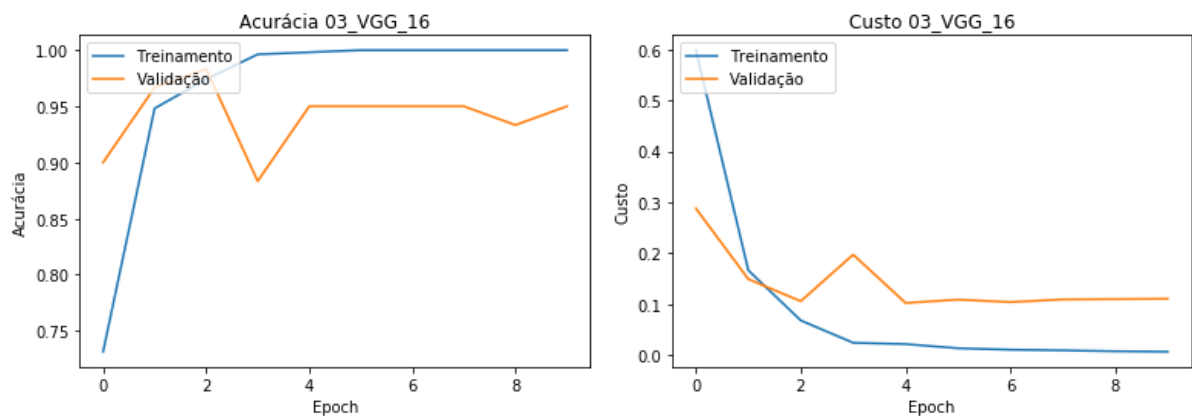
Fonte: Elaborada pelo autor

Analisando a Tabela 7 nota-se um valor equilibrado da medida F1 para ambas as classes, o que indica que a operação de redução de dimensionalidade foi relativamente eficaz na escolha de características relevantes para a inferência das fotomicrografias seccionadas. A Tabela 8 mostra um aumento da medida F1 para a classe esferoidita e uma queda para a perlita. Esse fato suporta a hipótese de que as características extraídas melhor descrevem a classe esferoidita. O melhor desempenho do modelo nas imagens seccionadas é um indicativo da possibilidade de utilizar o classificador para identificar regiões das fotomicrografias através de um algoritmo de janelas deslizantes. Por fim, ao analisar as Figuras 30 e 31, nota-se a grande diferença nas características entre os bancos de imagens (seccionados versus não seccionados). Os pontos da Figura 30 são melhor distribuídos entre os eixos do gráfico de dispersão, enquanto os da Figura 31 estão concentrados numa nuvem dentro da região de esferoiditas, apresentando o restante das perlitas como *outliers*. Isso provavelmente é resultado da não representatividade das imagens seccionadas em relação a fotomicrografia como um todo. Esse último ponto foi melhor estudado durante a aplicação das redes neurais convolucionais.

5.2 Redes Neurais Convolucionais

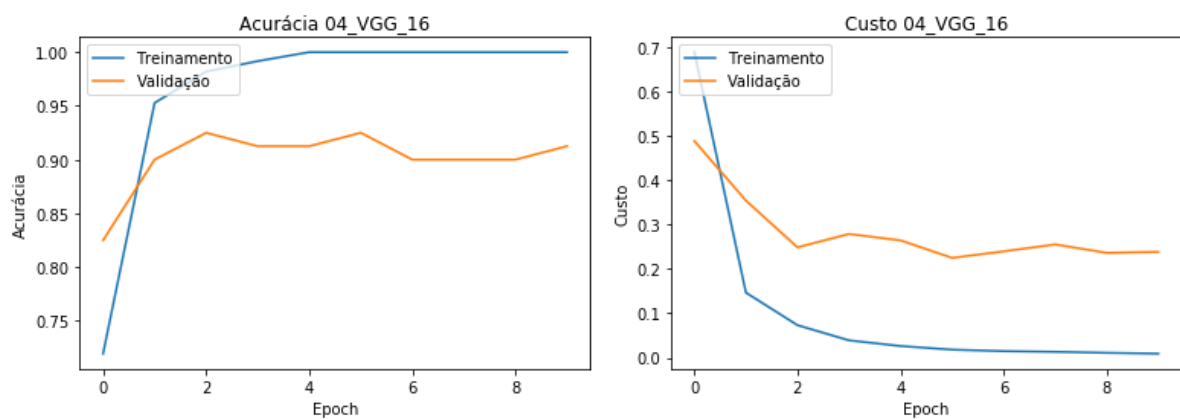
Os modelos foram treinados com 200 amostras por classe e por 10 *epochs*. Para manter a reprodutibilidade do trabalho, foi fixada uma semente para cada processo aleatório. As Figuras 32 a 34 apresentam os gráficos de acurácia e custo para cada modelo em uma das sessões de treinamento.

Figura 32 - Gráficos de acurácia e custo, modelo 03_VGG_16



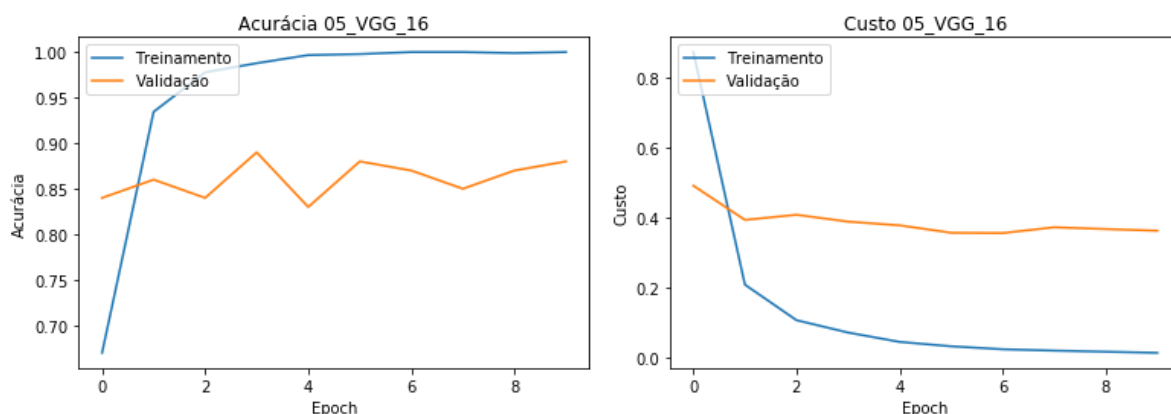
Fonte: Elaborada pelo autor

Figura 33 - Gráficos de acurácia e custo, modelo 04_VGG_16



Fonte: Elaborada pelo autor

Figura 34 - Gráficos de acurácia e custo, modelo 05_VGG_16



Fonte: Elaborada pelo autor

Nota-se que o platô foi atingido no décimo *epoch*, não havendo mais utilidade em continuar o treinamento a partir desse ponto, pois há o risco de adequação excessiva aos dados de treinamento o que impacta a capacidade do modelo em generalizar, *i.e.*, de classificar corretamente novas imagens. Também é notável a queda na acurácia e aumento do custo com o acréscimo de classes.

As matrizes de confusão e o relatório de classificação são apresentados nos Quadros 5 a 7 e nas Tabelas 9 a 11.

