INSTITUTO TECNOLÓGICO VALE

Programa de Pós-Graduação em Instrumentação, Controle e Automação de Processos de Mineração (PROFICAM)
Escola de Minas, Universidade Federal de Ouro Preto (UFOP)
Associação Instituto Tecnológico Vale (ITV)

Dissertação

MEDIÇÃO DE GRANULOMETRIA DE MINÉRIO DE FERRO ATRAVÉS DE IMAGENS EM CIRCUITO DE BRITAGEM PRIMÁRIA

Lucas Eugênio Ribeiro Souza

Ouro Preto
Minas Gerais, Brasil
2020

Lucas Eugênio Ribeiro Souza

MEDIÇÃO DE GRANULOMETRIA DE MINÉRIO DE FERRO ATRAVÉS DE IMAGENS EM CIRCUITO DE BRITAGEM PRIMÁRIA

Dissertação apresentada ao Programa de Pós-Graduação em Instrumentação, Controle e Automação de Processos de Mineração da Universidade Federal de Ouro Preto e do Instituto Tecnológico Vale, como parte dos requisitos para obtenção do título de Mestre em Engenharia de Controle e Automação.

Orientador: Prof. Gustavo Pessin, D.Sc.
Coorientador: Prof. Thiago Antonio Melo Euzébio, Ph.D.

Ouro Preto
2020

SISBIN - SISTEMA DE BIBLIOTECAS E INFORMAÇÃO

R484m Ribeiro e Souza, Lucas Eugênio .
Medição de granulometria de minério de ferro através de imagens em circuito de britagem primária. [manuscrito] / Lucas Eugênio Ribeiro e Souza. - 2020.
99 f.

Orientador: Prof. Dr. Gustavo Pessin.
Coorientador: Prof. Dr. Thiago Antônio Melo Euzébio.
Dissertação (Mestrado Profissional). Universidade Federal de Ouro Preto. Programa de Mestrado Profissional em Instrumentação, Controle e Automação de Processos de Mineração. Programa de Pós-Graduação em Instrumentação, Controle e Automação de Processos de Mineração.
Área de Concentração: Engenharia de Controle e Automação de Processos Minerais.

1. Ciência do solo - Granulometria. 2. Máquinas - Visão Computacional. 3. Sistemas de computação - Detecção de Objetos. 4. Redes neurais (Computação) - Rede neural convolucional. I. Euzébio, Thiago Antônio Melo. II. Pessin, Gustavo. III. Universidade Federal de Ouro Preto. IV. Título.
CDU 681.5:622.2

Bibliotecário(a) Responsável: Maristela Sanches Lima Mesquita - CRB-1716

MINISTÉRIO DA EDUCAÇÃO
UNIVERSIDADE FEDERAL DE OURO PRETO
REITORIA
ESCOLA DE MINAS
PROGR. POS GRAD. PROF. INST. CONT. E AUT. PROCESSOS DE MIN.

FOLHA DE APROVAÇÃO

Lucas Eugênio Ribeiro e Souza

Medição de Granulometria de Minério de Ferro Através de Imagens em Circuito de Britagem Primária

Membros da Banca

Gustavo Pessin - Doutor - Instituto Tecnológico Vale Mineração

Thiago Antonio Melo Euzébio – Doutor - Instituto Tecnológico Vale Mineração

Andrea Gomes Campos Bianchi – Doutora - Universidade Federal de Ouro Preto

Jefferson Rodrigo de Souza – Doutor - Universidade Federal de Uberlândia

Versão final

Aprovado em 09/09/2020

De acordo,

Agnaldo José da Rocha Reis (p/ Gustavo Pessin).

Documento assinado eletronicamente por **Agnaldo Jose da Rocha Reis**, **COORDENADOR(A) DO CURSO DE PÓS-GRADUACÃ EM INSTRUMENTAÇÃO, CONTROLE E AUTOMAÇÃO DE PROC DE MINERAÇÃO**, em 11/12/2020, às 13:46, conforme horário oficial de Brasília, com fundamento no art. 6º, § 1º, do Decreto nº 8.539, de 8 de outubro de 2015.

A autenticidade deste documento pode ser conferida no site http://sei.ufop.br/sei/controlador_externo.php?acao=documento_conferir&id_orgao_acesso_externo=0 , informando o código verificador **0114211** e o código CRC **91EF32D2**.

R. Diogo de Vasconcelos, 122, - Bairro Pilar Ouro Preto/MG, CEP 35400-000
Telefone: - www.ufop.br

Para meus filhos, Felipe e Lara.
Amor de todas as vidas.

Agradecimentos

Agradeço aos meus filhos Felipe e Lara por serem a razão de acreditar no amor de Deus e em sua obra. Agradeço pelo apoio, presença e torcida de vocês, que foram fundamentais. Peço perdão pelas horas que não pude dedicar à vocês como merecem. Espero deixar como legado para vocês o valor da educação, da curiosidade e do trabalho, que sempre foram muito importantes para mim. Agradeço à minha esposa Natália pela compreensão e pelo exemplo. Com certeza a sua fibra, energia e capacidade de se reinventar foram espelho e também essenciais para que eu pudesse concluir esta jornada. Amo vocês.

Agradeço aos meus pais, Francisco e Ivone e aos meus irmãos Leandra e Luís pelo apoio, amor e por toda a trajetória que contribuíram na minha formação até aqui. Obrigado a todos os familiares e amigos que de diversas formas contribuíram para realização deste trabalho.

Agradeço ao orientador Prof. Gustavo Pessin e ao coorientador Prof. Thiago Euzébio pela receptividade, paciência e disponibilidade em compreender o problema apresentado e impulsionar a pesquisa, investigação e o trabalho científico sobre o assunto, com contribuições valiosas e iluminando o caminho a seguir para realização do trabalho.

Agradeço à Vale e em especial a Gerência de Serviços de Tecnologia em Automação Sul (e todas as versões anteriores deste grupo de trabalho). O apoio na realização deste trabalho acadêmico, contribuições ao longo da jornada e constante troca de experiências foram fundamentais para construir conhecimento técnico para a empresa e para mim. Agradeço às equipes de Vargem Grande o apoio desde a formulação do problema, coleta de dados e trabalhos em campo.

O presente trabalho foi realizado com apoio da Coordenação de Aperfeiçoamento de Pessoal de Nível Superior, Brasil (CAPES), Código de Financiamento 001; do Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq); da Fundação de Amparo à Pesquisa do Estado de Minas Gerais (FAPEMIG); e da Vale SA.

"Tudo, aliás, é a ponta de um mistério, inclusive os fatos. Ou a ausência deles. Duvida? Quando nada acontece há um milagre que não estamos vendo."
(João Guimarães Rosa)

Resumo

Resumo da Dissertação apresentada ao Programa de Pós Graduação em Instrumentação, Controle e Automação de Processos de Mineração como parte dos requisitos necessários para a obtenção do grau de Mestre em Ciências (M.Sc.)

MEDIÇÃO DE GRANULOMETRIA DE MINÉRIO DE FERRO ATRAVÉS DE IMAGENS EM CIRCUITO DE BRITAGEM PRIMÁRIA

Lucas Eugênio Ribeiro Souza

Setembro/2020

Orientadores: Gustavo Pessin
Thiago Antonio Melo Euzébio

A distribuição de granulometria de partículas de minério é de grande importância no monitoramento e controle de processos em diversas fases do beneficiamento. A medição do tamanho das partículas na fase de britagem primária é importante para verificar a qualidade do fornecimento de material das fases de desmonte e operação de mina e também para controle dos equipamentos deste circuito afim de obter taxas de produção maximizadas. A análise granulométrica por imagens apresenta vantagens em função da boa precisão e qualidade das medições e da baixa interferência no processo produtivo. Este trabalho propõe a análise e desenvolvimento de algoritmos de identificação de partículas de minério de ferro e técnicas de medição do tamanho das partículas que atendam aos critérios operacionais de um circuito de britagem primária. Em especial são analisadas as técnicas de aprendizado profundo de máquina e uso de redes neurais convolucionais para detecção das partículas, localização e classificação de imagens. Implementações utilizando as redes SSD, Faster R-CNN, YOLOv3 e U-Net são apresentadas e discutidas no contexto de seu uso na área industrial.

Palavras-chave: Granulometria, Visão Computacional, Detecção de Objetos.

Macrotema: Usina; **Linha de Pesquisa:** Instrumentação no Processamento de Minérios; **Tema:** Redução de Variabilidade e Melhoria de Controle; **Área Relacionada da Vale:** Britagem Primária - Vargem Grande II - Itabirito, Minas Gerais, Brasil.

Abstract

Abstract of Dissertation presented to the Graduate Program on Instrumentation, Control and Automation of Mining Process as a partial fulfillment of the requirements for the degree of Master of Science (M.Sc.)

IRON ORE PARTICLE SIZE MEASUREMENT USING IMAGE ANALYSIS ON CRUSHING CIRCUITS

Lucas Eugênio Ribeiro Souza

September/2020

Advisors: Gustavo Pessin
Thiago Antonio Melo Euzébio

The ore particle size distribution is a key variable for process monitoring and control in various beneficiation stages. Particle size measurement on primary crushing circuit is important to verify quality of ROM (run-of-mine - raw mineral extracted from the mining pit for further processing or treatment) supply and also to control crushing equipments in order to obtain maximized production rates. Particle size distribution analysis through digital image processing presents advantages due to measurements good precision and quality and process low interference. This work proposes image acquisition and processing methods comparison and evaluation over crushing circuit operational requirements, algorithm development for iron ore particles identification and tracking and measuring particle size measuring techniques. Discussions about Deep Learning techniques and Convolutional Neural Networks for object detection, localization and image classification are presented and implementations using SSD, Faster R-CNN, YOLOv3 and U-Net networks are shown and discussed.

Keywords: Particle Size Distribution, Computer Vision, Object Detection.

Macrotheme: Processing Plant; **Research Line:** Instrumentation on Mining Processing; **Theme:** Variability Reduction and Control Improvement; **Related Area of Vale:** Primary Crushing Circuit - Vargem Grande II - Itabirito, Minas Gerais, Brazil.

