UNIVERSIDADE FEDERAL DO ESPÍRITO SANTO
CENTRO TECNOLÓGICO
DEPARTAMENTO DE ENGENHARIA ELÉTRICA
PROJETO DE GRADUAÇÃO

Gabriel Andrade Nunes de Moraes

REDES NEURAIS CONVOLUCIONAIS APLICADAS À DETERMINAÇÃO DE GRAU DE ATIPIA NUCLEAR

Vitória-ES

Dezembro/2018

Gabriel Andrade Nunes de Moraes

REDES NEURAIS CONVOLUCIONAIS APLICADAS À DETERMINAÇÃO DE GRAU DE ATIPIA NUCLEAR

Parte manuscrita do Projeto de Graduação do aluno Gabriel Andrade Nunes de Moraes, apresentado ao Departamento de Engenharia Elétrica do Centro Tecnológico da Universidade Federal do Espírito Santo, como requisito parcial para obtenção do grau de Engenheiro Eletricista.

Vitória-ES

Dezembro/2018

Gabriel Andrade Nunes de Moraes

REDES NEURAIS CONVOLUCIONAIS APLICADAS À DETERMINAÇÃO DE GRAU DE ATIPIA NUCLEAR

Parte manuscrita do Projeto de Graduação do aluno Gabriel Andrade Nunes de Moraes, apresentado ao Departamento de Engenharia Elétrica do Centro Tecnológico da Universidade Federal do Espírito Santo, como requisito parcial para obtenção do grau de Engenheiro Eletricista.

Aprovado em 7 de Dezembro de 2018.

COMISSÃO EXAMINADORA:

Prof. Dr. Jorge Leonid Aching Samatelo
Universidade Federal do Espírito Santo
Orientador

Prof. Dr. André Ferreira
Universidade Federal do Espírito Santo
Examinador

Msc. Fabio Ricardo Oliveira Bento
Instituto Federal do Espírito Santo
Examinador

Vitória-ES

Dezembro/2018

A todos que me ajudaram a chegar aqui.

AGRADECIMENTOS

Aos meus pais, amigos e família, por todo o apoio nos meus momentos de maior necessidade, e mesmo nos momentos tranquilos.

A meu orientador pela paciência e direcionamento certeiro, e aos membros do Laboratório VIROS pela ajuda sempre bem-vinda nas mais variadas etapas do trabalho.

À banca examinadora pela aceitação do convite e pelo tempo investido para leitura e avaliação desse trabalho.

RESUMO

O câncer de mama é a variação de câncer que possui a segunda maior incidência em mulheres no Brasil e no mundo, estando apenas atrás do câncer de pele não melanoma. É caracterizado pelo aparecimento de tumor(es) maligno(s) no tecido da mama. A atribuição de grau a um câncer tradicionalmente é feita a partir da análise direta de um histologista. No entanto, é considerada uma tarefa trabalhosa e delicada, pois além de exigir a análise de centenas de células por imagem, as características visuais que precisam ser identificadas são muitas vezes subjetivas, O presente projeto de graduação faz uso de uma rede convolucional para tentar solucionar o problema de classificação de imagens de células de tumores de câncer de mama em grau 1, 2 ou 3 de atipia nuclear. A tarefa a solucionar foi obtida do desafio Mitos-Atypia-14, que provê um banco de dados para realizar o treino das redes e para validar a solução. Como forma de solução adotada foram utilizadas e comparadas duas arquiteturas de rede neural, além de uma aprimoração de resultados baseada na junção de ambas. A partir da abordagem adotada foi obtida acurácia média de $73,44\%$ no conjunto de testes utilizado, aproximando-se de resultados conhecidos da literatura na mesma tarefa.

Palavras-chave: *Deep-learning*; Redes neurais convolucionais; *Transfer-learning*; *Ensemble Learning*.

ABSTRACT

Breast cancer is the cancer variation with the second highest incidence rate among women in Brazil and around the world, being surpassed only by non-melanoma skin cancer. It is characterized by the appearancce of malignant tumor(s) in the breast tissue. Attibution of a grade for a cancer is usually through the visual analysis of a histologist. However, that is considered a laborious and delicate task, because in addition to demanding analysis of hundreds of cells per image, the visual characteristics that require identification are often subjective ones. The graduation project presented here uses a convolutional neural network to try to perform the task of classifying images of breast cancer tumor cells as grade 1, 2 or 3 in regard to their nuclear atypia. Such task was obtained from the *Mitos-Atypia-14* challenge, which provides a dataset on which to perform trainment of the networks and validation of the solution. As a solution, two neural network architectures have been adopted and compared, as well as an enhancement of the results based on putting both of them together. That approach to the problem resulted in average accuracy of $73,44\%$ over the test set, which is close to known results in the literature of the area.

Keywords: Deep-learning; Convolutional Neural Networks, Transfer-learning; Ensemble Learning.

LISTA DE FIGURAS

Figura 1 – Imagens de tecidos com diferentes graus de atipia: 1 (esquerda), 2 (centro) e 3 (direita). 13
Figura 2 – Representação genérica da rede VGG-16. 20
Figura 3 – Bloco residual da Resnet. 21
Figura 4 – Representação da formação do conjunto de treino para as redes neurais 24
Figura 5 – Exemplo do método de *ensemble* aplicado à classificação dos *patches*. . 26
Figura 6 – Procedimento de classificação de uma imagem de teste 27
Figura 7 – Imagem presente no dataset de treino fornecido pelo concurso MITOS & ATYPIA 14 29
Figura 8 – Comparação entre imagem de classe 2 classificada pela rede como classe 3 (**b**) com casos de classificação correta de classe 2 (**a**) e 3 (**c**). Foi utilizada a inferência da ResNet-50 no exemplo. 33

LISTA DE TABELAS

Tabela 1 – Hiperparâmetros para o treinamento das redes CNN usadas. 31
Tabela 2 – Acurácia da classificação da ResNet-50 em função do *stride* das imagens de teste. 33
Tabela 3 – Matriz de confusão resultante do teste final da ResNet-50 com *stride* de 224 pixels nas imagens. 33
Tabela 4 – Matriz de confusão resultante do teste final da ResNet-50 com stride de 112 pixels nas imagens. 33
Tabela 5 – Matriz de confusão resultante do teste final da ResNet-50 com stride de 50 pixels nas imagens 34
Tabela 6 – Acurácia da classificação da VGG16 em função do *stride* das imagens de teste. 34
Tabela 7 – Matriz de confusão resultante do teste final da VGG16 com *stride* de 224 pixels nas imagens. 34
Tabela 8 – Matriz de confusão resultante do teste final da VGG16 com *stride* de 112 pixels nas imagens. 34
Tabela 9 – Matriz de confusão resultante do teste final da VGG16 com *stride* de 50 pixels nas imagens. 35
Tabela 10 – Acurácia da classificação do *ensemble* das redes em função do *stride* das imagens de teste. 35
Tabela 11 – Matriz de confusão resultante do *ensemble* das redes com *stride* de 224 pixels nas imagens. 35
Tabela 12 – Matriz de confusão resultante do *ensemble* das redes com *stride* de 112 pixels nas imagens. 35
Tabela 13 – Matriz de confusão resultante do *ensemble* das redes com *stride* de 50 pixels nas imagens. 36
Tabela 14 – Acurácia da classificação de cada rede e do *ensemble* em função do *stride* das imagens de teste, na execução no *Google Colaboratory*. . . . 36

LISTA DE ABREVIATURAS E SIGLAS

API	*Application Program Interface*
CNN	*Convolutional Neural Network*
DL	*Deep Learning*
GPU	*Graphics Processing Unit*
ML	*Machine Learning*
SGD	*Stochastic Gradient Descent*
UFES	Universidade Federal do Espírito Santo

SUMÁRIO

1 INTRODUÇÃO . . . 12
1.1 Apresentação e Objeto de Pesquisa . . . 12
1.2 Justificativa . . . 14
1.3 Objetivos . . . 15
1.4 Estrutura do Texto . . . 15

2 EMBASAMENTO TEÓRICO . . . 17
2.1 Introdução . . . 17
2.2 CNN . . . 17
2.2.1 Componentes Básicos de uma CNN . . . 17
2.2.2 Treinamento de uma CNN . . . 18
2.3 Arquitetura de redes CNN . . . 18
2.3.1 VGG . . . 19
2.3.2 ResNet . . . 20
2.4 *Transfer Learning* . . . 21
2.5 *Ensemble Learning* . . . 22
2.6 Resumo . . . 22

3 METODOLOGIA . . . 23
3.1 Introdução . . . 23
3.2 Preditor de Classe . . . 23
3.2.1 Arquiteturas de Redes CNN usadas . . . 24
3.3 *Ensamble* de Preditores . . . 25
3.4 Resumo . . . 26

4 RESULTADOS . . . 28
4.1 Introdução . . . 28
4.2 Recursos Materiais . . . 28
4.3 Recursos Computacionais . . . 29
4.4 Detalhes de implementação . . . 30
4.4.1 Tratamento da base de dados . . . 30
4.4.2 Implementação das redes . . . 30
4.4.3 Etapa de treinamento . . . 30
4.4.4 Descrição dos valores dos hiperparâmetros da rede . . . 31
4.5 Experimentos . . . 31
4.5.1 Métricas . . . 31

4.5.2 Experimento Efetuados 32
4.5.3 Comparação com outros trabalhos 35
4.5.4 Resultados Particulares 36
4.6 Resumo 37

5 CONCLUSÕES E PROJETOS FUTUROS 38
5.1 Conclusões 38
5.2 Temas a serem pesquisados 38

REFERÊNCIAS 40

1 INTRODUÇÃO

1.1 Apresentação e Objeto de Pesquisa

O câncer de mama é a variação de câncer que possui a segunda maior incidência em mulheres no Brasil e no mundo, estando apenas atrás do câncer de pele não melanoma. É caracterizado pelo aparecimento de tumor(es) maligno(s) no tecido da mama.