Quadro 5 – Matriz de confusão modelo 03_VGG_16

03_VGG_16		Previsto		
		Esferoidita	Martensita	Perlita
Classe real	Esferoidita	186	0	1
	Martensita	5	13	0
	Perlita	7	8	47

Fonte: Elaborada pelo autor

Quadro 6 - Matriz de confusão modelo 04_VGG_16

04_VGG_16		Previsto			
		Esferoidita	Martensita	Perlita	Network
Classe real	Esferoidita	172	0	0	15
	Martensita	0	12	0	6
	Perlita	10	9	43	0
	Network	10	3	0	93

Fonte: Elaborada pelo autor

Quadro 7 - Matriz de confusão modelo 05_VGG_16

05_VGG_16		Previsto				
		Esferoidita	Martensita	<i>Network</i>	Perlita	Perlita+Esferoidita
Classe real	Esferoidita	174	0	12	0	1
	Martensita	0	10	5	0	3
	<i>Network</i>	7	5	94	0	0
	Perlita	4	5	0	36	17
	Perlita+Esferoidita	6	5	7	1	35

Fonte - Elaborada pelo autor

Tabela 9 - Relatório de classificação 03_VGG_16

03_VGG_16	Precisão	Recall	Score F1	Nº de amostras
Esferoidita	0.94	0.99	0.97	187
Martensita	0.62	0.72	0.67	18
Perlita	0.98	0.76	0.85	62
Acurácia			0.92	267
Média	0.85	0.82	0.83	267
Média ponderada	0.93	0.92	0.92	267

Fonte: Elaborada pelo autor

Tabela 10 - Relatório de classificação 04_VGG_16

04_VGG_16	Precisão	Recall	Score F1	Nº de amostras
Esferoidita	0.90	0.92	0.91	187
Martensita	0.50	0.67	0.57	18
Perlita	1.00	0.69	0.82	62
<i>Network</i>	0.82	0.88	0.85	106
Acurácia			0.86	373
Média	0.80	0.79	0.79	373
Média ponderada	0.87	0.86	0.86	373

Fonte: Elaborada pelo autor

Tabela 11 - Relatório de classificação 05_VGG_16

05_VGG_16	Precisão	Recall	Score F1	Nº de amostras
Esferoidita	0.91	0.93	0.92	187
Martensita	0.40	0.56	0.47	18
<i>Network</i>	0.80	0.89	0.84	106
Perlita	0.97	0.58	0.64	62
Perlita+Esferoidita	0.62	0.65	0.64	54
Acurácia			0.82	427
Média	0.74	0.72	0.72	427
Média ponderada	0.83	0.82	0.82	427

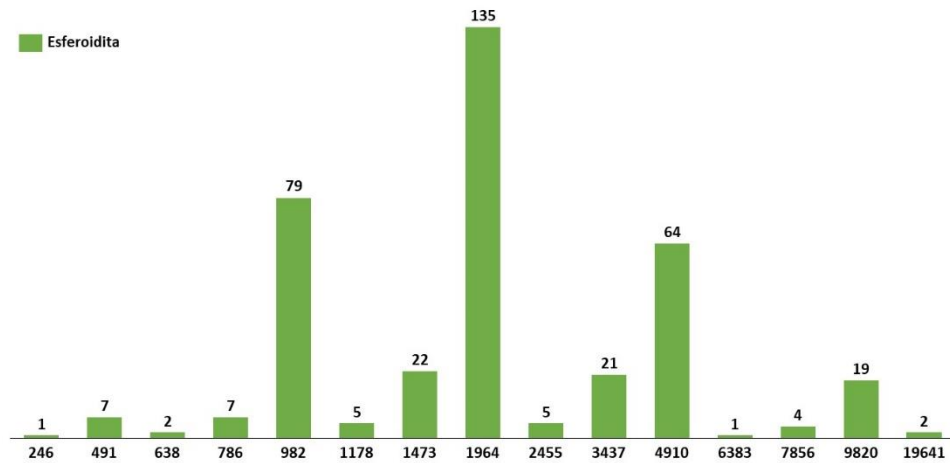
Fonte: Elaborada pelo autor

Os modelos apresentaram desempenho abaixo da média na tarefa de identificar a classe martensita, além disso houve queda de performance com o aumento de número de classes. Destaca-se a queda acentuada na medida F1, entre os modelos, com a adição da classe Perlita+Esferoidita. Foram feitas duas hipóteses para essa discrepância no desempenho dos modelos:

1. A distribuição irregular das imagens por magnificação impactou de maneira negativa o desempenho do classificador
2. As diferenças na textura e nos elementos das imagens apresentadas devido os diferentes níveis de ampliação impactaram de maneira negativa o desempenho do classificador

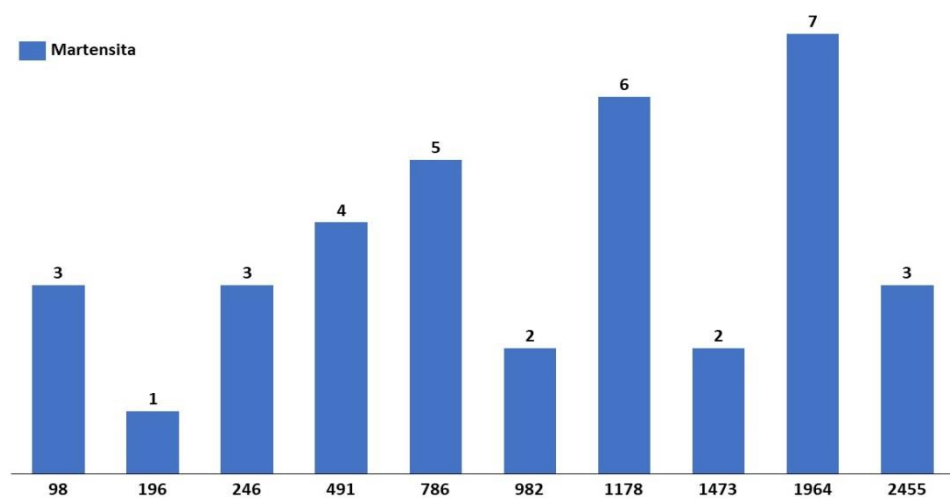
Para verificar as hipóteses, analisou-se a distribuição da magnificação por classe de imagem, apresentadas nas Figuras 35 a 39.