Lista de Figuras

Lista de Tabelas

Sumário

1. Introdução

1.1. Contextualização

O beneficiamento de minério de ferro é constituído de diversas etapas para processamento do material de origem das frentes de lavra até que se alcancem produtos finais de especificação adequada como as solicitadas pelo mercado.

A fase de lavra é constituída basicamente da extração do material mineral na sua condição disposta na natureza planejada de acordo com levantamentos realizados em pesquisas geológicas que definem qualidades físicas e químicas. Estas propriedades determinam quais porções têm concentrações de ferro adequadas para serem processadas para transformação em produto final para utilização em aplicações metalúrgicas, siderúrgicas, entre outras.

Comumente as reservas minerais de ferro apresentam o material em condições que necessitam um mínimo de processamento posterior à sua extração para redução dos blocos de composição heterogênea de ferro e outros componentes em um tamanho possível para manejo na indústria siderúrgica ou metalúrgica. Para tanto, é necessário cominuir o material extraído, ou seja, realizar operações de redução das partículas minerais de forma controlada até se atingir um objetivo determinado por uma especificação. Em depósitos minerais com baixo teor de ferro concentrado, são necessárias além de operações de cominuição, processos de separação do mineral de interesse (no caso o Ferro) de outros e também a aglomeração das partículas para resultar em um produto de qualidade final dentro das especificações exigidas (WILLS e FINCH, 2015). Portanto, se torna necessário em várias etapas do processo de beneficiamento, conforme observado na Figura 1.1, a medição da qualidade e do tamanho da massa mineral para se controlar os processos em questão.

A britagem é o primeiro estágio mecânico na cominuição, que tem o objetivo reduzir o tamanho dos fragmentos minerais obtidos na operação de mina após desmonte, lavra e transporte para tamanhos em que seja possível realizar o processamento e a separação de materiais de interesse econômico daqueles outros que não é possível aproveitamento nos estágios do beneficiamento.

Gupta e Yan (2016) detalham o processo de redução de tamanho mineral: geralmente é desenhado para ser feito em uma única fase sem realimentação (Figura 1.2a) ou em um circuito fechado (Figura 1.2b). Em alguns casos a combinação destes métodos é adotada. Em um circuito simples, com uma única passagem do material por um britador, ou por uma sequência de britadores, o produto consiste em uma faixa de tamanhos de partículas que não frequentemente atingem o grau de liberação desejado. A abertura na saída do britador determina o tamanho esperado do material. Porém, por se tratar de um processo mecânico da passagem do minério em uma câmara para um processo de

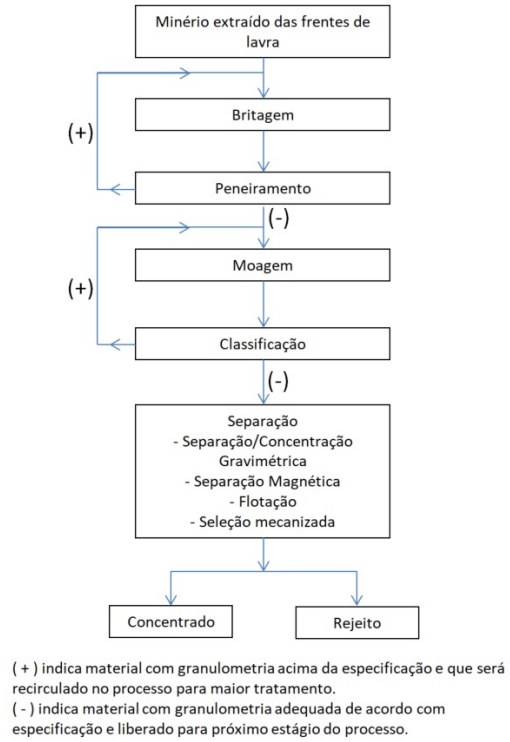


Figura 1.1: Fluxograma de processos de beneficiamento mineral. Adaptado de Wills e Finch (2015).

impactos sucessivos para quebra do material, nem toda a massa sai do equipamento com a granulometria determinada pela regulagem de abertura do britador e assim ocorre uma distribuição gaussiana de tamanhos de partículas dentro de uma faixa (WILLS e FINCH, 2015).

Dependendo da dureza e quantidade do material alimentado, da capacidade mecânica dos britadores e da taxa horária de produção desejada para o circuito de britagem, circuitos com dois ou três estágios são necessários para reduzir progressivamente o tamanho residual das partículas para o tamanho desejável para as próximas fases de beneficiamento (GUPTA e YAN, 2016). Sendo a identificação da granulometria uma variável muito importante para o ajuste do processo de controle, o objeto de estudo deste trabalho é a identificação de granulometria em circuitos de britagem primária para minérios antes do britador primário, de origem da extração da fase de mina.

1.2. Motivação

A medição de granulometria em tempo real de partículas minerais é um processo consolidado conforme cita o texto de Wills e Finch (2015). A verificação desta medição no processo de britagem é de grande relevância para ajustar os parâmetros de controle dos britadores (DI *et al.*, 2019) assim como para verificar se este circuito cumpre o plano

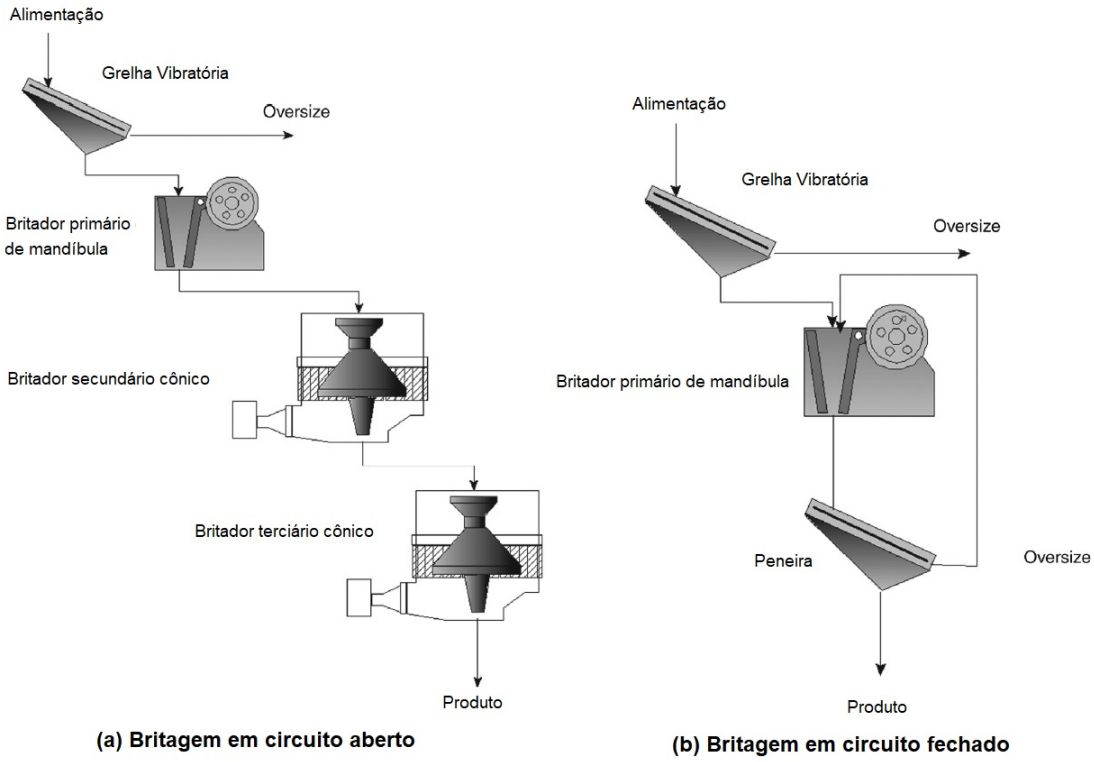


Figura 1.2: Tipos de circuitos de britagem. Adaptado de Gupta e Yan (2016).

de produção, uma vez que quando os equipamentos são alimentados com material mineral fora da faixa granulométrica esperada, existem problemas de desempenho do britador em função do atraso para o processamento e cominuição das partículas (QUIST e EVERTSSON, 2016).

Neste trabalho será analisado o circuito de britagem primária da unidade de Vargem Grande II, conforme descreve o Capítulo 2. Esta fase do beneficiamento mineral, nesta unidade operacional, foi desenhada para uma produção anual de 24,55 Mt/ano. Porém no ano de 2018 a produção medida na saída do conjunto de britadores primários foi de 12,9 Mt. Uma hipótese para este desempenho do circuito abaixo do planejamento inicial é o tamanho de partículas alimentadas, que se estiverem fora da especificação de tamanho, levam maior tempo para serem reduzidas ao tamanho de interesse.

As técnicas de medição de granulometria em minerais tradicionais são realizadas com amostragem manual e encaminhamento para análise em laboratório. Este processo leva um grande tempo até que a informação esteja disponível para as equipes de operação, manutenção, engenharia e demais partes interessadas. Além disso, exigem uma estrutura específica para possibilitar esta coleta manual e envolve riscos para as pessoas durante esta operação, além de perda de eficiência no funcionamento dos circuitos produtivos, caso seja necessário parar o fluxo de material para realizar a extração de amostras. Utilizar a informação de granulometria para corrigir ou controlar processos muitas vezes é inviável em função da demora para obtenção de dados atualizados e da mudança nas características

do processo entre uma amostragem e outra. Métodos que sejam de pouca ou nenhuma intervenção no processo e que atualizem a informação desta variável em menores prazos comparados aos métodos tradicionais elevam a produtividade, regularidade e segurança das operações.

Na operação da britagem primária de Vargem Grande atualmente não há medição *online* da granulometria do minério durante a sua operação. Conhecer o tamanho do material na entrada do britador trará a possibilidade de verificar se o fornecimento de material está na especificação correta e possibilitará desenvolver um controle da velocidade da taxa de alimentação deste equipamento. este controle abre uma alternativa para evitar paradas operacionais por nível cheio do britador, eventuais danos por entupimento e atingir taxas maiores de produtividade de todo o circuito. O objetivo deste trabalho é apresentar alternativas para a medição de granulometria em linha com o processo de britagem primária de Vargem Grande e as características específicas do ambiente industrial para que os métodos apresentados tenham sucesso em sua implantação.