Uma das etapas do diagnóstico dessa doença, após surgimento de suspeita de câncer em mamografia ou ressonância magnética, é a biópsia de tecido coletado na mama. Esse procedimento, também chamado de análise histológica, consiste na observação sob microscópio de finas camadas retiradas do possível tumor, sendo possível então estabelecer o diagnóstico e, em caso positivo, também é obtido um prognóstico mais preciso para a doença. O profissional responsável pelas análises histológicas é denominado histologista.

Atualmente, a partir de uma análise histológica, é possível atribuir uma classificação numérica a quadros de câncer de mama. Essa classificação, chamada de grau, representa o potencial agressivo de um tumor, sendo assim muito importante para a orientação do tratamento.

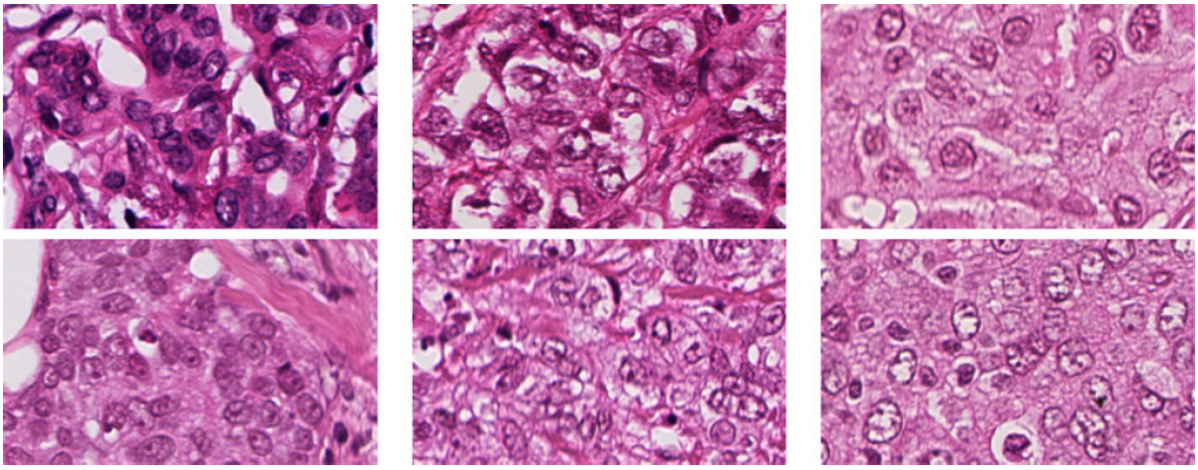
Existem mais de uma maneira de quantificar o grau de um câncer, mas o sistema recomendado pela Organização Mundial da Saúde - OMS é conhecido como sistema de classificação *Nottingham*(ELLIS et al., 2003), que leva em consideração três fatores visíveis em análises histológicas: quantidade de formações tubulares, nível de atividade mitótica e atipia (ou pleomorfismo) nuclear.

A atribuição de grau a um câncer tradicionalmente é feita a partir da análise direta de um histologista. No entanto, é considerada uma tarefa trabalhosa e delicada, pois além de exigir a análise de centenas de células por imagem, as características visuais que precisam ser identificadas são muitas vezes subjetivas (KHAN, 2014). A Figura 1 traz exemplos de tecidos com diferentes graus de atipia.

Tal grau de complexidade evidencia vantagens que surgem ao se automatizar esse tipo de procedimento. Entre elas uma maior objetividade e reprodutibilidade dos resultados, ao eliminar-se a variação existente no julgamento de um profissional para outro, e maior velocidade quando é necessário trabalhar sobre uma grande quantidade de dados.

Neste trabalho, pretende-se abordar especificamente a determinação de grau de atipia nuclear. Em termos computacionais, esse caso é visto como um problema de classificação.

Figura 1 – Imagens de tecidos com diferentes graus de atipia: 1 (esquerda), 2 (centro) e 3 (direita).



Fonte: (KHAN; SIRINUKUNWATTANA; RAJPOOT, 2015)

Um programa que se disponha a resolvê-lo receberia como entradas as imagens oriundas de análises histológicas de um tumor e, após o devido processamento, espera-se receber como resposta uma classificação numérica de quão alto é o nível de atipia das células observadas. Isso é feito avaliando-se a quantidade de células presentes que apresentam características atípicas, como núcleos e/ou nucléolos muito grandes, membrana celular muito espessa e contorno nuclear irregular.

Na literatura recente, vem-se cada vez mais aplicando técnicas de *Machine Learning* - ML para esse tipo de problema, especialmente um conjunto de algoritmos pertencentes a uma sub-área de ML chamada *Deep Learning* - DL. O procedimento, em geral, é a implementação de um sistema que consiga aprender a identificar autonomamente os padrões necessários para fazer a classificação, obtendo essa "habilidade" a partir de exemplos de imagens já classificadas.

Entre as diferentes arquiteturas de redes neurais, as *Convolutional Neural Networks* - CNN (tradução livre, redes neurais convolucionais) têm tido performances de destaque em aplicações de visão computacional. As características que essas redes extraem das imagens carregam alto valor semântico, possibilitando boa precisão, por exemplo, em tarefas de classificação (WANG et al., 2015).

No presente trabalho é proposta a aplicação de arquiteturas CNN já consolidadas da literatura, proporcionando sua utilização como ferramenta de atribuição de grau de atipia nuclear. Aqui serão usados dois paradigmas de aprendizagem, *Transfer Learning* - TL (tradução livre, transferência de aprendizado) e *Ensemble Learning* - EL (tradução livre,

aprendizado em conjunto). Técnicas de TL permitem a adaptação de redes neurais treinadas em uma tarefa para que apliquem o que foi aprendido a uma tarefa diferente, e técnicas de EL permitem que, a partir dos resultados de diferentes redes neurais independentes, seja possível combiná-los para obter resultados ainda melhores.

1.2 Justificativa

Em 2014, quando ocorreu efetivamente o desafio *Mitos-Atypia-14*, as redes CNN ainda não eram utilizadas da maneira que são hoje. Uma das maneiras mais efetivas para tratar o problema de classificação de atipia nuclear, usada em (KHAN, 2014), era a abordagem do problema como um caso de discriminação de texturas, fazendo a classificação final a partir dos conjuntos de texturas detectados em cada imagem usando-os para, entre outras coisas, segmentar as imagens em regiões de características distintas. De cada região, são posteriormente extraídas características com uso de descritores de imagem, utilizando-as para a classificação final.

O interesse do presente trabalho em aplicar redes CNN a esse problema específico vem das demonstrações mais recentes da efetividade dessa técnica na tarefa de classificação de imagens. Por exemplo, em (ESTEVA et al., 2017), CNNs foram utilizadas na classificação de casos de câncer de pele, tendo sido então demonstrado desempenho equivalente ao de dermatologistas na classificação de melanomas e carcinomas.

Finalmente, em (XU et al., 2017), a abordagem utilizada para classificação, também no dataset do *Mitos-Atypia-14*, é baseada em *Deep Learning*. Nesse caso, foi tirado proveito das três escalas de ampliação das imagens fornecidas no *dataset* do desafio, em ampliação x10, x20 e x40. Foram utilizadas então 3 redes neurais convolucionais, com a mesma arquitetura baseada na rede AlexNet. Cada uma delas foi treinada em um dos valores de ampliação, e para os testes finais cada uma delas alcançava uma classificação independente das demais. A classificação final foi feita pela somatória das probabilidades de classificação de resultante de cada CNN, optando-se pela classe de maior probabilidade após a somatória.