Figura 35- Magnificação por classe, esferoidita



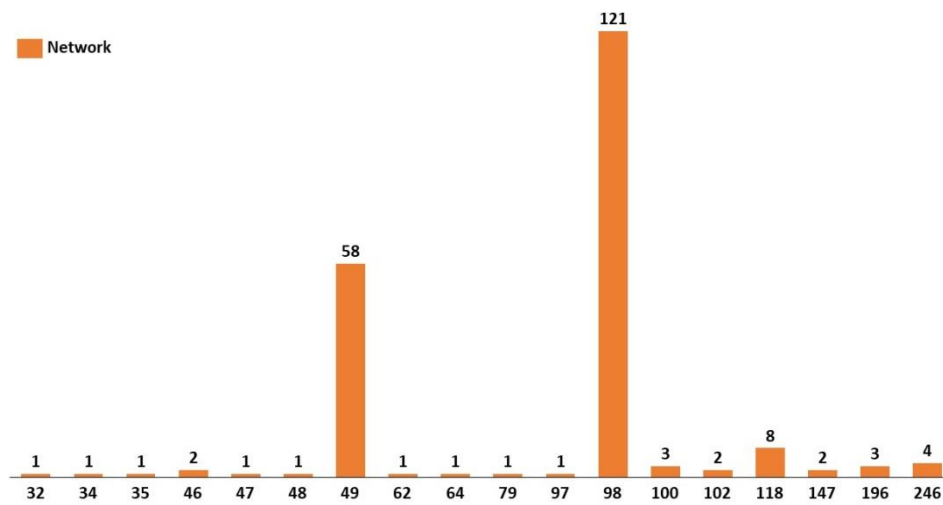
Fonte: Elaborada pelo autor

Figura 36 - Magnificação por classe, martensita



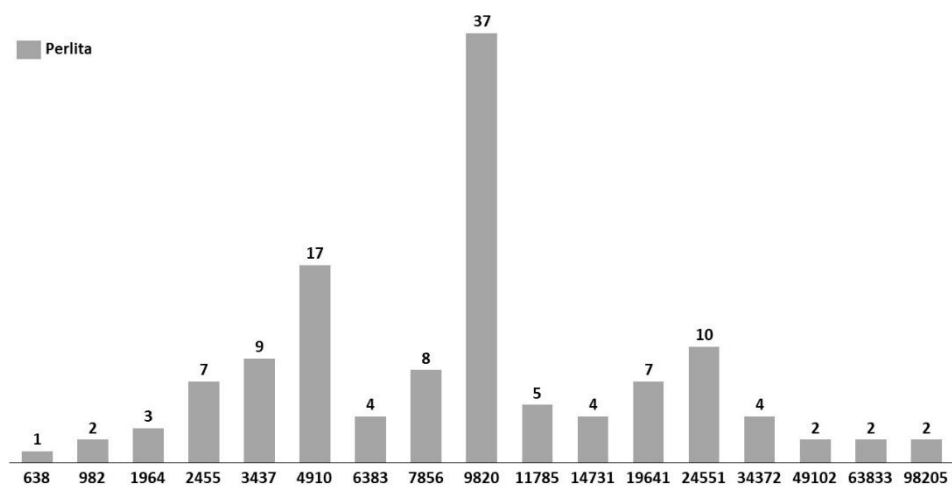
Fonte: Elaborada pelo autor

Figura 37 - Magnificação por classe, network



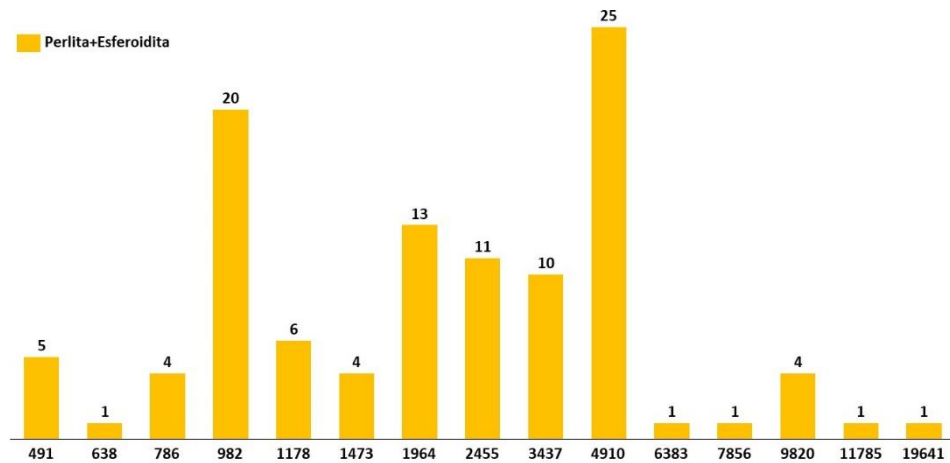
Fonte: Elaborada pelo autor

Figura 38 - Magnificação por classe, perlita



Fonte: Elaborada pelo autor

Figura 39 - Magnificação por classe, perlita+esferoidita



Fonte: Elaborada pelo autor

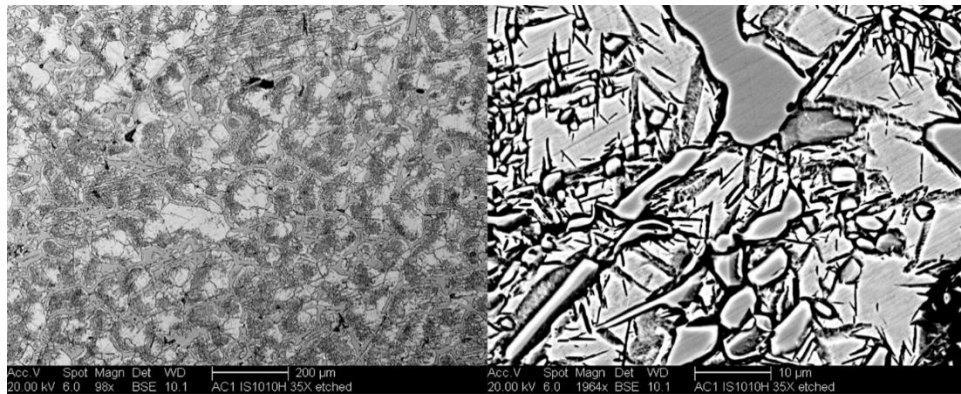
A distribuição é, de fato, irregular dentre as classes, no entanto não há evidências para atribuir uma correlação com o desempenho do classificador apenas com a análise gráfica, portanto uma análise exploratória das imagens foi conduzida. A seleção das fotomicrografias foi decidida da seguinte forma: com exceção da classe *network*, as demais foram selecionadas com base na maior ocorrência nas classes de maior e menor ampliação com a finalidade de apresentar os casos extremos do banco de dados, como mostram as Figuras 40 a 44.

Figura 40 - Fotomicrografias esferoidita



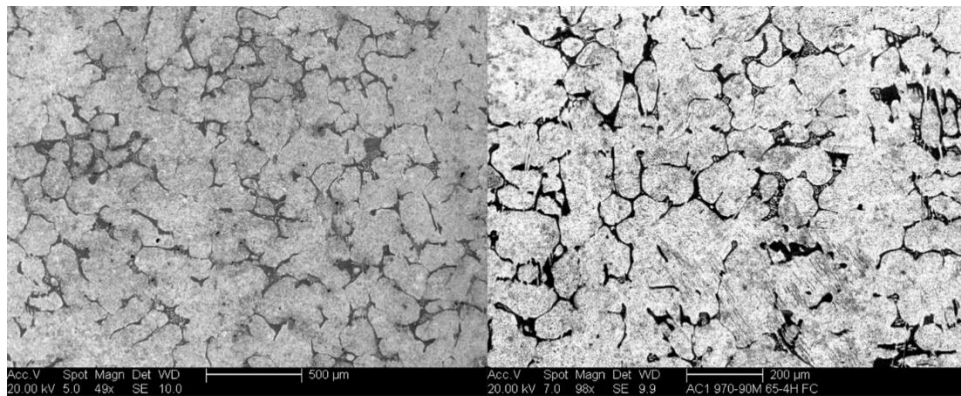
Fonte: *UltraHigh Carbon Steel Micrograph Database*