1.3. Objetivos

Propor algoritmos para identificação de partículas de minério e medição de suas dimensões lineares a partir de imagens obtidas na operação de britagem primária.

1.3.1. Objetivos Específicos

- Avaliar algoritmos de processamento digital de imagem para detecção de objetos que possam ser aplicados para o tipo de problema apresentado;
- Desenvolver método de processamento de imagens que inclua detecção de objetos, contagem de partículas e medição de dimensões;
- Verificar esforço computacional, caraterísticas específicas do ambiente industrial e viabilidade em cada um dos casos.

1.3.2. Avanços de projeto

Os objetivos abaixo serão buscados em função do progresso nos objetivos específicos da seção 1.3.1 e determinarão a utilização do trabalho em um ambiente industrial, assim como a sustentabilidade desta solução.

- Realizar montagem de câmeras em ambiente de testes e no prédio da britagem primária de Vargem Grande II e executar algoritmo desenvolvido em estrutura interligada em rede e em tempo real;

- Disponibilizar e armazenar dados de distribuição granulométrica para utilização posterior em sistemas de controle, simuladores ou para consulta de áreas operacionais, manutenção e engenharia para avaliação do processo.

Figura 1.3: Material mineral que se deseja medir o tamanho passando por grelha vibratória na Britagem Primária de Vargem Grande II.

1.4. Estrutura do documento

Este texto está organizado em 6 capítulos, que explicam a motivação e objetivos do trabalho, detalha a unidade produtiva em que o trabalho será realizado e também uma análise da questão operacional que motiva este trabalho técnico, bases conceituais sobre métodos de medição de granulometria de minerais, proposta de trabalho e cronograma. A estrutura e os conteúdos abordados neste documento, em cada um dos seus capítulos, estão resumidos a seguir:

- O Capítulo 2 contém uma descrição da unidade operacional de Vargem Grande II, onde o trabalho será desenvolvido, assim como um detalhamento do circuito da Britagem Primária, com as questões operacionais que motivam a medição de granulometria nesta fase produtiva.

- O Capítulo 3 realiza uma revisão bibliográfica na literatura sobre medição de granulometria *online* de minerais em diferentes metodologias e a evolução até a utilização de processamento digital de imagens para obtenção das dimensões lineares e espaciais dos fragmentos minerais. Detalha também os conceitos sobre processamento

digital de imagens voltado para medição de granulometria, desde a aquisição de imagens, amostragem, métodos de identificação e rastreamento de objetos e processamento computacional das rotinas de análise.

- O Capítulo 4 detalha a metodologia para a implementação de modelos para classificação de imagens, detecção de objetos e medição das partículas de minério de ferro, bem como o procedimento para coleta de dados para as etapas de treinamento e validação dos algoritmos.

- O Capítulo 5 descreve os resultados obtidos com a implementação dos modelos discutidos na Revisão Bibliográfica e Metodologia e uma discussão dos resultados encontrados.

- O Capítulo 6 detalha conclusões sobre o trabalho, discussões sobre melhorias e trabalhos futuros.

2. Processo de Britagem Primária da Usina de Vargem Grande II

Neste Capítulo será discutida a organização industrial da unidade produtiva de Vargem Grande II, a descrição do processo de britagem primária, os requisitos e premissas de projeto desta fase produtiva e a descrição detalhada da britagem primária desta unidade em conjunto com a descrição da forma como a medição da granulometria afeta o seu desempenho operacional.

2.1. A unidade operacional de Vargem Grande II

A unidade industrial onde o estudo será realizado é a Usina de Beneficiamento de Vargem Grande II, pertencente à empresa Vale S.A., localizada na cidade de Nova Lima, estado de Minas Gerais, Brasil. Esta planta é constituída pelas unidades necessárias à recepção e britagem do ROM [1], homogeneização, classificação, moagem, deslamagem, flotação, filtragem, reagentes e estocagem de produtos, compreendendo, basicamente, as seguintes operações unitárias:

- Recepção de ROM
- Britagem primária junto à mina de Abóboras
- TCLD (Transportador de correia de longa distância)
- Peneiramento
- Britagem secundária/terciária
- Britagem quaternária
- Estocagem em pilhas
- Moagem e classificação
- Deslamagem
- Espessamento de lamas
- Estocagem, preparação e dosagem de reagentes
- Condicionamento

[1] ROM - *Run-of-mine* - Massa mineral obtida diretamente da extração da etapa de mina. Composto apenas por materiais com relevância para posterior beneficiamento e excluídas as partes sem aproveitamento, chamadas de material estéril.

- Flotação
- Peneiramento do concentrado
- Espessamento de *pellet feed*[2]
- Filtragem de *pellet feed*
- Sistema de Bombeamento da Pelotização
- Estocagem de produto
- Embarque de produto
- Sistema de Descarga da Usina
- Sistema de Drenagem

2.2. Britagem Primária de Vargem Grande II

O minério lavrado na mina de Abóboras (ROM) é basculado na moega SI-2012VG-01 que é um equipamento que recebe uma descarga de massa mineral proveniente de caminhão ou carregadeira. No caso desta instalação industrial, se trata de uma abertura no piso (conforme Figura 2.3) onde os caminhões podem estacionar próximo e bascular o material. Nessa abertura o material se desloca para o nível inferior do piso onde está instalado o prédio da britagem primária. Esta moega é equipada com a grelha fixa GR-2012VG-02, como visto na tela de sistema supervisório representada na Figura 2.1 e Figura 2.3. A função da grelha fixa é impedir os blocos rochosos com dimensão maior do que a abertura da malha da grelha não entrem no circuito de britagem, uma vez que os equipamentos posteriores não têm capacidade para processamento de tais materiais com dimensões superiores. Estes blocos de grandes dimensões ($\geq$ 800 mm) são chamados de "matacos"pelo jargão industrial da mineração.

Os blocos maiores que a abertura desta grelha (800 mm) serão fragmentados por um rompedor de matacos (BR-2012VG-02). O material de tamanho inferior à abertura da grelha fixa GR-2012VG-02 (Figura 2.3) será transferido para a grelha vibratória GR-2012VG-01 através do alimentador de sapatas AL-2012VG-01 (Figura 2.2). Este equipamento consiste em uma esteira rolante que transporta o material rochoso até ser descarregado na próxima etapa, que é uma grelha vibratória. Esta é uma espécie de peneira que se movimenta constamente com o objetivo de reduzir a aglomeração de materiais

[2]Minério com granulometria menor do que 0,15 mm. É usado misturado ao sinter feed ou para alimentar o processo de pelotização, que transforma o fino de minério em pelotas que serão carga nos altos-fornos siderúrgicos.

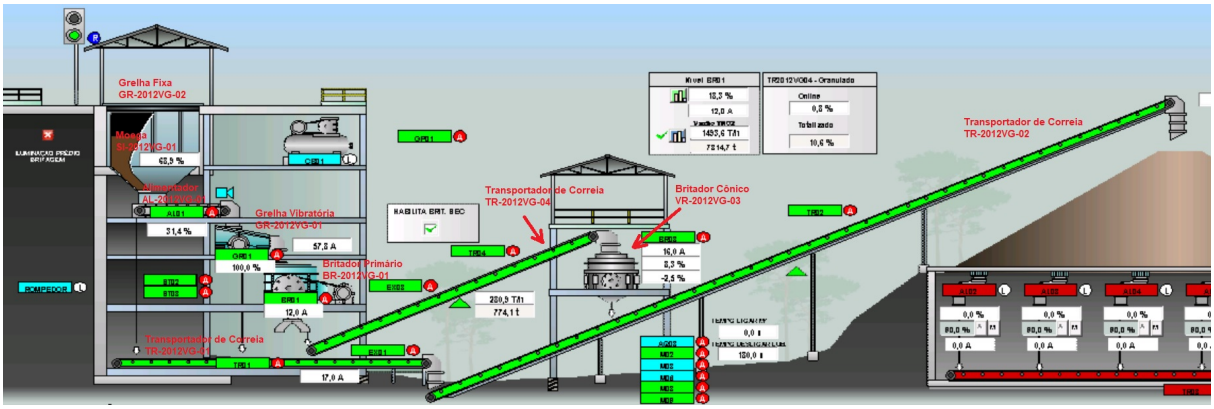


Figura 2.1: Tela de sistema supervisório do circuito de britagem primária de Vargem Grande II. Fonte: Vale S.A.

sob sua superfície e também para aumentar a movimentação dos materiais em direção às aberturas da grelha.

O material passante ($\leq$ 200 mm) da grelha GR-2012VG-01 (Figura 2.4), ou seja, o material que passa pelas aberturas da grelha vibratória e tem granulometria menor do que 200 mm, irá para o transportador de correia TR-2012VG-01. O material retido na grelha vibratória ($\geq$ 200 mm), ou seja, aquele que não passa pela sua abertura, alimentará o britador primário de mandíbulas BR-2012VG-01. Após britado, o material em granulometria já reduzida vai para o transportador de correia TR-2012VG-04 que leva o minério para o britador BR-2012VG-03, para mais uma etapa de redução, com o objetivo de atingir granulometria menor que 200 mm. Após esta segunda redução, o material se junta na correia transportadora TR-2012VG-01 com os fragmentos rochosos de tamanho inferior que passaram pela grelha fixa na primeira etapa do processo descrito.

Após as etapas de britagem e redução do material, o mesmo é encaminhado para uma pilha de estocagem intermediária para ser processado pelas próximas etapas da usina de beneficiamento.(VALE, 2012).

2.3. Requisitos e Premissas do Projeto de Vargem Grande II

Para atingir a produção anual de 10 Mt de minério (entre *sinter feed*[3] e *pellet feed*), a alimentação da fase úmida de beneficiamento deve ser um volume maior, uma vez que a recuperação mássica do ferro na massa de minério varia de acordo com a composição do mesmo. O fator utilizado no projeto é de 40,7%, ou seja, para uma produção final de 10 Mt/ano é necessária uma alimentação de 24,55 Mt/ano. A produtividade do circuito de britagem pode ser determinada através das taxas de cada equipamento e do rendimento

[3]Minério com granulometria entre 6,3 mm e 0,15 mm, que é aglomerado via processo de sinterização para permitir a sua utilização pelos altos-fornos siderúrgicos na forma de sínter.