1.3 Objetivos

Objetivo Geral

- Este trabalho tem como objetivo geral o uso de uma rede CNN que seja capaz de classificar o nível de atipia de células em tecidos de câncer de mama, utilizando-se arquiteturas consolidadas da literatura.

Objetivos Específicos

- Realizar uma revisão bibliográfica de trabalhos sobre o problema;
- Obter um banco de imagens no qual aplicar as técnicas propostas;
- Usando técnicas de TL em duas redes CNN consolidadas da literatura serão aplicadas à resolução do problema de classificação de atipia nuclear;
- Utilizar a técnica de EL para melhorar o resultado a partir das saídas das redes retreinadas para o problemas em questão;
- Validar e testar a técnica de classificação implementada;

1.4 Estrutura do Texto

O presente trabalho está estruturado da seguinte maneira:

- **Introdução**: o primeiro capítulo apresenta uma contextualização do problema ao qual será proposta uma solução, assim como ideias iniciais a respeito de como solucioná-lo e referências a trabalhos semelhantes. Além disso, são destacados os objetivos da realização deste projeto de graduação;
- **Embasamento Teórico**: aqui serão apresentados os assuntos tratados neste trabalho, tais como: redes neurais convolucionais e *ensemble learning*;
- **Metodologia**: neste capítulo serão descritas as etapas da técnica aplicada para a resolução do problema em estudo;

- **Resultados**: aqui são descritos os resultados obtidos com os experimentos realizados;
- **Conclusão**: no capítulo final deste trabalho são indicadas as conclusões relativas ao que foi apresentado.

2 EMBASAMENTO TEÓRICO

2.1 Introdução

Este capítulo tem por finalidade estabelecer os conceitos teóricos necessários usados no trabalho. Será então apresentada, inicialmente, uma descrição geral sobre redes CNN, seguida de uma caracterização mais específica das arquiteturas que serão utilizadas no trabalho. Por fim, há também uma explicação sobre as técnicas de TL e EL, que serão usadas no transcurso do projeto.

2.2 CNN

CNN é uma conhecida arquitetura de DL inspirada no mecanismo de percepção visual dos seres vivos. Sua concepção foi possibilitada por meio de uma série de desenvolvimentos em diferentes campos de estudo datando desde o final da década de 1960(GU et al., 2017).

A semelhança com o córtex visual dos seres vivos se dá pela estruturação em camadas. Cada camada da rede recebe dados de entrada, realiza um processamento específico sobre eles e entrega um conjunto de dados a ser tratado pela camada seguinte. As CNNs passaram a ser chamadas de *Deep Convolutional Neural Networks* - DCNN (tradução livre, redes neurais convolucionais profundas) devido a quantidades cada vez maiores de camadas, o que possibilita a execução de tarefas cada vez mais complexas a partir de dados.

2.2.1 Componentes Básicos de uma CNN

Existem inúmeras variantes de arquiteturas da CNN na literatura. No entanto, seus componentes básicos são muito parecidos. Os três tipos básicos de camadas de uma CNN costumam ser: (*i*) convolucionais, (*ii*) de *pooling*, e (*iii*) totalmente conectadas. Na continuação são descritas brevemente cada uma delas.

O objetivo das camadas de convolução é aprender a identificar características de interesse nos dados de entrada. As operações de convolução realizadas fazem a aplicação de filtros sobre as imagens de entrada. Esse processo gera um mapa de características, o qual pode ser usado para identificar o que diferencia uma classe de outra no processo de classificação.

A camada de *pooling* geralmente aparece entre duas camadas de convolução, e atuam

reduzindo a resolução dos mapas de características. Em outras palavras, ocorre um agrupamento das características identificadas em regiões pequenas de dados, sendo possível então identificar as características resultantes em uma visão mais macroscópica. A aplicação sucessiva desses dois primeiros tipos de camadas permite a extração de traços cada vez mais complexos dos dados de entrada.

Após o empilhamento de muitas camadas de convolução e *pooling*, cabe então às camadas totalmente conectadas reunir as características extraídas pelas anteriores. É essa camada que interpreta os dados adquiridos e realiza a classificação em alto nível da entrada recebida em uma das possíveis classes disponíveis.

2.2.2 Treinamento de uma CNN

CNNs em geral fazem uso de algoritmos de aprendizado para ajustar seus parâmetros livres (isto é, as bias e os pesos dos filtros) de modo que se obtenha os valores de saída esperados. O algoritmo comumente usado para este propósito é o algoritmo de *backpropagation* (RUMELHART et al., 1985) (LECUN et al., 1989). Tal algoritmo calcula o gradiente de uma função objetivo (também conhecida como função custo/perda/desempenho) para determinar como ajustar os parâmetros da rede a fim de minimizar os erros que afetam o desempenho na tarefa de predição.

Treinar uma CNN é então um problema de otimização global, em que o objetivo é minimizar a função de perda que fornece uma somatória dos erros existentes entre os valores previstos e os valores reais disponibilizados em um conjunto de testes.

O algoritmo *Stochastic Gradient Descent* (SGD) é a solução comum para otimizar os parâmetros de uma rede CNN. Ao contrário do *Gradient Descent* tradicional, nesse caso trabalha-se com um apanhado do conjunto de dados, e não com toda sua extensão. Isso evita mínimos locais da função de perda, ao mesmo tempo que possibilita uma menor necessidade de memória em disco para que ocorra o processamento.

2.3 Arquitetura de redes CNN

Existem na literatura arquiteturas de redes CNN que foram propostas em estudos da área, e desde então se tornaram populares graças ao desempenho comprovadamente superior em algum quesito de interesse. Essas arquiteturas foram e são aplicadas desde então para

trabalhar com todo tipo de problemas de visão computacional e processamento de imagens, sendo necessário somente que sejam feitos os ajustes para preparar a rede neural para receber e processar corretamente as entradas específicas de cada problema.

Muitas das arquiteturas de sucesso, incluindo as que serão aqui utilizadas, surgem como submissões a competições *online*. Possivelmente a mais famosa competição nessa área de estudo é o *ImageNet Large Scale Visual Recognition Challenge* - ILSVRC, uma competição que ocorre anualmente desde 2010, e que premia os melhores trabalhos em tarefas de visão computacional.

A seguir há um detalhamento das diferenciações trazidas pelas redes VGG e Resnet, que serão utilizadas no presente trabalho, e que são dois exemplos de arquiteturas que surgiram como vencedoras de edições passadas do ILSVRC.

2.3.1 VGG

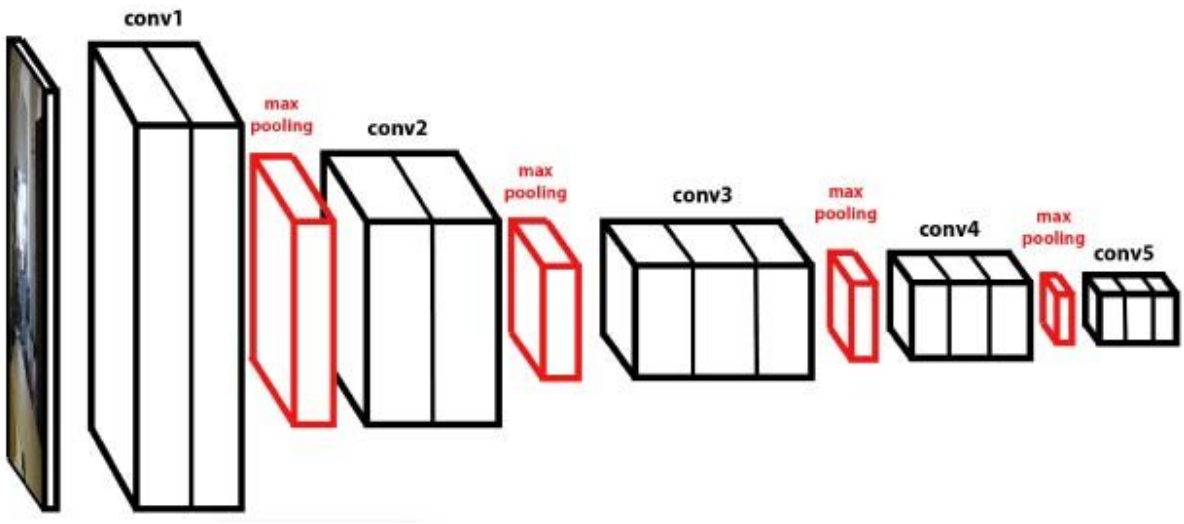
A arquitetura VGGNet foi criada pelo VGG Group (SIMONYAN; ZISSERMAN, 2014), de Oxford, para estudar os avanços provenientes da utilização de redes mais profundas do que as que vinham sendo utilizadas. Os modelos apresentados por eles possuem de 16 a 19 camadas.