Figura 41- Fotomicrografias martensita



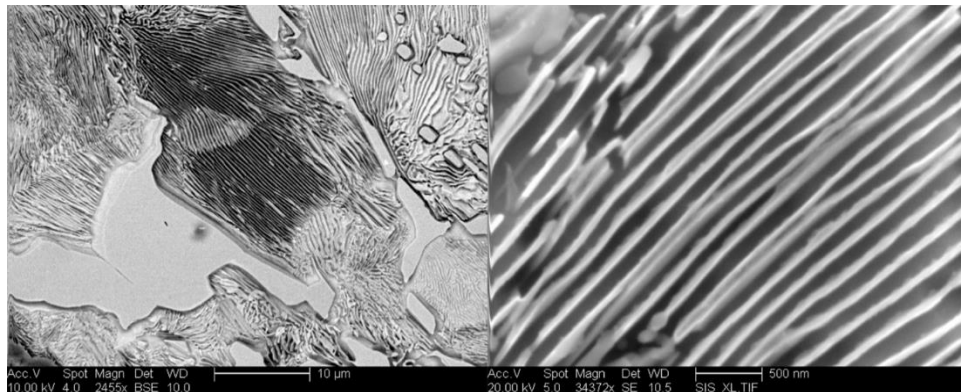
Fonte: *UltraHigh Carbon Steel Micrograph Database*

Figura 42 - Fotomicrografias network



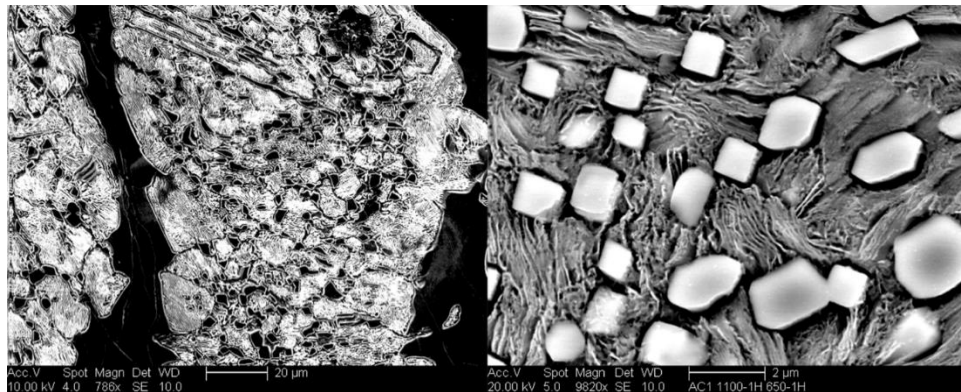
Fonte: *UltraHigh Carbon Steel Micrograph Database*

Figura 43 - Fotomicrografias perlita



Fonte: *UltraHigh Carbon Steel Micrograph Database*

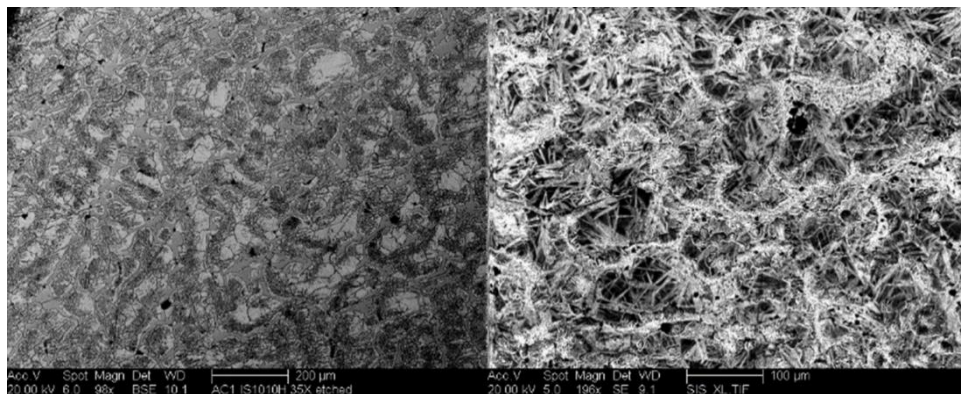
Figura 44 - Fotomicrografias perlita + esferoidita



Fonte: *UltraHigh Carbon Steel Micrograph Database*

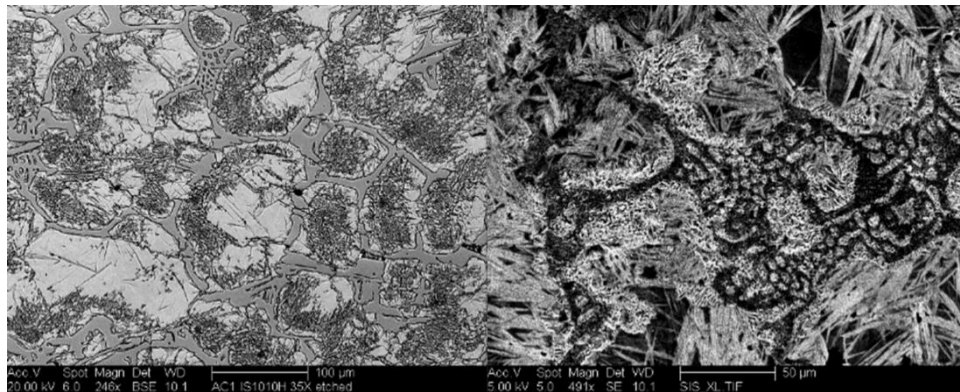
A classe martensita é a que apresenta as diferenças mais visíveis entre os extremos de magnificação. Nas demais classes, há elementos em comum nas imagens mesmo que em escalas diferentes. As demais categorias de magnificação da martensita foram verificadas a fim de averiguar se o mesmo ocorre para os outros níveis de amplificação, como mostram as Figuras 45 a 49.

Figura 45 - Fotomicrografias martensita



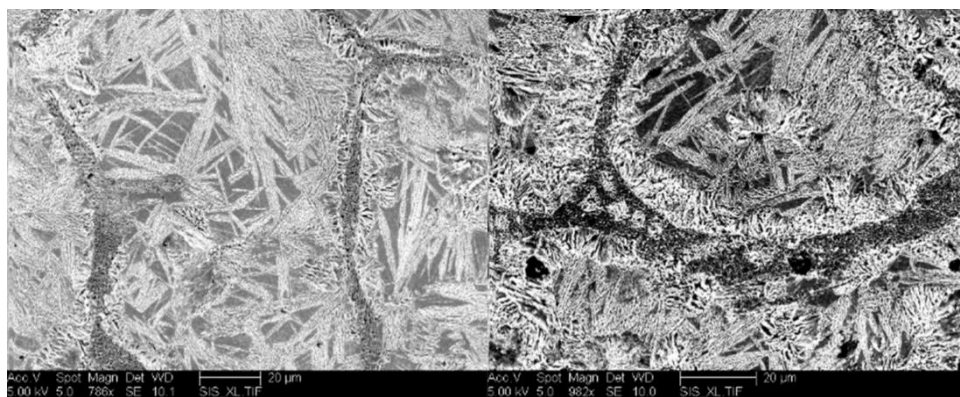
Fonte: *UltraHigh Carbon Steel Micrograph Database*