Figura 2.2: Vista do prédio da britagem primária de Vargem Grande II, alimentador AL-2012VG-01 e britador de mandíbula BR-2012VG-01.

Figura 2.3: Vista da grelha GR-2012VG-02.

Figura 2.4: Vista da grelha GR-2012VG-01.

operacional dos mesmos[4]. A taxa de projeto do britador BR-2012VG-03, que é a saída da etapa de redução dos materiais entre 200 mm e 800 mm é de 441 t/h. Considerando o rendimento operacional deste equipamento em 68,5%, em um total de 6.000 horas de operação por ano, a produção anual do mesmo é de 2,65 Mt/ano. Ou seja, do total de 24,55 Mt/ano de produção do circuito de britagem, 2,65Mt (ou 10,77%) pode ser processado pelas linhas do britador primário de mandíbula e o britador cônico. O restante da fração do material (89,28%) deverá ser de material com granulometria inferior a 200 mm no momento da alimentação no circuito de britagem primária, passando pela grelha vibratória GR-2012VG-01 como *undersize* (Figura 2.5). Esta restrição afeta a taxa de produção do circuito uma vez que materiais de maior granulometria gastam mais tempo nas fases de britagem (britadores BR-2012VG-01 e BR-2012VG03) para serem reduzidos no tamanho especificado e podem representar paradas de produção por enchimento das câmaras dos britadores, uma vez que o britador primário BR-2012VG-01 tem uma capacidade de projeto de 473,2 t/h e o britador cônico BR-2012VG-03 tem capacidade de projeto de 441

[4]Rendimento Operacional: produto entre a Disponibilidade Física e a Utilização Física de cada equipamento. Disponibilidade Física: fração de tempo em que um equipamento está disponível para exercer a sua função, excluindo o tempo em que esteve parado para manutenção. Utilização Física: fração de tempo que é a razão entre as horas efetivamente utilizadas (excluídas aquelas não utilizadas por razões operacionais) e o total de horas disponíveis (tempo total de um período menos a parte dedicada à manutenção de um equipamento).

t/h (VALE, 2012).

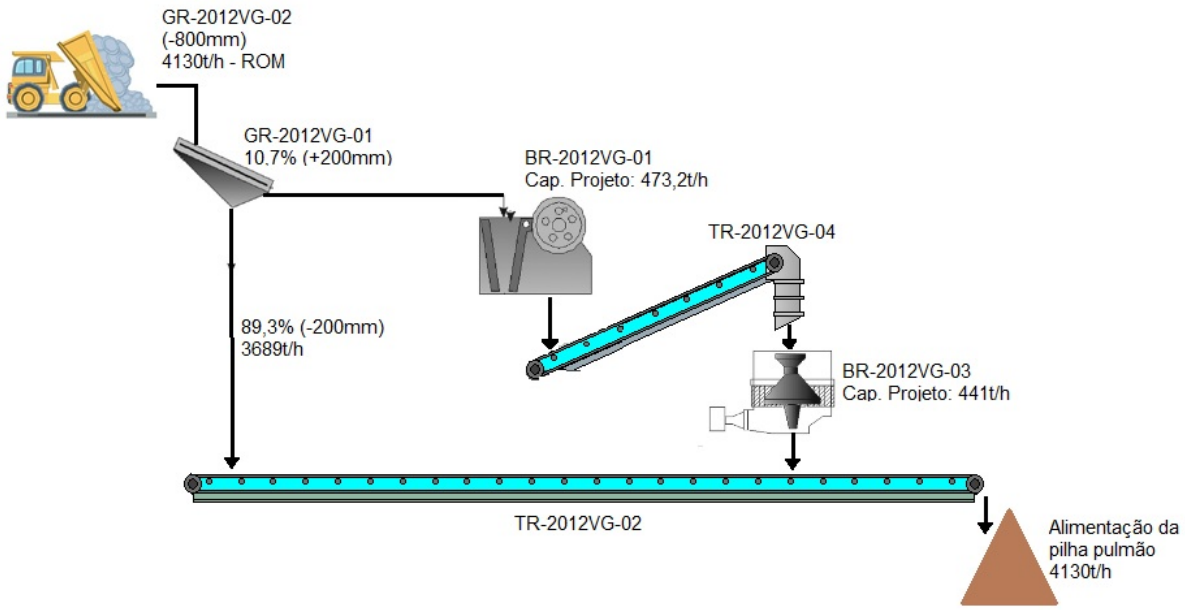


Figura 2.5: Esquema da vazão de material no circuito de britagem primária de Vargem Grande II de acordo com distribuições granulométricas segundo projeto (VALE, 2012).

2.4. Medição de granulometria na britagem primária para verificação de gargalos operacionais

De acordo com os dados de produção do ano de 2018, observa-se que a vazão total mássica na saída da TR-2012VG-02 foi de aproximadamente 12,9 Mt (Figura 2.6) e a taxa média da vazão mássica desta correia foi de 2566,59 t/h, como observa-se na Figura 2.7 à partir da análise da vazão instantânea de material durante o ano de 2018 (gráfico à esquerda) e a média durante o período (gráfico à direita). Uma hipótese para o desempenho inferior é de que o material alimentado está fora das faixas de especificação, ou seja, há uma fração maior do que 10,77% de material com granulometria superior à 200 mm em processamento na britagem primária e desempenho da produção fica abaixo do estipulado pois os britadores BR-2012VG-01 e BR-2012VG-03 gastam mais tempo para processar materiais de granulometrias maiores e cominuí-los até frações menores do que 200 mm para serem entregues na próxima etapa do beneficiamento, conforme especificação e capacidade dos equipamentos posteriores. Também não há controle na taxa de alimentação do material, ou seja, o alimentador funciona com taxa constante e tem apenas intertravamento[5] com o nível cheio da câmara do britador BR-2012VG-01.

[5]Sequência de condições ou ações que existem para garantir segurança efetiva na operação de um equipamento. No caso, no momento em que a câmara do britador está com seu nível máximo preenchido por material aguardando ser processado, um sinal é enviado para o equipamento alimentador para parar sua operação de envio para o britador, afim de garantir a segurança do mesmo.

No caso de fornecimento em tempo prolongado de materiais com granulometria acima de 200 mm, a própria velocidade do alimentador pode disponibilizar mais material do que a câmara do britador BR-2012VG-01 comporta, criando um cenário de atraso na britagem do material, ou até mesmo entupimento da câmara do britador e consequente parada operacional para desobstrução.

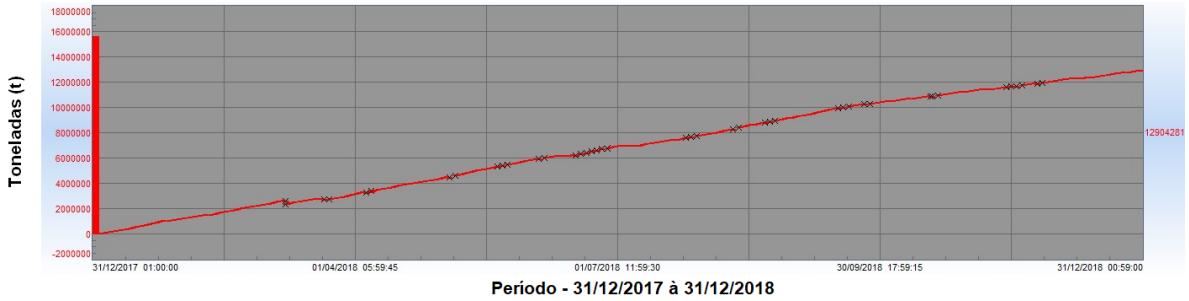


Figura 2.6: Vazão total de material da correia TR-2012VG-02 no ano de 2018. Fonte: PIMS Vale S.A.

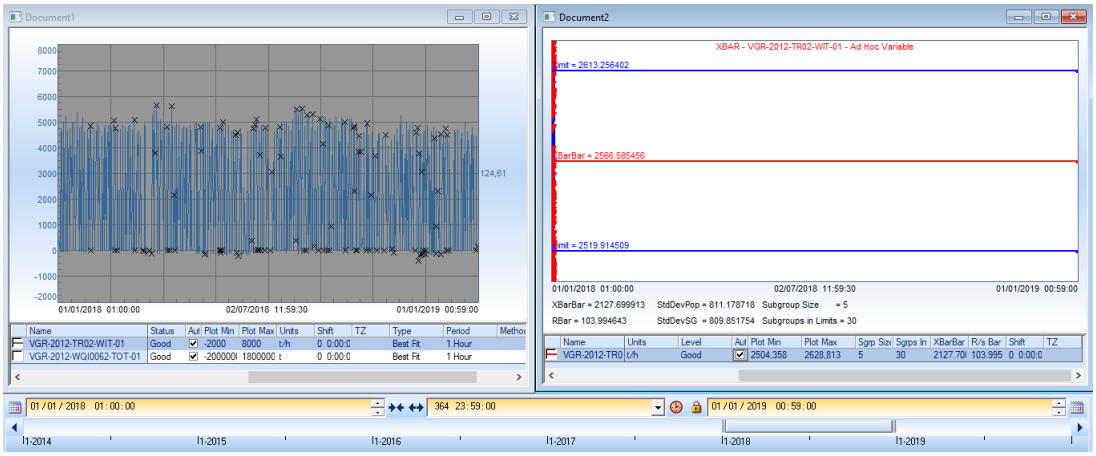


Figura 2.7: Vazão instantânea e média de material da correia TR-2012VG-02 no ano de 2018. Fonte: PIMS Vale S.A.

Atualmente não há nenhuma medição da granulometria do minério no *oversize* da grelha GR-2012VG-01. Dessa forma, conhecer o tamanho do material na entrada do britador BR-2012VG-01 trará a possibilidade de verificar se o fornecimento de material pela fase de mina está na especificação correta (abaixo de 800 mm e no máximo 10,77% com tamanho entre 200 mm e 800 mm). Além disso, possibilitará desenvolver um controle da velocidade do alimentador de sapatas AL-2012VG-01 de acordo com a distribuição granulométrica do material que será disponibilizado para britagem. Isto possibilita evitar paradas operacionais por nível cheio do britador e também é ponto de partida para otimizar o circuito de acordo com a distribuição granulométrica do material que é alimentado.