A possibilidade de aumento da profundidade em relação a modelos mais antigos veio da escolha de utilizar filtros menores, de tamanho 3×3, entre suas camadas. Foi demonstrado que o empilhamento de 3 camadas 3×3 possui um campo receptivo equivalente a uma camada 7×7, porém com a vantagem de que uma rede com mais camadas permite que haja aprendizado de características mais complexas da imagem analisada.

Esse modelo de CNN foi inicialmente divulgado como uma submissão ao *ImageNet Challenge* 2014, no qual a equipe responsável alcançou o primeiro lugar em localização e segundo lugar em classificação. No artigo publicado posteriormente (SIMONYAN; ZISSERMAN, 2014), foi melhor demonstrado como generalizar a utilização desse modelo para utilização com outros *datasets*.

A Figura 2 ilustra a estrutura das camadas convolucionais e de *max pooling* da rede VGG-16.

Figura 2 – Representação genérica da rede VGG-16.



Fonte: (SAKLA; KONJEVOD; MUNDHENK, 2017)

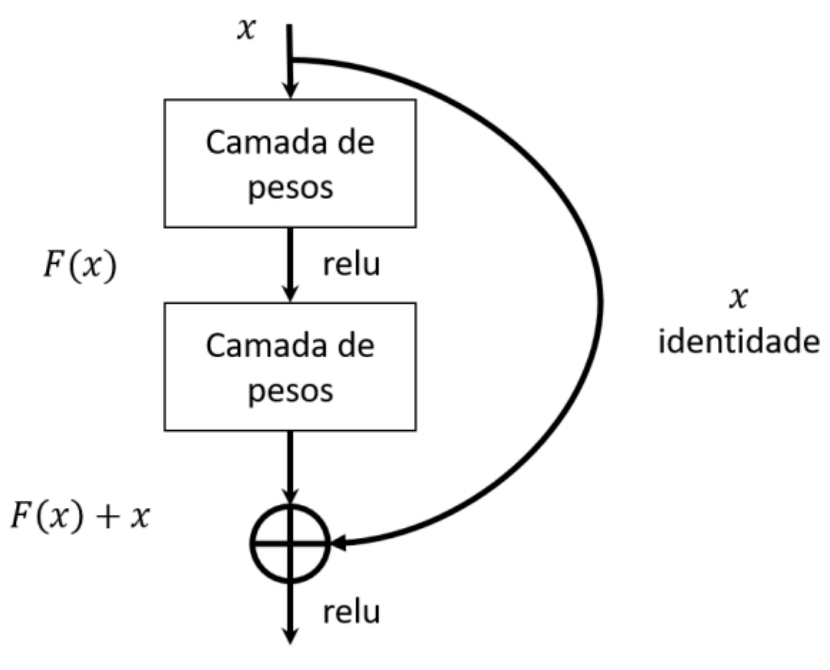
2.3.2 ResNet

ResNet, abreviação de *Residual Networks*, foi uma arquitetura que surgiu, proposta em (HE et al., 2016), para resolver um problema específico: a degradação de precisão que surgiu com a crescente profundidade, ou seja, o crescente número de camadas em redes neurais. A solução encontrada foi a criação de ligações que pulam algumas camadas por vez, permitindo o acesso de camadas mais profundas a funções de ativação de camadas que não são diretamente superiores a elas.

Tais ligações foram implementadas através do elemento chave de uma Resnet, chamado bloco residual. Como visto na Figura 3 cada bloco é geralmente composto por duas camadas de convolução com um ramo identidade em paralelo, ou seja, um caminho que soma ao resultado das convoluções o valor de entrada do bloco.

A utilização de ResNet obteve o primeiro lugar no *ImageNet Challenge 2015*, na tarefa de classificação. As redes utilizadas na ocasião eram oito vezes mais profundas que redes VGG (chegando a 152 camadas), e ainda assim mantinham menor nível de complexidade. O aumento de profundidade proporcionado por redes ResNet é o principal fator que explica o desempenho obtido por elas.

Figura 3 – Bloco residual da Resnet.



Fonte: (ARAÚJO et al., 2017)

2.4 *Transfer Learning*

O treinamento de uma CNN é um processo que exige a aplicação de grande quantidade de dados e processamento para que se obtenham desempenhos aproveitáveis. Por isso, tornou-se prática comum a aplicação da técnica de TL (CHU, 2017).

De posse de uma rede neural vastamente treinada para uma tarefa específica, a técnica de TL consiste em utilizar os valores de pesos obtidos nesse treinamento para a aplicação em uma outra tarefa. Isso é feito removendo-se a última camada totalmente conectada da rede previamente treinada e utilizando o restante da rede como um extrator de características das novas imagens. Somente então ocorre novo treinamento, para ajustar o classificador á base de dados da nova tarefa.

Essa técnica é indicada principalmente quando a base de dados disponível é relativamente pequena, o que já estabelece, por si só, uma limitação à realização de um treinamento tradicional de CNN, que em muitos casos envolve centenas de milhares de imagens.

2.5 *Ensemble Learning*

EL pode ser entendido como aprendizado em conjunto. Isso porque essa técnica consiste em realizar uma combinação das previsões de dois ou mais classificadores, fornecendo como saída um conjunto de previsões mais eficiente do que as que cada classificador foi capaz de fornecer separadamente. Essa é uma das técnicas mais populares na área de ML atualmente, visto que classificadores vencedores de concursos quase sempre realizam *ensembles* em algum ponto (FLENNERHAG, 2018).

O que ocorre na etapa de *ensemble*, em resumo, é uma comparação entre as discrepâncias ocorridas nas previsões de cada uma dos classificadores envolvidos. Definindo-se inicialmente que o resultado final é aquele que a maioria dos classificadores adotou, é possível que haja gradualmente um aprendizado sobre como escolher a melhor opção quando há previsões diferentes recebidas como entrada. Desse modo, as fronteiras entre as possíveis classificações que serão dadas a uma entrada se tornam mais plausíveis do que as que os classificadores iniciais possuíam separadamente.

Outro aspecto favorável para a utilização dessa técnica é que a aplicação de EL nunca piora o desempenho do sistema como um todo. A classificação obtida com essa técnica, na pior das hipóteses, é tão boa quanto a melhor das classificações obtidas pelas redes neurais que alimentam a etapa de *ensemble*.

2.6 Resumo

Neste capítulo, foi apresentado o conhecimento teórico básico sobre as ferramentas e técnicas da área de DL que serão utilizadas no trabalho proposto. Já no próximo capítulo, será descrito especificamente como tais conceitos serão aplicados à resolução do problema em questão.

3 METODOLOGIA

3.1 Introdução

Neste capitulo é descrita a técnica proposta para o problema de classificação do grau de atipia nuclear de células extraídas de tumores de câncer de mama. Tal proposta está baseada em duas etapas, especificamente:

- **Preditor de Classe**. Usando uma metodologia baseada em *patches*, duas redes CNN são treinadas para prever a probabilidade de cada *patch* da imagem de entrada pertencer a uma das três possíveis classes.
- ***Ensamble* de Preditores**. Uma vez treinadas as redes, cada CNN é aplicada a alguns pixels da imagem de microscopia, onde a entrada para cada rede é um *patch* centrado no pixel a ser analisado e a saída é um vetor de probabilidade que indica o grau de associação do *patch* de entrada a cada uma das possíveis classes. A partir dos vetores de probabilidade dos *patches* de cada imagem, são gerados os rótulos equivalentes à classificação de cada um dos *patches*, e pelo critério da maioria (ou moda) é determinada a classe da imagem como um todo.

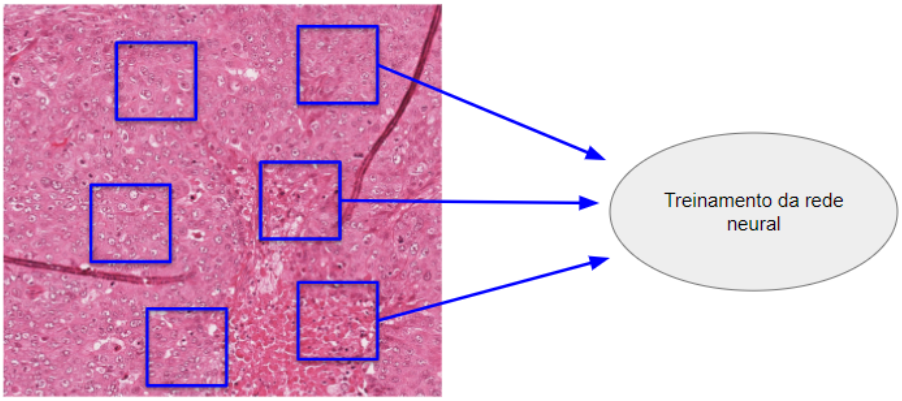
Ambas as etapas serão explicadas em detalhes nas próximas seções.