Figura 46 - Fotomicrografias martensita



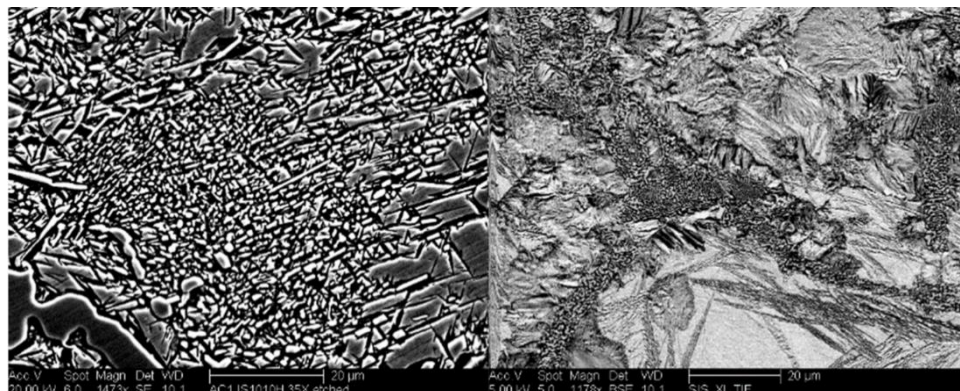
Fonte: *UltraHigh Carbon Steel Micrograph Database*

Figura 47 - Fotomicrografia martensita



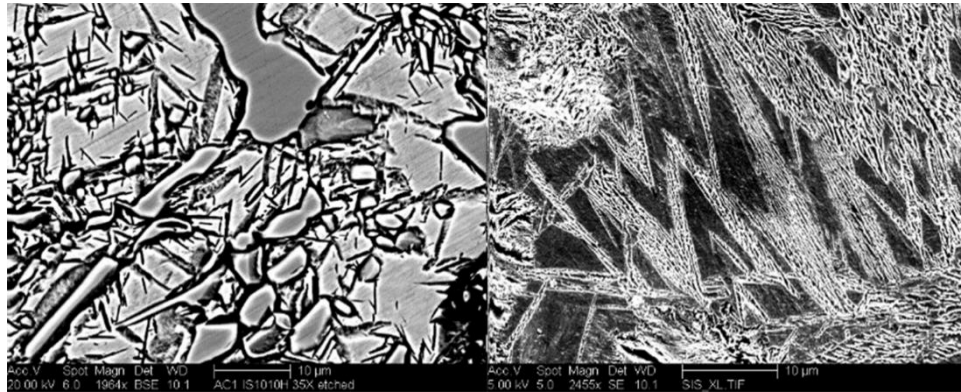
Fonte: *UltraHigh Carbon Steel Micrograph Database*

Figura 48 - Fotomicrografias martensita



Fonte: *UltraHigh Carbon Steel Micrograph Database*

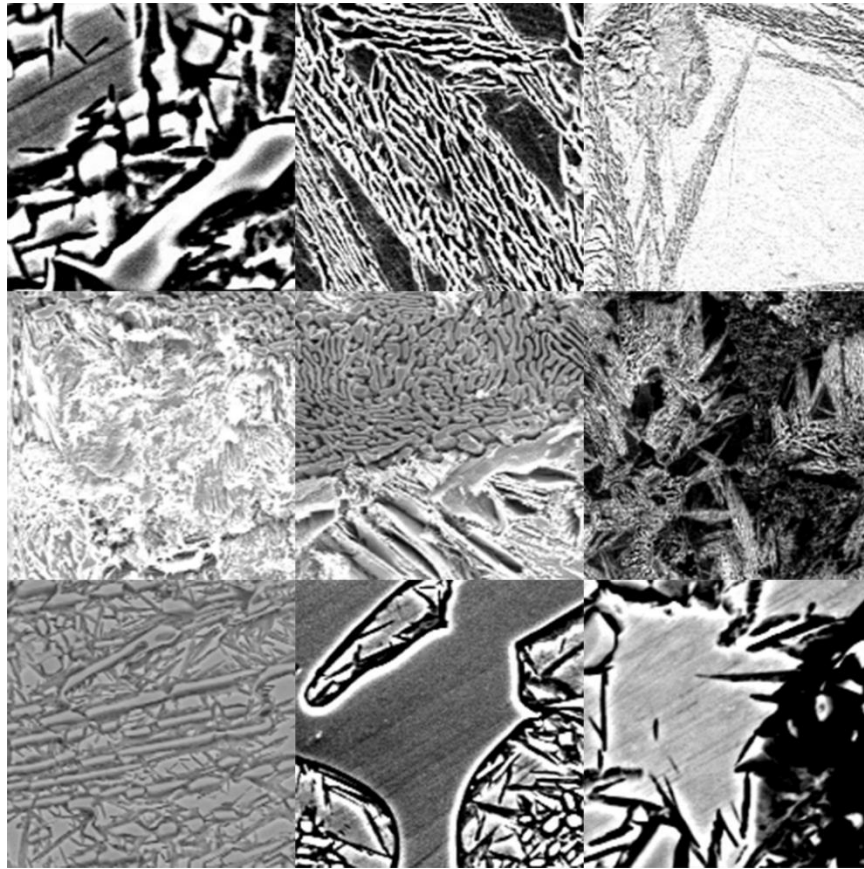
Figura 49 - Fotomicrografias martensita



Fonte: *UltraHigh Carbon Steel Micrograph Database*

Notam-se padrões no conjunto, no entanto há uma grande discrepância nos elementos dependendo da ampliação utilizada. Um outro ponto agravante foi o uso de secções das imagens para o treinamento o que aumentou a discrepância entre os elementos dessa classe. A Figura 50 apresenta uma seleção aleatória de algumas das secções utilizadas como elementos de treinamento. A ideia de utilizar secções como imagens de treinamento parte da premissa de que cada secção da imagem é representativa do todo. As imagens apresentadas nas Figuras 45 a 49 dão subsidio para crer que, a nível individual, as secções são relativamente representativas para uma mesma imagem. No entanto, é difícil acreditar que elas sejam capazes de representar a classe martensita como um todo devido as grandes diferenças visuais apresentadas em função da ampliação. Uma possível solução para esse problema seria aumentar o número de imagens representativas em cada classe e retreinar o classificador com imagens inteiras ao invés de secções.