Conhecendo a distribuição granulométrica também é possível simular o funcionamento do britador de mandíbulas e estimar o tempo que ele gasta para processar uma de-

terminada massa com distribuição granulométrica específica ou ainda estimar a produção do circuito ao final de um determinado período (QUIST e EVERTSSON, 2016).

3. Revisão Bibliográfica

Neste Capítulo será realizada a revisão bibliográfica sobre os métodos de medição de granulometria e volumetria em minerais. Serão abordadas também as questões relacionadas ao processamento de imagens e características específicas para implantação de tais métodos na medição da distribuição de tamanho de partículas. A revisão foi organizada seguindo uma linha da evolução dos métodos de medição de granulometria que são aplicáveis ao processo apresentado no Capítulo 2.

Na Seção 3.1, foram listadas desde as abordagens manuais para coleta e medição dos fragmentos rochosos, passando por métodos de físicos e químicos de medição em laboratório, análise de dados correlacionados para determinação do tamanho de partículas, técnicas de análise de imagem para determinação das medidas de granulometria até o tratamento estatístico de dados de imagem para reconhecimento de padrões para posterior obtenção dos valores de dimensão linear e volumétrica dos materiais.

Diante da proposta do trabalho de estabelecer um método de processamento digital de imagens e de detecção automatizada e medição de partículas de minério de ferro, o desenvolvimento passa pelo entendimento do processamento de imagens e da interpretação entre classificação de imagens e detecção de objetos, da técnica para realizar a detecção das partículas, os métodos para estabelecer rotinas para a detecção automática e as métricas comumente utilizadas para avaliação de desempenho de tais rotinas e modelos de detecção automática.

A Seção 3.2 trás uma revisão dos conceitos de classificação de imagens, localização de objetos e segmentação de imagens que são o tipo de resultado fornecido pelos métodos e algoritmos de aprendizagem de máquina e orientam a escolha dos modelos utilizados neste trabalho.

A Seção 3.3 descreve as técnicas de processamento de imagem para extração das características de um objeto em uma imagem de forma que seja entrada para um modelo de aprendizagem para predição ou que sejam incorporadas aos próprios modelos no momento de receber uma nova imagem como entrada e realizar as operações para predição de existência e localização de um objeto na mesma.

A Seção 3.4 aborda os métodos de aprendizagem de máquina para detecção de objetos e que utilizam estruturas mais simples para a tarefa proposta. A Seção 3.5 aborda uma revisão dos métodos baseados em *deep learning* para detecção de objetos e um detalhamento sobre o uso de redes neurais convolucionais para este objetivo.

A Seção 3.6 discute indicadores para avaliação de desempenho dos modelos de detecção.

3.1. Métodos para medição de granulometria de minério de ferro e outros minerais

Hamzeloo *et al.* (2014) descrevem que a determinação da distribuição de tamanho de partículas é necessária para aumentar eficiência energética e desempenho dos circuitos de britagem/moagem. Apesar disso, devido ao tamanho e peso das partículas em fases iniciais da britagem, a medição da granulometria por amostragem manual ou o uso de técnicas como o peneiramento é invasivo, inconsistente e leva muito tempo tanto para acesso ao material, quanto para obtenção dos valores medidos. No processo da britagem primária de Vargem Grande II atualmente é feita uma amostragem manual do material na saída do processo 3 vezes ao dia e não são realizadas amostragens em etapas intermediárias do processo. Devido a este fato, não é possível acompanhar com precisão a eficiência dos equipamentos destinados para redução do tamanho dos fragmentos rochosos.

Wills e Finch (2015) citam os métodos possíveis para determinação da distribuição de granulometria de minerais de acordo com a característica dos materiais e também do processo que eles estão inseridos. O peneiramento é uma técnica utilizada que necessita da amostragem do material e faz passá-lo por sucessivas peneiras de tamanhos diferentes e progressivamente menores até que todo o material seja retido nos diversos estágios. Em seguida o material é pesado e tem-se então a distribuição granulométrica de acordo com o tamanho das partículas. Para partículas que são de tamanhos não adequados ao teste com peneiras, Wills e Finch (2015) mencionam outras metodologias possíveis de medição, como o diâmetro equivalente de Stokes, dado pela equação de velocidade terminal de partículas em um fluído, ou o método de avaliação do tempo de sedimentação, análise por microscópio eletrônico, método de avaliação por impedância elétrica e métodos de difração por laser. Todos estes métodos são laboratoriais e adequados para partículas de tamanhos microscópicos. Ainda sim, são métodos invasivos no processo produtivo. O texto também cita os métodos de análise *online*, para avaliação em polpa de minério ou em superfícies secas como correias transportadoras. No caso de análises em polpa os métodos discutidos envolvem uma tomada de material para amostragem e posterior medição por princípios de reflexão, difração ou deflexão mecânica. Para as análises em correias transportadoras ou outras superfícies em que o fluxo do material ocorre em meio seco, as medições ocorrem através de análise computacional de imagens e também por medição da refração de raios ultrasônicos. Estes últimos métodos não dependem de interrupção no sistema produtivo e são mais adequados para materiais com maior granulometria.

Williams *et al.* (2000) citam em seu artigo a viabilidade de utilizar sensores tomográficos sobre uma correia transportadora em operação para estimação de granulometria. As interferências do material da correia e características do minério transportado alteram o funcionamento e calibração do aparelho, se tornando pouco prático para medições

contínuas e de materiais em volume e composição não uniforme, conforme processo na britagem primária de Vargem Grande II, discutido no Capítulo 2.

Del Villar *et al.* (1996) propõem a criação de um *softsensor* baseado em redes neurais artificiais, um modelo de média-móvel auto-regressivo (ARMA) e filtro de Kalman para prever os valores de granulometria em circuitos de moagem. Foram utilizadas informações de entrada nos modelos como taxa de vazão de alimentação do ciclone, taxa de vazão do *overflow* do ciclone e as densidades destas duas vazões e verificou a correlação das mesmas com medições de laboratório de granulometria do material. Os modelos apresentam boa correlação quando o horizonte de amostragem é relativamente curto e os dados tem baixo nível de ruídos. No caso de uma aplicação como na britagem primária de Vargem Grande II, encontrar correlações em uma vazão que não é linear e que a faixa de granulometria dos materiais é muito dispersa, inviabiliza este tipo de abordagem.

Kaartinen e Tolonen (2008) discutem em seu artigo a utilização de sensores laser para medição da altura do material em uma correia transportadora e uma posterior conversão para um modelo em três dimensões da partícula em que é possível obter as medidas das dimensões e também o volume da partícula. Também foi utilizada uma rede neural artificial para determinação de um objeto a partir da nuvem de pontos coletada dos sensores laser de posição. O método é apurado, porém apresenta falhas como a não diferenciação do tipo de material analisado na correia e que geram nuvem de pontos com dimensões e volumes que não representam um mineral na correia transportadora. Esta abordagem é compatível com o processo descrito no Capítulo 2, porém demanda instalação de quantidade significativa de instrumentação no local. Esta instrumentação estará exposta à condições de partículas em suspensão e outras interferências em um ambiente não controlado, que precisam ser levadas em consideração no momento da implantação.

Liao e Tarng (2009) propõem em seu trabalho um sistema automático de inspeção óptica, que consiste em quatro módulos: carregamento/descarga do material, separação de partículas, módulo de aquisição de imagens e calibração e controlador lógico programável com a lógica para carregamento do material, aquisição de imagens e processamento de imagens. É um sistema que retira amostras do processo corrente, mantém as partículas em uma superfície plana e homogênea e a aquisição de imagens é realizada de maneira estática. O processamento de imagens é realizado com a definição de contraste em escalas de cinza e a posterior determinação de uma elipse equivalente de acordo com a área de interesse que representa cada partícula. O trabalho apresenta resultados consistentes comparados a dados de laboratório, mas demanda a instalação de uma estrutura para realização de amostragem do material de processo e os resultados não são *online*, o que o torna menos atrativo para o processo discutido no capítulo anterior.

Hamzeloo *et al.* (2014) realizaram um trabalho para coleta de imagens em um transportador de correia parado na saída de um britador de mandíbulas. Após a aquisição de imagens, tratamentos como conversão para escala de cinza e medição de dimensões

através do método do maior círculo inscrito na região de interesse que representa cada partícula. Posteriormente, as medições foram utilizadas como entrada para uma rede neural artificial para determinação da distribuição de tamanho de partículas. Os resultados foram relevantes por ser um método de medição de baixo custo e alto nível de assertividade. Mas os resultados dependem muito de realizar o treinamento da rede neural com o maior número de cenários possíveis, com partículas de tamanhos diversos, presença de partículas com tamanhos fora da área de interesse (chamadas de finos quando a faixa granulométrica é menor do que a de menor interesse). Esta metodologia é interessante para o caso do processo da britagem primária por demandar pouca instrumentação e por considerar casos distintos para treinamento da identificação. Em caso de mudança das características do material ou do ambiente, o modelo de identificação deverá ser recalibrado.

Jemwa e Aldrich (2012) descrevem uma montagem realizada para análise em tempo real de tamanho de partículas de carvão em correias transportadoras utilizando a detecção de partículas utilizando um *framework* de identificação estatística de padrões de texturas do mineral. Os resultados apresentados mostram boa eficiência na identificação de partículas, mas o desempenho é diretamente relacionado ao nível de informações da forma esperada na análise. A classificação por texturas na determinação de tamanho de partículas tem bom potencial de aplicação no monitoramento *online* em correias em movimento. Da mesma forma que o método anterior, é aplicável ao problema da britagem primária de Vargem Grande II. A análise de caracteríticas como cor e formas das partículas é muito dependente do material que passa pelo circuito e demanda treinamento com uma massa considerável de dados. Em caso de alteração nas características físicas do material, o modelo deverá ser recalibrado.