3.2 Preditor de Classe

Para solucionar o problema de classificação do grau de atipia em imagens de células de tumores de câncer de mama, optou-se pela classificação por *patches* das imagens, ou seja, de pequenos pedaços extraídos de cada imagem da base de dados, para depois utilizar uma função de decisão para fazer a classificação de cada imagem como um todo. A metodologia proposta baseada em *patches* se dá com dois passos principais, explicados abaixo:

1. Para cada uma das imagens de treinamento são extraídos aleatoriamente um conjunto de *patches*.
2. As redes CNN são treinadas usando os *patches* do passo 1, conforme ilustrado pela Figura 4. As arquiteturas das rede são descritas na Subseção 3.2.1.

Figura 4 – Representação da formação do conjunto de treino para as redes neurais



Fonte: o autor

3.2.1 Arquiteturas de Redes CNN usadas

Foram utilizadas duas arquiteturas diferentes de rede neural, e com cada uma separadamente foram desenvolvidas as duas etapas a seguir, explicadas nas subseções seguintes: (*i*) treino da rede neural para classificação dos *patches* a partir dos conjuntos de treino e validação gerados; (*ii*) classificação geral das imagens de teste, dividindo-as em *patches* a serem classificados pelas redes treinadas.

As redes escolhidas para serem treinadas via o paradigma de TL foram as arquiteturas VGG16 e ResNet-50. Foram escolhidas por serem ambas conhecidas por seus bons resultados em tarefas de visão computacional e, dada principalmente a diferença de profundidade das arquiteturas, não são redes similares. Dessa forma, devido à intenção de realizar *ensemble* com seus resultados, espera-se certa variabilidade na resposta das redes à tarefa descrita, não havendo redundância ao utilizar duas redes para a mesma tarefa.

Ao efetuar TL sobre redes CNN, é possível que nem todas as camadas da rede forneçam bons resultados. Uma vez que a mesma foi treinada para uma tarefa distinta. Logo, é comum aproveitar apenas algumas camadas da rede, e inserir outras a fim de serem treinadas do início. Sendo assim, neste trabalho foi definido:

- **Em relação à arquitetura da VGG16**. Foram utilizadas apenas as 13 primeiras camadas convolucionais da VGG16 e foram inseridas mais 3 camadas totalmente conectadas no final. Sendo a primeira com 1024 neurônios, a segunda com 256 e a

última, a qual é ativada pela função *softmax*, com 3, sendo um neurônio para cada uma das classes a serem preditas.

- **Em relação à arquitetura da ResNet-50**. Foram utilizadas as 50 camadas residuais da ResNet-50 e foram inseridas mais 3 camadas totalmente conectadas no final. Sendo a primeira com 1024 neurônios, a segunda com 256 e a última, a qual é ativada pela função *softmax*, com 3, sendo um neurônio para cada uma das classes a serem preditas.

É importante ressaltar que, a saída de interesse de cada rede, pode ser considerada de duas formas: (*i*) como um vetor de três elementos (cada elemento do vetor de saída indica a probabilidade do *patch* de entrada ser de cada uma das possíveis classes) ou (*ii*) como um rótulo de classe, calculado via a aplicação da função *argmax* sobre o vetor de probabilidades. O primeiro formato é de interesse quando é usado um *ensamble* e o segundo formato é de utilidade quando unicamente é usada uma rede.

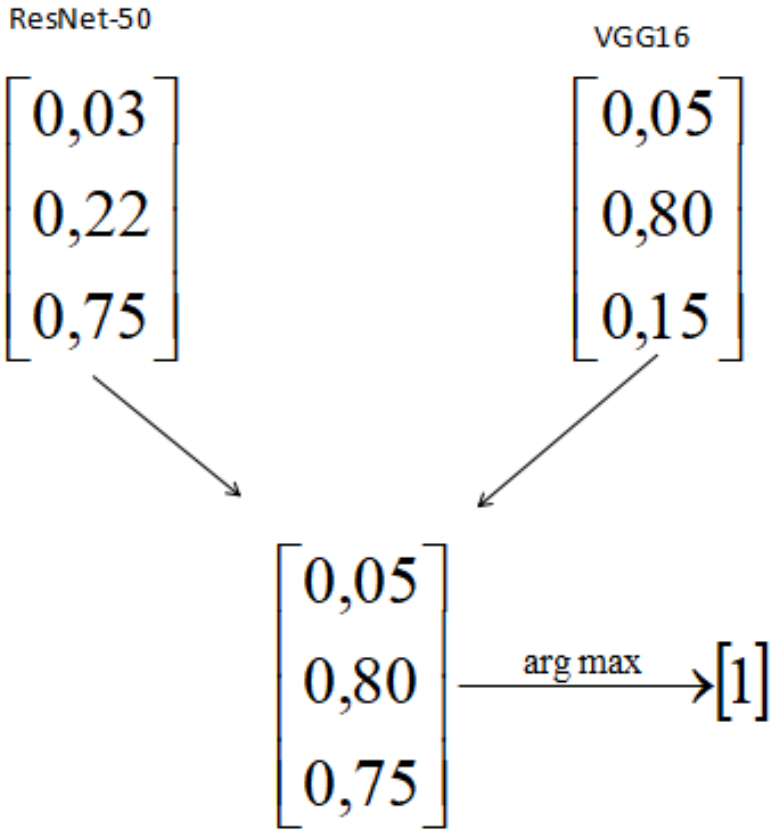
3.3 *Ensamble* de Preditores

Após treinar as redes VGG16 e ResNet-50 para a classificação de *patches*, é efetuada a determinação do grau de atipia nuclear para cada imagem de teste, através do seguinte conjunto de passos:

- Cada imagem de teste é dividida em um *grid* de *patches*. Tal que cada *patch* é centrado em um pixel da imagem e tem algum nível de sobreposição em relação aos outros *patches*.
- Cada *patch* extraído da imagem de teste é individualmente classificado por cada rede já treinada, gerando-se como saída o vetor de probabilidades de que o *patch* pertença a cada uma das classes.
- O resultado da classificação do *patch* é definido considerando o vetor de probabilidades gerados por ambas redes. De modo que, se a classe inferida por elas difere, opta-se por manter a classe que resulta de maior valor de probabilidade, conforme exemplo apresentado na Figura 5.
- Como critério de decisão da classe da imagem como um todo, optou-se pelo critério da moda, ou seja, a cada imagem foi atribuída a classe que aparece mais vezes na classificação dos correspondentes *patches*.

A Figura 6 ilustra o processo como um todo.

Figura 5 – Exemplo do método de *ensemble* aplicado à classificação dos *patches*.

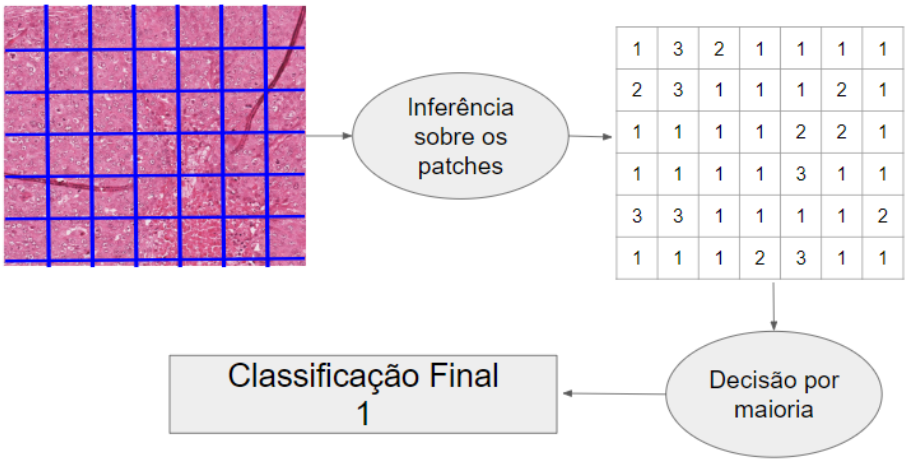


Fonte: o autor

3.4 Resumo

Neste capítulo, foi descrito todo o procedimento adotado para realizar a classificação apresentada como o problema. Foi estabelecido o procedimento de classificação por *patches*, seguido de *ensemble* dos resultados das duas arquiteturas apresentadas, indo da implementação da rede neural à maneira de realização dos testes que levam ao resultado final. No capítulo seguinte, são expostos tais resultados.