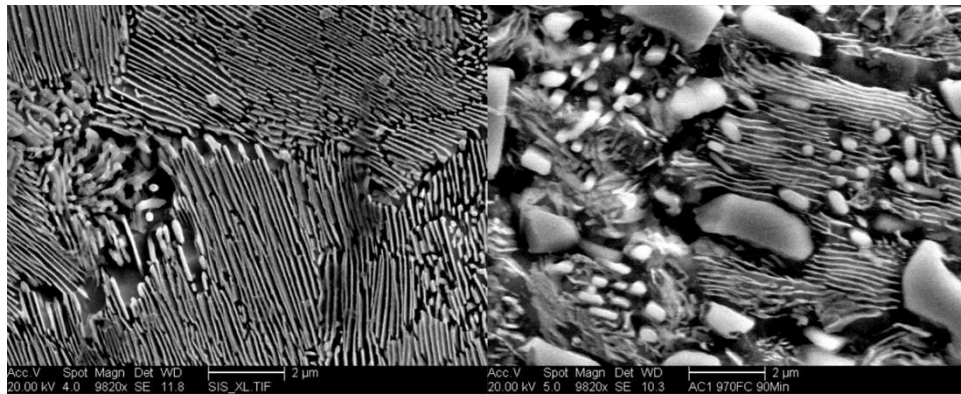
Figura 50 - Amostra de imagens do banco de treinamentos, martensita



Fonte: Elaborada pelo autor

Quanto a tarefa de identificar a classe perlita, nota-se uma queda de desempenho justamente após a introdução da classe perlita+esferoidita. Retomando a matriz de confusão do modelo 05_VGG_16, no Quadro 7, observa-se que 17 observações foram erroneamente classificadas como pertencentes a classe perlita+esferoidita, ao invés da classe perlita. Analisando as imagens do banco de dados tais como os apresentados na Figura 51 é natural esperar tal confusão visto que as fotomicrografias estão sendo analisadas como um todo sem haver distinção entre os elementos locais e há muita semelhança na textura das mesmas.

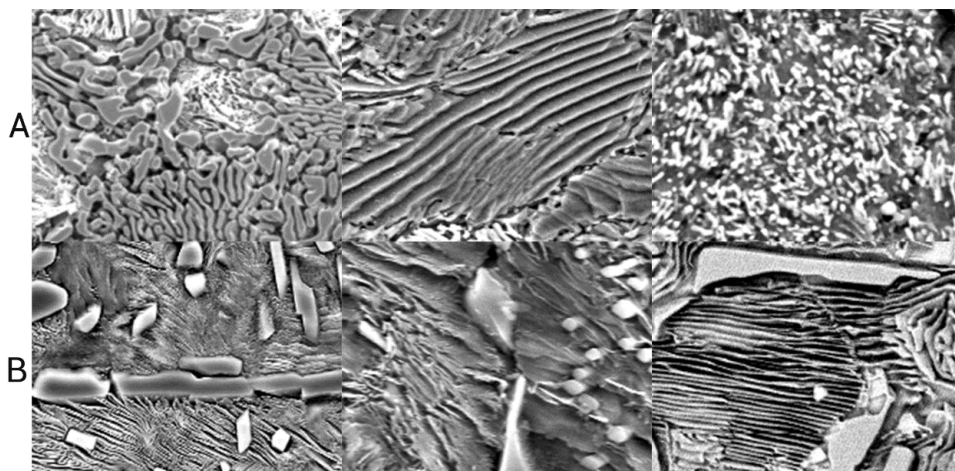
Figura 51 - Fotomicrografias perlita, perlita + esferoidita



Fonte *UltraHigh Carbon Steel Micrograph Database*

Essa semelhança é mais nítida na amostra dos dados de treinamentos apresentados na Figura 52.

Figura 52 - Amostra de dados de treinamento. A: perlita, B: perlita + esferoidita

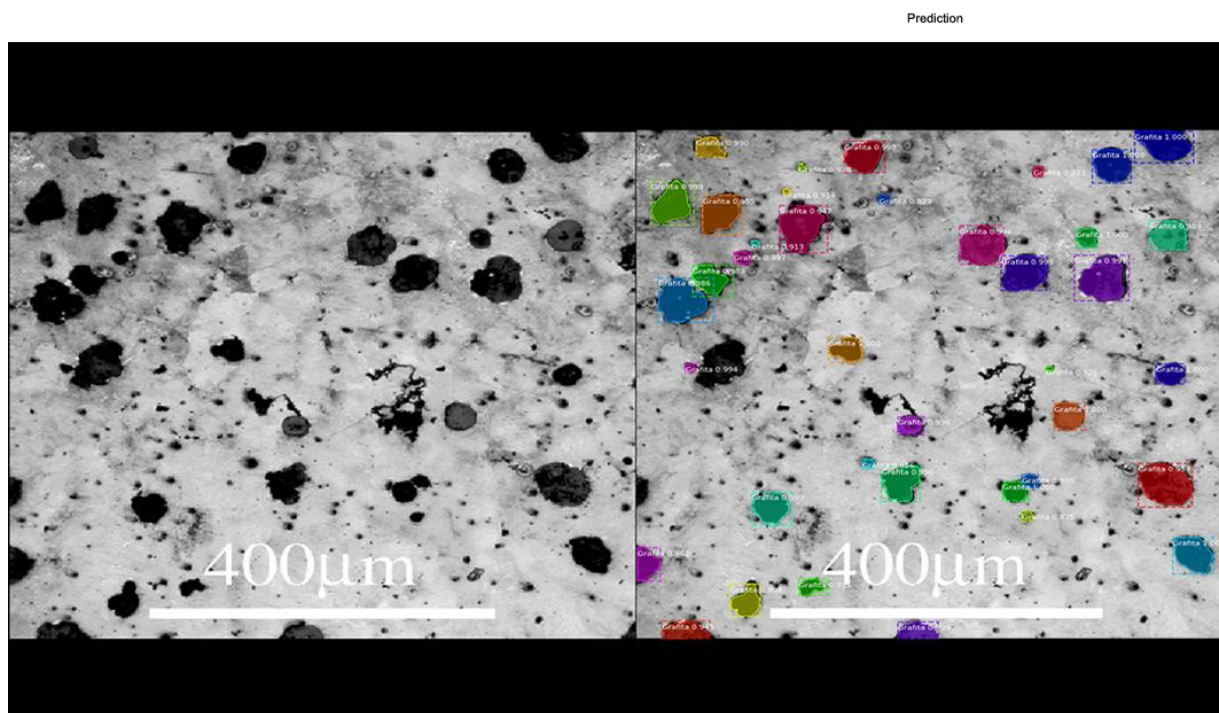


Fonte: Elaborada pelo autor

Para esse caso, onde há elementos que se confundem, uma abordagem mais adequada é o uso de classificadores que trabalham com regiões.