Thurley e Andersson (2008) descrevem em seu artigo um protótipo para estimação de volume e medição de dimensões de pelotas de minério através de análise de imagens e reconstrução em 3 dimensões através de sensores laser de posição, utilizando a técnica da triangulação. O trabalho apresenta técnicas de avaliação de tamanho de partículas que estejam visíveis e não visíveis por detecção de bordas e vizinhança das partículas e do perfil de altura do material na superfície. Com estes dados, e após a execução de algoritmos de treinamento de redes neurais artificiais, um sistema computacional fornece uma resposta estimada de quantidade de partículas. O tamanho das partículas é obtido através da execução de algoritmo de avaliação *best-fit rectangle*, ou seja, um retângulo equivalente que encaixe cada partícula detectada de acordo com as bordas identificadas em passo anterior. O sistema apresenta respostas rápidas em função do tipo de câmera de alta velocidade utilizada e taxa de assertividade elevada quando comparado à medidas de laboratório para partículas entre 5 mm e 16 mm. É um método que depende de câmeras mais específicas e que pode não ser robusto o suficiente para condições industriais. Além disso, as condições de iluminação do ambiente, partículas em suspenção e outras

interferências devem ser mantidas sob controle para maior assertividade do método.

Este trabalho aborda a medição de partículas minerais à partir do processamento digital de imagens. Contudo, uma etapa anterior ao processamento é necessária: identificar nas imagens qual parte desta representa efetivamente um fragmento mineral e quais os limites que separam de regiões que não são referentes ao objeto de interesse. De forma similar à Thurley e Andersson (2008), este trabalho utilizará técnicas de otimização para definir a região que melhor representa um fragmento mineral. Porém, em função do ambiente com alto nível de interferências e baixa uniformidade de cores e formas, serão utilizadas técnicas de aprendizado de máquina para devida classificação e localização dos objetos nas imagens, conforme detalhado nas seções seguintes.

3.2. Classificação de Imagens, Detecção de Objetos, Segmentação de Imagens

O conceito de classificação de imagens remete a identificar se existe um objeto nestas e se este objeto pertence a uma determinada classe de um conjunto do qual estão sendo comparados. A localização de objetos em imagens determina onde estes estão e então a detecção de objetos pode ser realizada. Dada a localização de um objeto ainda desconhecido, é realizada então a classificação daquela sub-região da imagem a partir de uma classe pré-definida.

A segmentação de imagens consiste em determinar regiões em que um determinado conjunto de pixels tenha um significado ou uma semântica. Ou seja, a segmentação pode ser uma técnica utilizada para detecção de objetos. E a partir de um treinamento de uma biblioteca de pixels organizada, a classificação pode ser utilizada para determinar a qual classe pertence um dado segmento de imagem. Pela terminologia, a segmentação semântica é a marcação de áreas na Figura que pertençam a uma classe de objetos. A segmentação de instâncias é a aplicação das máscaras para identificação única de cada objeto. A Figura 3.1 demonstra estes conceitos. Usualmente, os algoritmos de detecção de objetos determinam uma caixa retangular que contém um objeto na imagem, mas não trazem nenhuma informação sobre a forma do mesmo. Já a segmentação é a tradução da forma de um objeto.

Para o problema analisado neste trabalho, idealmente é mais útil a informação entregue pela segmentação, uma vez que esta delimita as formas dos objetos detectados. Porém, o esforço para elaborar uma base de dados para treinamento e o desenvolvimento de algoritmos para determinação dos segmentos de imagem pode ser mais trabalhoso do que os algoritmos de detecção de objetos que retornam as caixas retangulares que são mais ajustadas ao objeto que se deseja detectar. Como as medidas de granulometria mineral são dadas pela maior distância linear de uma partícula (e não informações adicionais como

perímetro ou área da partícula), a medição através das caixas de detecção ajustadas também fornecem uma informação útil para o problema apresentado.

Figura 3.1: Exemplos de classificação, detecção de objetos e segmentação semântica e de instância (LI *et al.*, 2016).

Como técnicas para classificação, alguns algoritmos apresentados na Seção 3.5 utilizam redes neurais classificadoras como SVM (*Support Vector Machines*) e *Softmax*. Para etapa de localização, é necessário estabelecer uma rotina para definir o que são os objetos de interesse na imagem e o que são áreas pertencentes ao fundo ou áreas não relevantes para análise. Para tal, existem técnicas diferentes para extração de características de objetos como HOG (*Histogram of Oriented Gradients*) ou métodos em cascata como Filtros de Haar ou LBP (*Local Binary Patterns*). Os resultados são comparados a biblioteca de imagens utilizada como verdadeiras para estabelecer uma correlação de resultados e assim progredir o treinamento de forma a tornar a detecção com a menor taxa de erro possível. A comparação pixel-a-pixel da imagem com as máscaras geradas pela segmentação também é um método de definição dos objetos nas imagens.

Neste trabalho serão abordadas as redes que utilizam a extração de características através de HOG e também a comparação pixel-a-pixel através de máscaras de segmentação. A Figura 3.2 mostra diversos tipos de redes neurais convolucionais agrupadas por propósito de classificação de imagens, localização e detecção de objetos e também de determinação de pixels pertencentes a um objeto, ou seja, a marcação mais aproximada da forma de um objeto na imagem.

3.3. Descritores de objetos e extração de *features*

A tarefa de obtenção do descritor de um objeto (chamado daqui em diante por *feature*) está diretamente associada a qualidade dos métodos de classificação, detecção e segmentação (YANG *et al.*, 2011). A extração de características é utilizada na etapa de treinamento das redes para classificação e detecção.

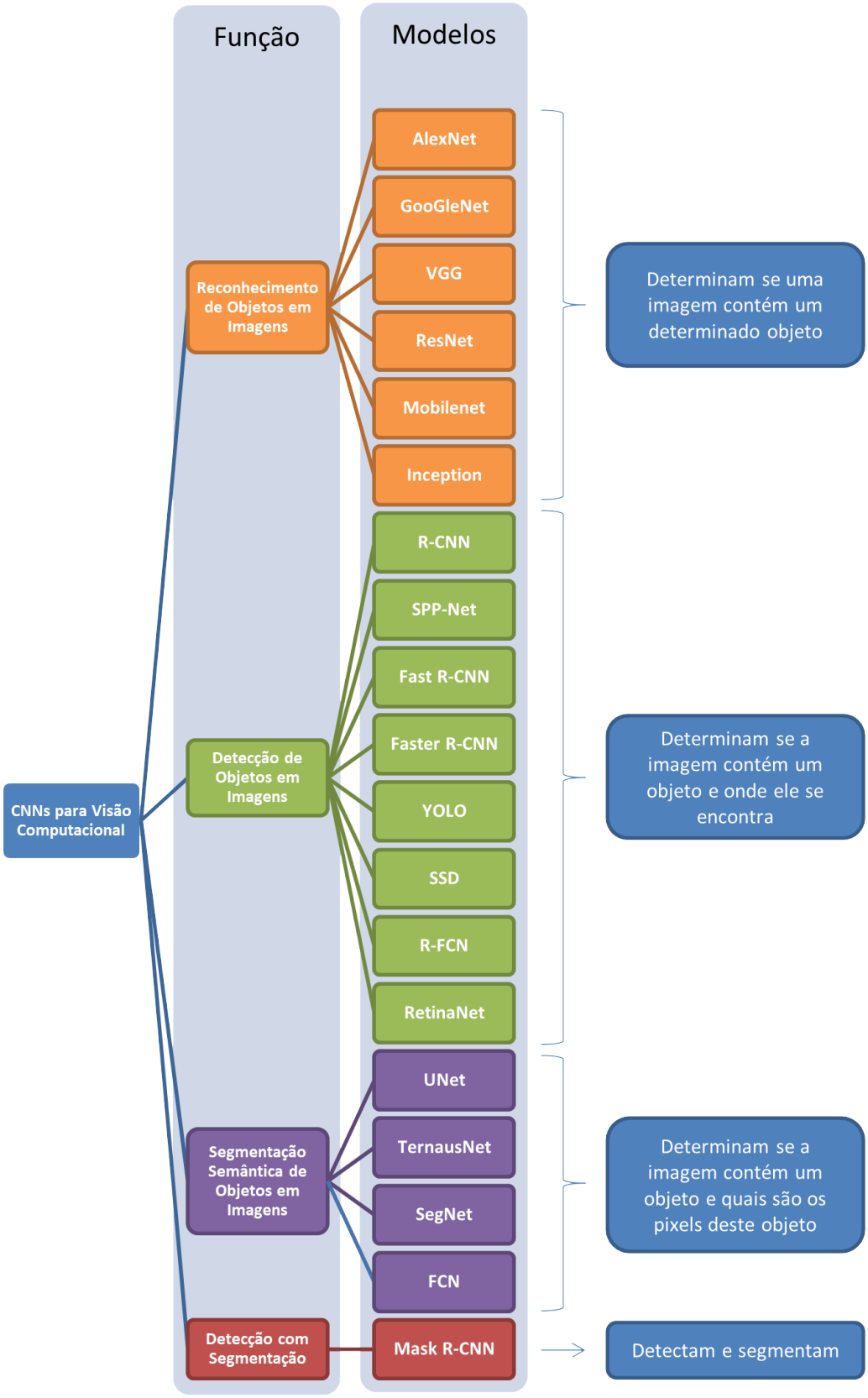


Figura 3.2: Organização de redes neurais convolucionais de acordo com o propósito para reconhecimento e classificação de imagens, localização e detecção de objetos e segmentação semântica de objetos (VON WANGENHEIM, 2019a).

Yang *et al.* (2011) descrevem em seu trabalho os tipos de descritores utilizados para caracterização de um objeto e uso em um algoritmo de rastreamento. O primeiro deles é o de gradiente da imagem que pode ser dividida em duas grandes categorias: contorno ou forma do objeto, conforme o método dos Histogramas de Gradientes Orientados descrito à seguir, ou ainda uma avaliação estatística do gradiente dos pontos, comparado a um padrão previamente estabelecido. O segundo tipo de descritor mencionado é a cor na imagem, em que cada ponto da imagem é avaliado de acordo com sua composição RGB (ou outra escala de cores) e comparado à um padrão previamente estabelecido. O terceiro descritor é o de texturas. A análise destas se dá com o tratamento da imagem com filtros após a conversão para uma escala de cinza e a comparação com um padrão previamente estabelecido. O quarto descritor é o de recursos espaço-temporais: consiste em analisar quadros e estabelecer referencias fixas e compará-las quadro-a-quadro. É necessária uma calibração para transformação da imagem em um volume espacial, ou seja, em 3 dimensões para que seja feita a comparação correta do sentido e direção do movimento. A combinação destes descritores também é utilizada para a caracterização de objetos. Um exemplo é a utilização de matrizes de covariância com dados de gradiente de forma e cores. Outra combinação é a de Mínimos Quadrados Parciais que combinam informações de gradiente de forma, textura e cores.