Figura 6 – Procedimento de classificação de uma imagem de teste



Fonte: o autor

4 RESULTADOS

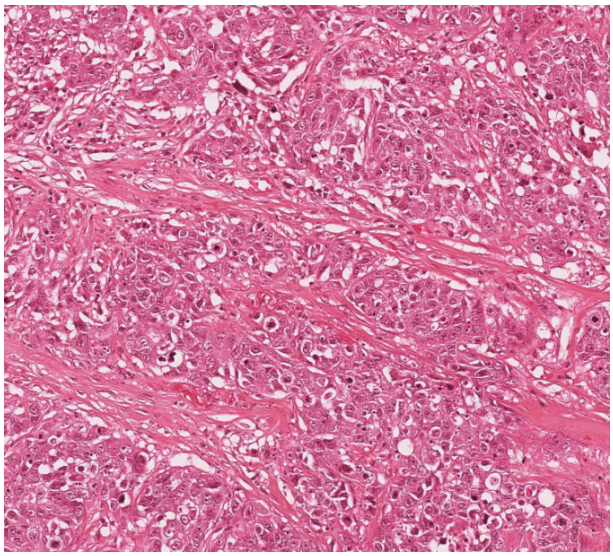
4.1 Introdução

O objetivo do presente capítulo é apresentar os resultados obtidos pela efetuação do que foi descrito no capítulo anterior. Inicialmente são fornecidos detalhes acerca dos recursos que foram necessários para a realização dos estudos e experimentos. Em seguida, apresentam-se os resultados propriamente ditos, os quais são comparados com os de trabalhos semelhantes na área.

4.2 Recursos Materiais

Base de dados Mitos & Atypia 14. Para executar o treinamento e os testes de validação na rede neural desenvolvida no trabalho, é necessário utilizar uma base de dados relativa ao problema em questão. Foi escolhido para esse fim a base de dados do concurso Mitos & Atypia 14, que possui, entre outras vantagens, a de ter sido difundida na literatura a partir de quando foi realizado o concurso, em 2014. O material, disponível publicamente em (MITOS-ATYPIA-14, 2014), foi montado para ser usado em dois tipos de análise: contagem de mitoses e classificação do grau de atipia. A parte de interesse a esse trabalho, relativa à atipia, consiste de 848 imagens obtidas em microscópios, em ampliação $\times 20$, cada uma contendo dezenas de células do tecido observado. Um exemplo desse tipo de imagem é apresentada na Figura 7. O conjunto de imagens da base de dados foi dividido em um grupo para treino e um grupo para testes, especificamente 600 imagens para treino e 248 imagens para teste. Cada imagem recebeu uma classificação de 1 a 3 para seu grau de atipia, a qual foi dada a partir da opinião de dois patologistas, com a intervenção de um terceiro em caso de discordância. A divisão em dois grupos foi feita com o intuito de utilizar o grupo de treino para a calibragem inicial dos algoritmo de classificação supervisionados, utilizando-se depois o de teste para validação e avaliação do desempenho final do classificador.

Figura 7 – Imagem presente no dataset de treino fornecido pelo concurso MITOS & ATYPIA 14



Fonte: MITOS & ATYPIA 14 Contest, 2014

4.3 Recursos Computacionais

Recursos de *Software*. Neste trabalho, será utilizado o software chamado *Tensorflow* para as etapas de treinamento e validação das redes neurais propostas. Trata-se de uma biblioteca de código aberto desenvolvida pelo *Google Brain team*, e a programação envolvida será feita na linguagem *Python*. A implementação das redes neurais tomadas como base para o processamento foi feita em *Keras*, uma API que é executada sobre o *backend* de Tensorflow, possibilitando a utilização de funções de mais alto nível que simplificam a operação das redes.

Recursos de *Hardware*. Para as tentativas iniciais de ajuste dos hiperparâmetros das redes, etapas de treino e teste das redes neurais foi utilizada a plataforma *Colaboratory*, disponibilizada pelo *Google*, em que é possível dispor de uma máquina virtual no servidor remoto da empresa. Tal recurso foi necessário pois a execução da parte mais pesada do trabalho pôde ser assim realizada com auxílio de GPU existente na máquina virtual, pois não era prático o acesso constante a máquinas físicas de grande capacidade.

Para a execução dos treinos e testes para os resultados finais, após já estarem bem definidos os hiperparâmetros, foi utilizada uma máquina do laboratório Viros com especificações: (*i*) processador Intel(R) Core(TM) i7-6850K, 3.60GHz; (*ii*) memória RAM de 64 GB; (*iii*) placas de vídeo NVIDIA Titan V e NVIDIA Titan Xp, cada uma com 12GB de memória.

O computador utilizado para a execução local de parte do trabalho, como a organização

das imagens e labels do dataset de forma que fosse amigável à execução dos programas, é de posse do autor, e possui as seguintes especificações: (*i*) processador Intel Core i5-7200U, 2.70GHz; (*ii*) memória RAM de 8 GB; (*iii*) placa de vídeo integrada *Intel(R) HD Graphics 620*.

4.4 Detalhes de implementação

4.4.1 Tratamento da base de dados

As imagens disponibilizadas na base de dados descrita anteriormente correspondem a capturas feitas por dois diferentes microscópios. Para o presente estudo, foi selecionado o conjunto de imagens capturadas por um deles, de modelo Aperio, correspondendo então a um total de 297 imagens.

Vale adicionar que não foi possível a utilização da conjunto de teste do desafio original, pois as classes das imagens de teste não foram divulgadas. Por isso, todas as imagens usadas para treino e teste nesse estudo estavam contidas no conjunto de treino original obtido.

4.4.2 Implementação das redes

O código para treino e teste das redes neurais foi elaborado utilizando-se *Keras*, uma API para *Tensorflow* que simplifica a montagem e os procedimentos relativos a redes neurais. As funções disponibilizadas pela API permitem a fácil descarga das arquiteturas já implementadas e sua inicialização com os pesos de pré-treinamento na base de dados *ImageNet*.

4.4.3 Etapa de treinamento

O treinamento das redes neurais ocorre através da chamada de uma função específica de *Keras*, que recebe como argumentos funções geradoras que devem ser criadas pelo usuário, as quais repassam ao modelo gradualmente os *batches* de imagens e *rótulos* ao longo do treinamento. Também especifica-se aqui a quantidade de épocas que o treino deve ter como duração.

Optou-se por treinar a rede com *patches* de dimensão 224×224 pixels, e de cada imagem do conjunto de treino foram então extraídos aleatoriamente 512 *patches* com essa dimensão. Como a base de dados original é altamente desbalanceada (23 imagens de classe 1, 222 de classe 2 e 52 de classe 3), não foram utilizados no treinamento da rede todos os *patches* gerados. Somente os *patches* da classe 1 foram usados por completo, limitando-se a quantidade das outras classes a serem iguais à da classe 1, dessa forma o treino foi feito de forma balanceada.

4.4.4 Descrição dos valores dos hiperparâmetros da rede

Os hiperparâmetros que foram utilizados em comum na rede VGG16 e na ResNet-50 são listados na Tabela 1. Devido à diferença de velocidade com que os resultados passavam a apresentar características de *overfitting*, a quantidade de épocas utilizadas com cada arquitetura foi diferente. Na rede VGG16, o treino ocorreu por 20 épocas, enquanto na ResNet-50 foram 7.

Tabela 1 – Hiperparâmetros para o treinamento das redes CNN usadas.

Dropout	0.5
Tamanho de *Batch*	32
Regularizador	L2 com valor de $l = 0,001$
Otimizador	SGD com taxa de aprendizado inicial de $0,001$

4.5 Experimentos

4.5.1 Métricas

No concurso Mitos-Atypia-14, de onde foi obtido a base de dados utilizada, a métrica para a classificação dos competidores foi a acurácia na determinação do grau de atipia de cada imagem fornecida para teste. A cada classificação correta, a equipe recebia 1 ponto, porém se a diferença entre o grau de atipia apontado e o real fosse maior que 1, era descontado 1 ponto como penalidade.

Ao descargar a base de dados do site oficial, não foram disponibilizados os rótulos do conjunto de teste, por isso não é possível aplicar o sistema de pontuação utilizado no desafio original. O conjunto de teste utilizado no presente estudo foi criado como uma porção do conjunto de treino original, e acaba por ser menor do que o conjunto de teste original, além de não ser formado pelas mesmas imagens.

Por esses motivos, os resultados apresentados foram obtidos por meio de *5-fold validation*, técnica em que as etapas de treino e teste são feitas do zero por cinco vezes seguidas, tomando como resultado a média das acurácias obtidas. A cada repetição, a divisão das imagens entre treino e teste é refeita, mas sempre mantendo a mesma proporção, tendo sido aqui utilizada divisão em 80-20%.