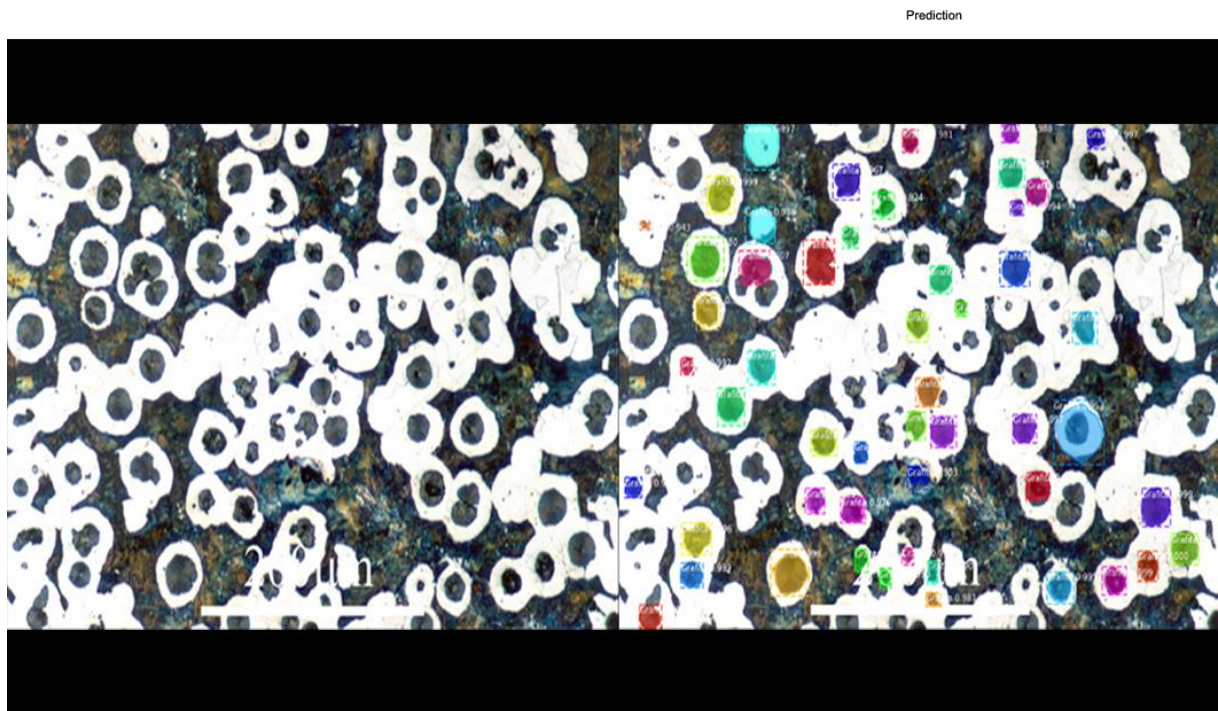
forma com a qual o indicador é calculado, pois, essa métrica leva em conta a posição exata informada durante o processo de anotações, processo este, sujeito a inconsistências. O modelo também foi testado em algumas amostras obtidas no repertório © DoITPoMS para avaliar sua capacidade de generalização. Os resultados são apresentados nas Figuras 57 a 59.

Figura 57 - Teste do modelo



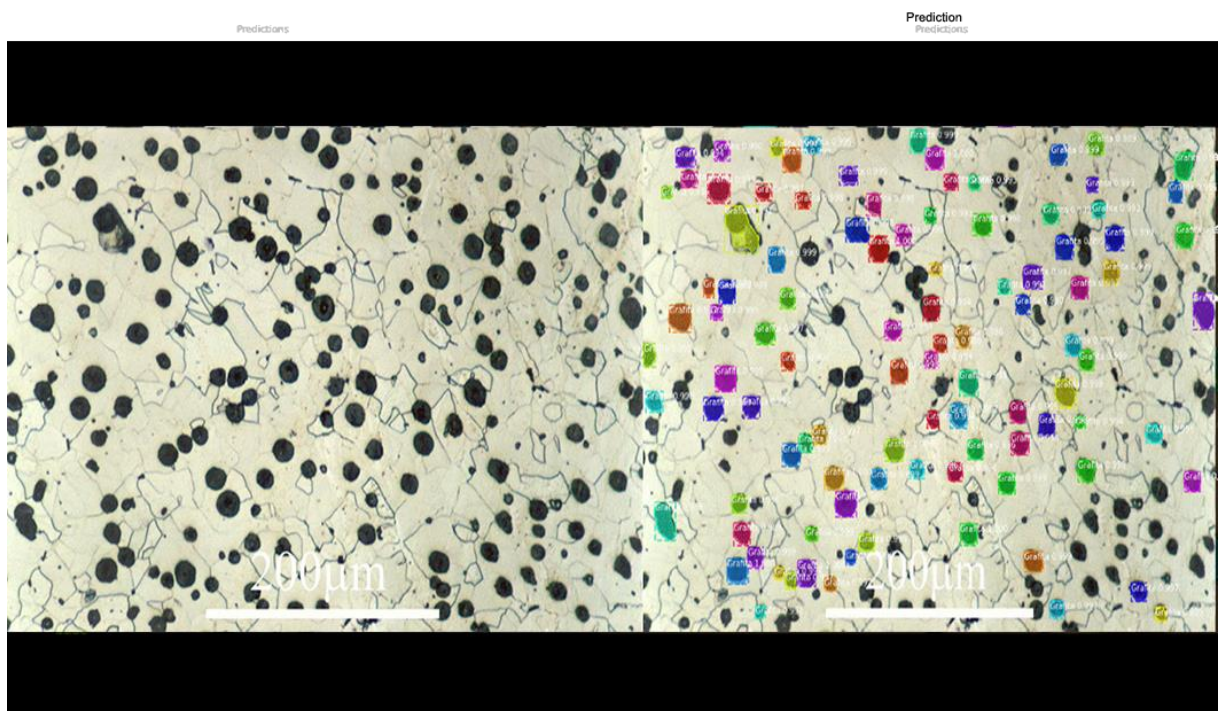
Fonte: © DoITPoMS Micrograph Library, University of Cambridge

Figura 58 - Teste do modelo



Fonte: © DoITPoMS Micrograph Library, University of Cambridge

Figura 59 - Teste do modelo



Fonte: © DoITPoMS Micrograph Library, University of Cambridge

A métrica de precisão média não foi reportada pois essas imagens foram retiradas diretamente do diretório e não foram anotadas. Os resultados são bem positivos ao considerar que as imagens analisadas nunca foram vistas pelo modelo, foram obtidas em condições diferentes daquelas do banco de treinamento, tem qualidade inferior e estão em preto e branco. Esses pontos são fortes indícios de que é possível obter um modelo mais generalizável ao adicionar imagens obtidas sob condições mais diversificadas.

6 CONCLUSÃO

Os modelos SVM são de fácil implementação e interpretação e ainda possuem aplicabilidade frente aos modelos de redes neurais em aplicações mais simples, tais como a classificação binária. É importante utilizar um método de extração de características adequado e refinar os hiper parâmetros levando em consideração o *trade off* precisão – revocação.

Os classificadores baseados em redes neurais apresentaram boa capacidade de classificação dada as limitações do banco de dados. A comparação entre os modelos fornece fortes indícios de que a performance de classificação é dependente da quantidade de classes e da representatividade das imagens utilizadas no banco de treinamento.

A aplicação *Mask RCNN* para a segmentação de instancias de fotomicrografias de ferro fundido nodular apresentou boa acurácia dentro do banco de validação, além de ter mostrado indícios da capacidade de generalização devido ao desempenho nas imagens do banco © DoITPoMS.

De uma maneira geral, os modelos de aprendizado de máquina demonstrados são uma excelente ferramenta no arsenal de um engenheiro principalmente com a crescente acessibilidade proporcionada por linguagens de programação, como python, contendo bibliotecas *open source* para a rápida implementação e teste dos modelos.

7 TRABALHOS FUTUROS

Do presente trabalho surgiram as seguintes sugestões para trabalhos futuros:

1. Classificação de grafitas por tipo e tamanho;
2. Mapeamento e classificação completa de fotomicrografias de ferro fundido (microconstituintes e matriz);
3. Identificação automática de inclusões indesejáveis em microestruturas.