Um método para extração de *features* discutido em Viola *et al.* (2001) é a utilização de filtros em áreas delimitadas de uma imagem para definição das características que compõem uma biblioteca. Estas áreas são definidas por retângulos divididos em partes iguais. Em cada parte é aplicada uma filtragem de contraste de cores em que o resultado é binário e complementar, ou seja, uma metade do retângulo tem resultado igual a 0 ou "branco"e a outra metade é representada por 1 ou "preto". Em sequência os retângulos tem a orientação em divisão em partes iguais trocada, ou seja, se na primeira iteração a divisão foi vertical, a segunda será horizontal. O procedimento de filtragem e contraste é repetido. Na próxima iteração, a mesma área é dividida em 3 partes iguais e aplicada a filtragem de modo que cada parte tenha um resultado 0 ou 1. A combinação entre os resultados das 3 iterações gera uma caraterística única para cada região da imagem. Para se determinar uma biblioteca consistente de uma identificação de objeto é necessário o treinamento do modelo com um número expressivo de amostras. Em seguida, é necessário utilizar algum algoritmo de classificação como *Winnow* ou *perceptron* (YANG *et al.*, 2000).

Para otimizar o processamento de imagens, a construção da biblioteca de comparação pode ser melhor orientada. Uma técnica mais assertiva é a obtenção do Histograma de Gradientes Orientados (HOG), conforme descrito na subseção à seguir, que significa a amostragem de partes de interesse direto na imagem e obtenção do histograma de cores da imagem. A forma e a aparência ou textura de um objeto podem ser descritas pela distribuição de gradientes de intensidades ou pela direção dos limites do objeto, a partir de uma variação expressiva no valor do gradiente. Para uma implementação prática,

a imagem é dividida em várias regiões menores e o histograma é calculado através do cálculo dos gradientes de direção de variação de tonalidade dos pixels. A caracterização do objeto (ou descritor) é a concatenação destes histogramas (DALAL e TRIGGS, 2005). A implementação completa deste método consiste em elaborar uma biblioteca de amostras de imagens que sejam semelhantes ao objeto de interesse e elaborar os histogramas de gradientes orientados. Também devem ser analisadas uma base de dados de imagens que não sejam semelhantes ao objeto de interesse para que a comparação forneça tenha maior precisão para evitar falsos positivos na detecção (ROSEBROCK, 2014). Com a base de dados composta, devem ser utilizados os classificadores para comparação e determinação do resultado, da mesma forma que o método anterior.

3.3.1. Histogramas de Gradientes Orientados

Dalal e Triggs (2005) exploram em seu artigo o uso de Histogramas de Gradientes Orientados como um descritor para detecção de objetos. O conceito é que a aparência e forma de cada objeto podem ser caracterizados por uma distribuição de gradientes de intensidade dos pixels ou pelas direções das bordas, mesmo sem um conhecimento preciso de um padrão para este gradiente ou dos limites das formas do objeto (CRUZ *et al.*, 2013).

Na prática, esta sequência de análises é realizada dividindo a imagem em pequenas regiões espaciais (ou células). Para cada célula, é realizado um histograma de gradientes orientados para uma direção (horizontal ou vertical) (Figura 3.3). Sendo assim, com o conjunto de gradientes orientados formam a representação. Tratamentos adicionais devem ser realizados para adequar condições de iluminação e sombreamento das imagens, como realizar normalização de contraste antes de utilizá-las (DALAL e TRIGGS, 2005).

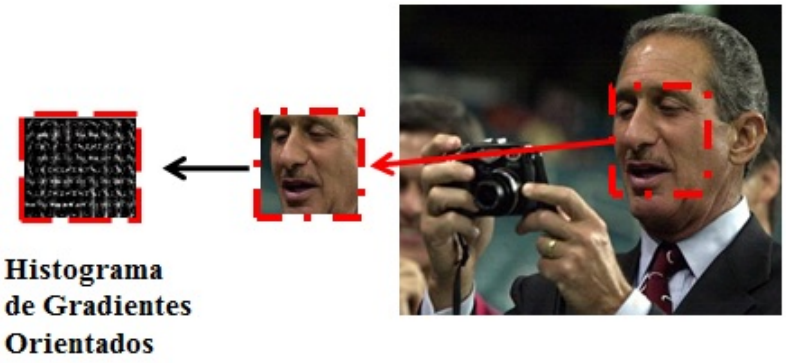


Figura 3.3: Extração de características de imagens (RAMÍREZ CERNA, 2014).

Este método pode ser dividido em quatro etapas: cálculo do gradiente em cada pixel, agrupamento de pixels em células, agrupamento das células em blocos e obtenção do descritor. O descritor é uma lista dos histogramas de todas as células de todos os blocos (CRUZ *et al.*, 2013). A partir do conhecimento de histogramas de referência de um determinado objeto que se deseja identificar, compara-se as listas de histogramas e

então obtém-se uma identificação do objeto em interesse (RAMÍREZ CERNA, 2014).

3.4. Métodos de aprendizado de máquina para extração de características, detecção de formas e objetos

A detecção de objetos pode ser realizada a partir de estruturas de aprendizado de máquina com um fluxo bem definido de atividades como representa o fluxograma na Figura 3.4.

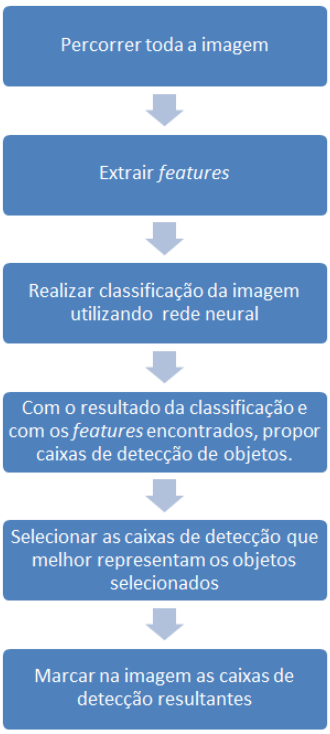


Figura 3.4: Sequência de atividades para detecção de objetos utilizando estrutura de aprendizado de máquina.

As etapas de busca na imagem são descritas na Seção 3.4.1. As etapas de extração de *features* são tratadas na Seção 3.3. As etapas de classificação serão revisadas na Seção 3.5. A etapa de seleção de caixa de detecção mais relevante é tratada na Seção 3.4.2.

A tarefa de identificação de objetos e formas é executada como a operação mais básica das redes convolucionais para classificação, detecção e segmentação de objetos e gera as *features* que construírão um mapa para a devida predição dos objetos nas imagens. A detecção de objetos e formas pode ser realizada com técnicas de processamento digital de imagens como uso de filtros de cores, detecção de bordas entre outros. Porém, para a utilização das imagens como matrizes de informações, aprendizagem de padrões e estabelecimento de algoritmos que sejam capazes de analisar cada imagem independente

de suas condições iniciais de cores e formas, é necessário utilizar técnicas para organizar as informações das imagens em camadas devidamente organizadas. As etapas consistem em análise e varredura da imagem (tratado na Seção 3.4.1, seleção de setor mais representativo para análise (Seção 3.4.2) e extração de características (descrito na Seção 3.3).

3.4.1. Métodos para busca em imagens: *Sliding Windows* e *Image Pyramids*

Um problema apresentado na identificação das partículas de minério durante a aquisição de imagens na britagem primária é que em um mesmo quadro aparecem mais de um fragmento rochoso que precisa ser analisado, com tamanhos e formas diferentes. Além disso, estes objetos estão em distâncias diferentes da lente e é necessário algum método para que todos os objetos de interesse sejam registrados. Além disso, é necessário um procedimento para analisar toda a extensão da imagem para garantir que nenhuma região fique fora da classificação.

Uma técnica possível é a de pirâmides de imagens, descrito por Taylor *et al.* (1997). Consiste em uma representação da imagem em várias escalas, e ao procurar pelas várias camadas criadas com escalas diferentes, é possível identificar objetos de interesse em diferentes tamanhos e posições na imagem original. Esta técnica depende de realizar a amostragem das imagens utilizando resoluções diferentes no mesmo momento de coleta, ou de reduzir ou ampliar a imagem através de métodos computadorizados. Após a implementação da pirâmide de imagens, algum dos métodos descritos anteriormente devem ser utilizados para pesquisa dos objetos de interesse nas imagens. Esse processo deverá ser feito recursivamente entre as camadas criadas pela pirâmide de imagens. Esta adaptação foi proposta por Dalal e Triggs (2005), em que são realizadas extrações da imagem a partir de janelas de tamanho definido e estes extratos são analisados. A janela então move-se para próxima posição adjacente e realiza uma nova análise até cobrir toda extensão da imagem, daí o nome *Sliding Windows*. O mesmo procedimento deve ser realizado para cada escala diferente utilizada no procedimento de pirâmide de imagens.

3.4.2. *Non-Maxima Supression* - NMS

As etapas anteriores após a análise da imagem, extração de *features*, classificação e proposição de uma caixa com o objeto detectado, geram um grande número de propostas em função das características de pesquisas como *sliding windows*, ou pirâmide de imagens ou as duas técnicas combinadas. Em função disso, é necessário gerar uma seleção de qual identificação é mais representativa do objeto de interesse. Uma técnica para selecionar a amostra mais representativa é a Supressão Não-Máxima (*Non-Maxima Supression* - NMS) descrita em Felzenszwalb *et al.* (2010). Consiste em analisar cada região detectada e compará-la com as demais. A intercessão entre duas áreas comparadas contém o objeto

de interesse dentro de uma margem de correspondência e coerência com o objeto de interesse identificado na imagem. Esta margem de correspondência deverá ser calculada de acordo com a classificação da imagem selecionada com um modelo previamente criado à partir de uma biblioteca de imagens semelhantes ao padrão desejado (BODLA *et al.*, 2017). A área com maior correspondência é selecionada e as demais são descartadas.