4.5.2 Experimento Efetuados

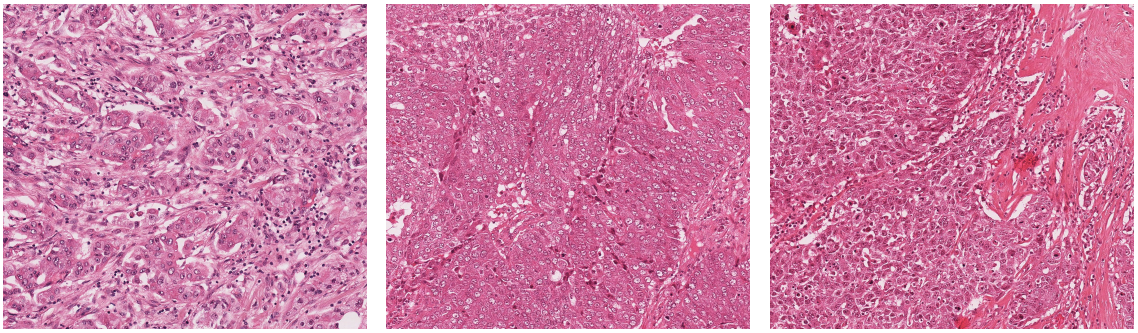
Foram efetuados três experimentos, o primeiro experimento, usando unicamente a rede ResNet-50, o segundo experimento, usando unicamente a rede VGG16 e o terceiro experimento, usando ambas redes e o procedimento de *ensamble* explicado na Seção 3.3.

Os resultados do primeiro experimento são sumarizados nas Tabelas 2, 3, 4 e 5. Onde, a Tabela 2 apresenta os valores de acurácia na classificação do conjunto de teste pela rede ResNet-50 com cada valor de *stride* utilizado, e nas Tabelas 3, 4 e 5 estão as matrizes de confusão correspondentes a cada caso. De forma similar os resultados do segundo experimento são sumarizados nas Tabelas 6, 7, 8 e 9. Por último, os resultados do terceiro experimento, são descritos nas Tabelas 10, 11, 12 e 13.

Observando os resultados em conjunto, percebe-se que a melhor acurácia, $73,44\%$, foi obtida pelo ensamble quando o *stride* utilizado nas imagens de teste foi de 50 pixels. Nenhuma das redes neurais, e nem os *ensembles* realizados apresentaram grande discrepância em seus resultados de acurácia.

A diferença de resultado surgida entre os valores apresentados de uma rede para outra pode ser atribuída à variabilidade que ocorre naturalmente entre duas execuções do mesmo processo, inclusive com os mesmos hiperparâmetros, devido à natureza aleatória de partes do processo de treino de uma rede neural. Isso é reforçado tendo em vista que foi realizada *5-fold validation*, ou seja, os resultados nas tabelas são a média de 5 execuções de treino e teste, com diferentes imagens selecionadas para cada conjunto a cada execução.

As matrizes de confusão mostram que, a classe 1 tem a maior taxa de acerto, chegando em alguns casos ao 100% de acerto (ver Tabelas 5, 8, 12 e 13). A maior quantidade de erros cometidos pelas redes, correspondem à classe 2, sendo elas classificadas como da classe 3. Na Figura 8 há uma ilustração desse caso. É evidente que as imagens destas classes tem uma forte similaridade, implicando uma relação interclasses alta, o que dificulta a tarefa de classificação.



(a) (b) (c)

Figura 8 – Comparação entre imagem de classe 2 classificada pela rede como classe 3 (**b**) com casos de classificação correta de classe 2 (**a**) e 3 (**c**). Foi utilizada a inferência da ResNet-50 no exemplo.

Fonte: MITOS & ATYPIA 14 Contest, 2014

Tabela 2 – Acurácia da classificação da ResNet-50 em função do *stride* das imagens de teste.

Valor do *stride* em pixels	$Ac(\%)$
224	72,46
112	70,49
50	**73,44**

Tabela 3 – Matriz de confusão resultante do teste final da ResNet-50 com *stride* de 224 pixels nas imagens.

	Classe prevista		
Classe real	1	2	3
1	80%	20%	0%
2	2,22%	75,56%	22,22%
3	0%	9,09%	90,91%

Tabela 4 – Matriz de confusão resultante do teste final da ResNet-50 com stride de 112 pixels nas imagens.

	Classe prevista		
Classe real	1	2	3
1	60%	40%	0%
2	11,11%	77,78%	11,11%
3	0%	36,36%	63,64%

Tabela 5 – Matriz de confusão resultante do teste final da ResNet-50 com stride de 50 pixels nas imagens

	Classe prevista		
Classe real	1	2	3
1	**100%**	0%	0%
2	2,22%	**80%**	17,78%
3	0%	45,45%	**54,55%**

Tabela 6 – Acurácia da classificação da VGG16 em função do *stride* das imagens de teste.

Valor do *stride* em pixels	$Ac(\%)$
224	73,11
112	72,13
50	71,15

Tabela 7 – Matriz de confusão resultante do teste final da VGG16 com *stride* de 224 pixels nas imagens.

	Classe prevista		
Classe real	1	2	3
1	80%	20%	0%
2	4,44%	73,33%	22,22%
3	0%	9,09%	90,91%

Tabela 8 – Matriz de confusão resultante do teste final da VGG16 com *stride* de 112 pixels nas imagens.

	Classe prevista		
Classe real	1	2	3
1	100%	0%	0%
2	6,67%	80%	13,33%
3	0%	9,09%	90,91%

Tabela 9 – Matriz de confusão resultante do teste final da VGG16 com *stride* de 50 pixels nas imagens.

	Classe prevista		
Classe real	1	2	3
1	80%	0%	20%
2	2,22%	75,56%	22,22%
3	0%	18,18%	81,82%

Tabela 10 – Acurácia da classificação do *ensemble* das redes em função do *stride* das imagens de teste.

Valor do *stride* em pixels	*Ac*(%)
224	72,79
112	72,46
50	**73,44**

Tabela 11 – Matriz de confusão resultante do *ensemble* das redes com *stride* de 224 pixels nas imagens.

	Classe prevista		
Classe real	1	2	3
1	80%	20%	0%
2	2,22%	75,56%	22,22%
3	10%	9,09%	90,91%

Tabela 12 – Matriz de confusão resultante do *ensemble* das redes com *stride* de 112 pixels nas imagens.

	Classe prevista		
Classe real	1	2	3
1	100%	0%	0%
2	11,11%	68,89%	20%
3	0%	0%	100%

4.5.3 Comparação com outros trabalhos

A aquisição das métricas deste e de outros trabalhos não é padronizada pois não pode ser baseada no conjunto de testes do desafio *Mitos-Atypia-14*, devido à não divulgação dos rótulos do conjunto de testes. Portanto, os testes foram feitos em uma parte do conjunto de treino original segundo o 5-*fold validation*.

Os resultados aqui apresentados atingiram acurácia próxima a registros existentes de outros trabalhos na literatura, principalmente o segundo listado a seguir, que é referência entre os que utilizam técnicas de *Deep Learning*.

Em (KHAN, 2014), mencionado na justificativa deste estudo, foi reportada acurácia de até $83,7\%$ sobre a classificação do conjunto de teste fornecido. Já em (XU et al., 2017), é proposta abordagem baseada em DL, a partir de um modelo MR-CN-PV (*Multi-Resolution*

Tabela 13 – Matriz de confusão resultante do *ensemble* das redes com *stride* de 50 pixels nas imagens.

	Classe prevista		
Classe real	1	2	3
1	**100%**	0%	0%
2	2,22%	**75,56%**	22,22%
3	0%	18,18%	**81,82%**

Convolutional Network with Plurality Voting), também como descrito na justificativa. Em (DAS; NAIR; PETER, 2018), a técnica de Xu é aplicada à base de dados *Mitos-Atypia-14*, atingindo acurácia de 80%.

4.5.4 Resultados Particulares

Em um primeiro momento, houve a tentativa de realizar todos os experimentos na máquina remota do *Google Colaboratory*. Devido a limitações de memória, não era possível o treinamento com toda a base de dados de uma vez, de modo que tentou-se contornar o problema enviando à rede neural aproximadamente 1/5 das imagens para treino a cada época. Isso, somado ao fato de que foram utilizadas menos épocas no treinamento (5 para ResNet-50 e 10 para VGG16), gerou resultados inferiores aos descritos acima.

Vale notar que, também por limitações de memória RAM, as imagens de treino eram divididas em *patches* de 100×100 pixels. Os *strides* aplicados foram de 25, 50 e 100 pixels na extração dos *patches* das imagens de teste. Os resultados obtidos nessas condições são mostrados para efeito de comparação na Tabela 14.