REFERÊNCIAS

- ABADI, Martin et al. TensorFlow: Large-scale machine learning on heterogeneous systems, 2015. **Software available from tensorflow. org**, v. 1, n. 2, 2015.
- ABDULLA, W. Mask R-CNN for object detection and instance segmentation on Keras and TensorFlow, 2017. Disponível em: <https://github.com/matterport/Mask_RCNN>. Acesso em: 23 de Out. 2019.
- Alpaydin E. **Introduction to machine learning**, MIT Press:MA, 2004
- AMERICAN SOCIETY FOR METALS, **ASM Handbook Volume 9: Metallography and Microstructures**. ASM International, 2004
- AZIMI, S. M. et al. Advanced steel microstructural classification by deep learning methods. **Scientific Reports**, v. 8, n. 1, 1 dez. 2018.
- BISHOP C. **Pattern Recognition and Machine Learning**, Springer-Verlag New York, 2006
- BOSER, B. E.; GUYON, I. M.; VAPNIK, V. N. A training algorithm for optimal margin classifiers. In: PROCEEDINGS OF THE FIFTH ANNUAL WORKSHOP ON COMPUTATIONAL LEARNING THEORY - COLT '92, New York, New York, USA. **Anais...** New York, New York, USA: ACM Press, 1992.
- CALLISTER JR., W. D.; RETHWISCH, D. G. **Ciência Engenharia de Materiais - Uma Introdução**. 8ª Ed. Rio de Janeiro: Nova Guanabara, 2012.
- CHOLLET, F. Keras. **GitHub repository**, GitHub, 2015. Disponível em <<https://github.com/fchollet/keras>>. Acesso em: 23 de Out. 2019.
- COVER, T. M. Geometrical and Statistical Properties of Systems of Linear Inequalities with Applications in Pattern Recognition. **IEEE Transactions on Electronic Computers**, v. EC-14, n. 3, p. 326–334, 1965.
- CRISTIANINI N.; SHAW-TAYLOR J.; **An introduction to support vector machines and other kernel-based learning methods**, Cambridge University Press, 2000.
- DALAL, N.; TRIGGS, B. Histograms of oriented gradients for human detection. In: Proceedings - 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR 2005, San Diego, CA, USA, USA. **Anais...** San Diego, CA, USA, 2005.
- DEOCST, B.L.; HECHT, M.D.; FRANCIS, T. et al. UHCSDB: UltraHigh Carbon Steel Micrograph DataBase, **Integrating Materials and Manufacturing Innovation**, Volume 6, Issue 2, pp 197–205, jun. 2017
- DUTTA A.; ZISSERMAN A. The VIA Annotation Software for Images, Audio and Video. In **Proceedings** of the 27th ACM International Conference on Multimedia, Nice, France. ACM, New York, NY, USA, 2019
- GEELS, K. et al. **Metallographic and materialographic specimen preparation, light microscopy, image analysis and hardness testing**. ASTM International, 2007.

GOODFELLOW, I.; BENGIO, Y., COURVILLE, A. Deep Learning, MIT Press, 2016. Disponível em < <http://www.deeplearningbook.org>>. Acesso em: 23 de Out. 2019.

HAYKIN S. **Neural Networks A comprehensive foundation**, 3ª Ed. Prentice Hall, 2008

HE, K.; GKIOXARI, G.; DOLLÁR, P.; & GIRSHICK, R . Mask r-cnn. In: **Proceedings of the IEEE international conference on computer vision**. 2017. p. 2961-2969.

HE, K.; ZHANG, X.; REN, S.; SUN, J.; **Deep Residual Learning for Image Recognition. arXiv preprint arXiv:1512.03385**, 2015.

HUNTER, J.D. Matplotlib: A 2D graphics environment. **Computing in Science & Engineering**. V.9, n.3, p.90-95, 2007

KHAN, S. RAHMANI, H.; SHAH, S. A. A.; BENNAMOUN, M. A guide to convolutional neural networks for computer vision. **Synthesis Lectures on Computer Vision**, v. 8, n. 1, p. 1-207, 2018.

LE CUN, Y. A Theoretical Framework for Back-Propagation. In: **Anais...**1988.

LI, F.F.; JOHNSON, J.; YEUNG, S.; **Stanford University CS231n: Convolutional Neural Networks for Visual Recognition**. Stanford, CA: Stanford Vision Lab, Stanford University, 2018 Disponível em: <<http://cs231n.stanford.edu/>>. Acesso em: 23 de Out. 2019.

LORENA, A. C.; DE CARVALHO, A. C. P. L. F. Uma Introdução às Support Vector Machines. **Revista de Informática Teórica e Aplicada**, v. 14, n. 2, p. 43–67, 20 dez. 2007.

LOWE, D. G. Distinctive Image Features from Scale-Invariant Keypoints. **International Journal of Computer Vision**, 2004.

PEDRINI, H.; SCHWARTZ, W. R.; **Análise De Imagens Digitais: Princípios, Algoritmos E Aplicações**. Cengage Learning, 2007.

PEDREGOSA, F.; VAROQUAUX, G. ;GRAMFORT, A. ; MICHEL, V.; THIRION, B.;GRISEL, O. ;BLONDEL, M.;PRETTENHOFER, P.;WEISS, R.;DUBOURG, V.; VANDERPLAS, J.;PASSOS, A.; COURNAPEAU, D.;BRUCHER, M.;PERROT, M.;DUCHESNAY, E. Scikit-learn: Machine Learning in {P}ython. **Journal of Machine Learning Research**. v.12, p. 2825-2830, 2011,

RAFAEL C. GONZALEZ, R. E. W. **Processamento digital de imagens**, 3ª Ed. Pearson, 2010.

R Core Team. R: A Language and Environment for Statistical Computing. Vienna, Austria, 2016. Disponível em <<https://www.R-project.org/>>. Acesso em: 23 de Out. 2019.

REN, S.; HE, K.; GIRSHICK, R.; SUN, J. Faster r-cnn: Towards real-time object detection with region proposal networks. In: **Advances in neural information processing systems**. p. 91-99, 2015

ROSSUM, G. VAN. Python tutorial, technical report CS-R9526. Amsterdam: Centrum voor Wiskunde en Informatica (CWI), 1995.

RYCHETSKY, M. **Algorithms and architectures for machine learning based on regularized neural networks and support vector approaches**. Shaker, 2001.

SEGAL, D. **Materials for the 21st century**. Oxford University Press, 2017.

SHLENS, J. A tutorial on principal component analysis. **arXiv preprint arXiv:1404.1100**, 2014.

SIMONYAN, K.; ZISSERMAN, A. Very Deep Convolutional Networks for Large-Scale Image Recognition. In: **Anais...** 2014.

VENKATESAN, R.; LI, B.. **Convolutional Neural Networks in Visual Computing: A Concise Guide**. CRC Press, 2017.

WICKHAM H. **ggplot2: Elegant Graphics for Data Analysis**. Springer-Verlag, New York, 2016.

WOJNAR, L. **Image analysis : applications in materials engineering**. CRC Press, 1999.