Tabela 14 – Acurácia da classificação de cada rede e do *ensemble* em função do *stride* das imagens de teste, na execução no *Google Colaboratory*.

Valor do *stride* em pixels	ResNet-50 *Ac* (%)	VGG16 *Ac* (%)	*Ensemble* *Ac* (%)
100	65,52	62,07	63,79
50	67,24	62,07	62,07
25	65,52	63,79	62,07

Nota-se que as mudanças realizadas para a execução dos testes como descrito na proposta melhoraram os resultados em aproximadamente 10 pontos percentuais.

4.6 Resumo

Neste capítulo, foram expostos os resultados dos experimentos realizados, colocando-os em comparação com outros trabalhos da área. No capítulo seguinte, evidenciam-se as conclusões a que tais resultados permitiram chegar.

5 CONCLUSÕES E PROJETOS FUTUROS

5.1 Conclusões

O objetivo principal deste trabalho foi propor uma metodologia que permita automatizar computacionalmente a classificação de imagens de células de câncer de mama de acordo com seu grau de atipia nuclear. Para isso optou-se pela utilização de uma abordagem baseada em *Deep Learning*, a partir do uso de arquiteturas de redes neurais convolucionais já consolidadas da literatura, devido à capacidade de tais redes de serem adaptadas a diferentes tarefas, via o paradigma de *Transfer Learning*. Resumidamente, a técnica proposta para o problema de classificação do grau de atipia nuclear está baseada em duas etapas: (i) um preditor de Classe, conformado pelas redes VGG16 e ResNet-50, as quais são treinadas em relação a *patches* extraídos das imagens de treino; (ii) um Ensamble de Preditores, cada imagem de teste tem seus *patches* classificados por uma combinação dos preditores treinados na etapa anterior, para finalmente ser feita a classificação da imagem de teste como um todo, pelo critério da maioria das classificações de seus *patches*.

A técnica proposta, foi avaliada usando-se a base de dados Mitos Atypia 14, considerando a técnica de *5-fold validation*, atingindo no melhor caso uma acurácia média de $73,44\%$. O referido valor de acurácia se aproxima dos resultados de outras abordagens da literatura, comprovando o potencial da proposta para realização da tarefa descrita, principalmente devido à dependência que o desempenho de uma rede neural tem de ajuste de hiperparâmetros no seu treinamento, podendo ser possível a obtenção de resultados melhores com um ajuste mais otimizado de seus valores no futuro. Em conclusão, toma-se como contribuição deste estudo a demonstração do potencial de utilização da metodologia proposta para a resolução do problema abordado, levando em conta a possibilidade de futuras modificações que possam vir a melhorar ainda mais os resultados obtidos.

5.2 Temas a serem pesquisados

Para melhora do desempenho geral, há a possibilidade de utilização de uma rede neural em cascata para fazer a classificação do conjunto de teste. Pretendia-se utilizar essa abordagem neste estudo, treinando-se uma segunda rede neural para classificar as imagens de teste em função do conjunto de classificações de seus *patches*, possivelmente atingindo resultados melhores do que o uso da moda como decisão. No entanto, tal ideia precisou ser descartada

nesse caso específico pois julgou-se o conjunto de imagens disponíveis para treino da segunda rede como muito pequeno, mas pode-se aplicar essa abordagem em eventuais condições mais favoráveis no futuro.

Em desenvolvimentos futuros do estudo, seria também interessante a obtenção de outras métricas sobre os resultados obtidos, como valores de sensibilidade e especificidade, ou *F1-score*, os quais podem trazer à tona outros aspectos do resultado a serem analisados. Vale citar que trariam mais pontos de comparação com outros trabalhos sobre o tema, que disponibilizam uma variedade maior de métricas.

Finalmente, como citado anteriormente, há possibilidade de melhora do desempenho do sistema otimizando-se o treinamento das redes neurais utilizadas através de uma melhor combinação de hiperparâmetros. Sendo possível melhorar a acurácia da classificação dos *patches* dessa forma, também melhoraria a classificação final do conjunto de teste.

REFERÊNCIAS

ARAÚJO, F. H.; CARNEIRO, A. C.; SILVA, R. R.; MEDEIROS, F.; USHIZIMA, D. Redes neurais convolucionais com tensorflow: Teoria e prática. SOCIEDADE BRASILEIRA DE COMPUTAÇÃO. III Escola Regional de Informática do Piauí. Livro Anais-Artigos e Minicursos, Sociedade Brasileira de Computação, v. 1, p. 382–406, 2017. Citado na página 21.

CHU, G. How to use transfer learning and fine-tuning in Keras and Tensorflow to build an image recognition system and classify (almost) any object. 2017. Disponível em: <https://deeplearningsandbox.com/>. Acesso em: 2018-05-19. Citado na página 21.

DAS, A.; NAIR, M. S.; PETER, S. D. Sparse representation over learned dictionaries on the riemannian manifold for automated grading of nuclear pleomorphism in breast cancer. IEEE Transactions on Image Processing, IEEE, 2018. Citado na página 36.

ELLIS, I.; SCHNITT, S.; SASTRE-GARAU, X.; BUSSOLATI, G. Invasive breast carcinoma. In: _____. World Health Organization Classification of Tumours. Pathology & Genetics. Tumours of the Breast and Female Genital Organs. Lyon: International Agency for Research on Cancer, 2003. p. 18–19. Citado na página 12.

ESTEVA, A.; KUPREL, B.; NOVOA, R. A.; KO, J.; SWETTER, S. M.; BLAU, H. M.; THRUN, S. Dermatologist-level classification of skin cancer with deep neural networks. Nature, Macmillan Publishers Limited, part of Springer Nature. All rights reserved., v. 542, p. 115, jan 2017. Citado na página 14.

FLENNERHAG, S. Introduction to Python Ensembles. 2018. Disponível em: <https://www.dataquest.io/blog/introduction-to-ensembles/>. Acesso em: 2018-05-18. Citado na página 22.

GU, J.; WANG, Z.; KUEN, J.; MA, L.; SHAHROUDY, A.; SHUAI, B.; LIU, T.; WANG, X.; WANG, G.; CAI, J. et al. Recent advances in convolutional neural networks. Pattern Recognition, Elsevier, 2017. Citado na página 17.

HE, K.; ZHANG, X.; REN, S.; SUN, J. Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition. [S.l.: s.n.], 2016. p. 770–778. Citado na página 20.

KHAN, A. M. Algorithms for breast cancer grading in digital histopathology images. Tese (Doutorado) — The University ofWarwick, Department of Computer Science, Coventry, UK, 2014. Citado 3 vezes nas páginas 12, 14 e 35.

KHAN, A. M.; SIRINUKUNWATTANA, K.; RAJPOOT, N. A global covariance descriptor for nuclear atypia scoring in breast histopathology images. IEEE journal of biomedical and health informatics, IEEE, v. 19, n. 5, p. 1637–1647, 2015. Citado na página 13.

LECUN, Y.; BOSER, B.; DENKER, J. S.; HENDERSON, D.; HOWARD, R. E.; HUBBARD, W.; JACKEL, L. D. Backpropagation Applied to Handwritten Zip Code

Recognition. Neural Comput., MIT Press, Cambridge, MA, USA, v. 1, n. 4, p. 541–551, 1989. ISSN 0899-7667. Citado na página 18.

MITOS-ATYPIA-14. 2014. Disponível em: <https://mitos-atypia-14.grand-challenge.org/>. Acesso em: 2018-05-19. Citado na página 28.

RUMELHART, D.; HINTON, G.; WILLIAMS, R.; CALIFORNIA, S. D. I. f. C. S. University of. Learning Internal Representations by Error Propagation. [S.l.]: Institute for Cognitive Science, University of California, San Diego, 1985. (ICS report). Citado na página 18.

SAKLA, W.; KONJEVOD, G.; MUNDHENK, T. Deep multi-modal vehicle detection in aerial isr imagery. p. 916–923, 03 2017. Citado na página 20.

SIMONYAN, K.; ZISSERMAN, A. Very deep convolutional networks for large-scale image recognition. CoRR, abs/1409.1556, 2014. Citado na página 19.

WANG, L.; OUYANG, W.; WANG, X.; LU, H. Visual tracking with fully convolutional networks. In: Proceedings of the IEEE International Conference on Computer Vision. [S.l.: s.n.], 2015. p. 3119–3127. Citado na página 13.

XU, J.; ZHOU, C.; LANG, B.; LIU, Q. Deep learning for histopathological image analysis: Towards computerized diagnosis on cancers. In: Deep Learning and Convolutional Neural Networks for Medical Image Computing. [S.l.]: Springer, 2017. p. 73–95. Citado 2 vezes nas páginas 14 e 35.