Um problema para esta abordagem descrita na sequência de operações descritas na Figura 3.4 é o grande número de operações de classificação (e o consequente uso de redes neurais convolucionais para classificar cada proposta gerada) apresenta alto custo computacional (LI *et al.*, 2016). A seguir serão analisados métodos que utilizam redes neurais convolucionais e estruturas de *deep learning* que apresentam melhores resultados tanto em relação a velocidade quanto precisão na detecção.

3.5. Métodos baseados em aprendizagem profunda de máquina (*deep learning*) para classificação, detecção e segmentação

Algoritmos de aprendizagem de máquina são usualmente em três tipos: aprendizagem supervisionada, semi-supervisionada ou não-supervisionada. Para os métodos supervisionados, são providos conjuntos de dados de entrada e saídas esperadas ou "alvo", mas que já sejam previamente organizados com algum tipo de correlação. O algoritmo tenta então aprender tais padrões que podem ser usados para mapear uma relação entre os dados de entrada e a saída alvo correspondente, em que o algoritmo checa a precisão a cada iteração e em caso de detecção fraca, uma nova tentativa de estabelecimento de correlação é realizada, assim como um novo teste. Para sistemas não-supervisionados, as entradas e saídas disponibilizadas não tem nenhuma organização prévia e o sistema irá executar testes de forma a estabelecer o melhor conjunto de relações entradas x saídas após n iterações. Para os sistemas semi-supervisionados, algumas entradas são devidamente organizadas e identificadas e outras não, sendo que as entradas identificadas são um ponto de partida para auxiliar na criação das relações e guiam o aprendizado para as entradas não identificadas. Algoritmos supervisionados são utilizados para problemas bem conhecidos de identificação e classificação onde se conhece as entradas e as saídas correspondentes e os demais métodos são para problemas mais complexos (ROSEBROCK, 2017). Neste trabalho serão estudados algoritmos de classificação e detecção de objetos que atuam de forma supervisionada, ou seja, em que são utilizadas bibliotecas prévias para orientar o treinamento dos modelos de classificação e detecção para em seguida utilizá-los para identificar fragmentos rochosos após aquisição de imagens em um ambiente de testes ou similar ao encontrado em uma usina de tratamento de minério.

Zhao *et al.* (2019) descrevem em seu artigo uma revisão dos métodos para detecção de objetos através do uso de aprendizagem de máquina e do *deep learning*. Neste artigo, os autores dividem a detecção de objetos através de algoritmos de aprendizado de máquina em dois grandes métodos. O primeiro é a detecção de objetos genérica, que é realizada a partir de áreas pré-determinadas de pesquisa chamadas de caixas limitadoras (ou posteriormente citadas neste trabalho como *bounding boxes*), em que o treinamento consiste em comparar as *bounding boxes* ao longo da imagem com bibliotecas previamente armazenadas. O segundo é a detecção de objetos salientes nas imagens que é realizada através de análise pixel a pixel da imagem e a identificação de objetos a partir do agrupamento de pixels em um contexto, obtido através de uma biblioteca de imagens em que os objetos estão devidamente marcados, determinando as regiões que os pixels formam uma classe. Zhao *et al.* (2019) citam duas linhas principais de modelos de algoritmos de detecção de objetos sendo:

1. Geração de propostas de regiões para pesquisa (*"region proposal"*) em que cada região é classificada em diferentes categorias de acordo com a biblioteca de pesquisa e treinamento. A identificação ocorre através de uma varredura de uma região de pesquisa por toda a extensão espacial da Figura. A variação do tamanho das regiões propostas fornecem resultados diferentes que quando combinados fornecem o resultado de uma identificação de objeto.

2. O segundo modelo trata o problema de detecção de objetos como um problema de regressão ou classificação. Ao contrário do primeiro método, em que ocorre a varredura de toda a imagem por região proposta de tamanho padrão e a classificação em cada iteração, neste modelo de regressão a imagem é inicialmente dividida em regiões iguais e cada região é analisada também por caixas com resoluções ou tamanhos diferentes, porém centralizadas em cada uma dessas subdivisões da imagem. O resultado é uma regressão das regiões fixas iniciais em caixas de detecção combinadas que tem uma confiança em relação a uma detecção de objeto.

A Figura 3.5 representa os dois modelos de algoritmos citados no parágrafo anterior e as implementações típicas construídas e os anos da divulgação dos trabalhos referentes.

Na linha de algoritmos baseados em região proposta, a primeira implementação observada foi a de rede neural convolucional por regiões (R-CNN apresentada por Girshick *et al.* (2014)). O conceito de *Spatial Pyramid Pooling* (SPP), apresentado por He *et al.* (2015), uma camada adicional é adicionada na rede para que seja possível conectar as camadas convolucionais e as camadas totalmente conectadas com o objetivo de que uma imagem de qualquer tamanho que passe como entrada nas camadas convolucionais seja tratada e entregue para etapa de classificação nas camadas de rede totalmente conectadas. Esta aplicação abre a possibilidade de utilizar resoluções diferentes das imagens

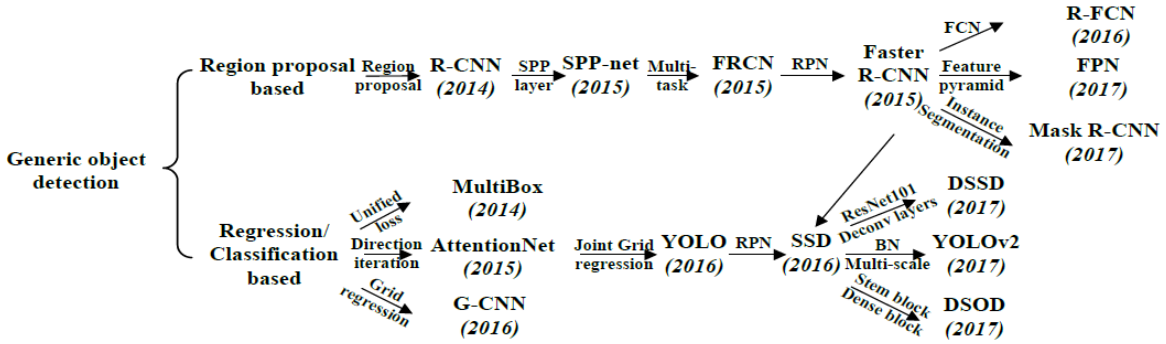


Figura 3.5: Dois tipos de modelos de algoritmos para detecção de objetos: baseados em região proposta e baseados em regressão ou classificação (ZHAO *et al.*, 2019).

(ampliações e reduções) juntamente com as *bounding boxes* para a pesquisa por objetos com diferentes níveis de detalhamento e tamanhos na imagem. Em seguida, o conceito de *multi-task learning* realiza a combinação de busca por regiões e também a segmentação de objetos nas imagens através da análise pixel-a-pixel. Girshick (2015) apresenta um algoritmo para "*Fast R-CNN*"que mostra um aprimoramento do seu modelo anterior ao utilizar o conceito do *multi-task learning* e enviar para a rede convolucional a imagem e também uma região de interesse com uma possível instância do objeto pesquisado para análise em uma única iteração, reduzindo o tempo de treinamento e de detecção de objetos. O maior problema da Fast R-CNN é para encontrar uma região de interesse com uma possível ocorrência do objeto. Sendo assim, várias iterações de busca com *bounding boxes* são necessárias para classificar e formar uma região proposta ideal para análise na camada convolucional. Ren *et al.* (2015) propõem uma rede anterior a esta etapa, com a criação de um mapa de características (ou *features*) e após as operações da rede de proposição de região, são determinadas as âncoras (pontos centrais das *bounding boxes*). Com estes pontos, a região para pesquisa está determinada e são passadas para as camadas de classificação. A partir desta abordagem, otimizações em torno da elaboração da região proposta foram propostas. Dai *et al.* (2016) apresentam uma rede totalmente convolucional (R-FCN - *Region Fully Convolutional Networks*) desde a rede para determinação da região proposta até a etapa de classificação, gerando mapas com pontuações nas etapas de detecção de objetos e de classificação. Lin *et al.* (2017) apresentam uma rede com detecção de *features* utilizando a técnica de pirâmides deslizantes (FPN - *Feature Pyramid Networks*). Nas redes mais primárias, a extração de características acontece em uma etapa inicial, com a escala fixa da imagem. O mapa de características extraído é passado para a rede convolucional para cálculo. Para obtenção da detecção completa, são necessárias varias iterações. A FPN proposta utiliza a extração do mapa de características em cada escala, carregando mais informações em níveis aproximados ou afastados da imagem para uma etapa de convolução. Sendo assim, segundo os autores, ocorre um ganho de qualidade nas detecções com uma redução do tempo de treinamento, aproveitando-se dos benefícios

do *multi-task learning*. He *et al.* (2017) apresentam uma variação da rede Faster R-CNN com a adição de uma rede do tipo FCN para geração de uma área (chamada de máscara) que delimita um determinado objeto através da análise pixel-a-pixel de segmentação.

A outra linha de algoritmos apresentada por Zhao *et al.* (2019) é a dos baseados em Regressão ou Classificação. Várias dos estágios discutidos nos modelos anteriormente se repetem nestes algoritmos, como geração de região proposta, extração de *features* com uma CNN e regressão dos resultados obtidos com uma *bounding box* para detecção de objetos e classificação. A diferença está no fato que os algoritmos baseados em regressão calculam a probabilidade de um determinado pixel pertencer a uma determinada classe de uma biblioteca de treinamento, o que pode trazer uma redução nos tempos de treinamento e de execução da detecção. As primeiras implementações do modelo "MultiBox"proposto por Szegedy *et al.* (2014) utilizam um mapa de pontuações de uma função de erro de classes de detecção unificada para determinar através destes resultados a presença de um objeto para detecção. Yoo *et al.* (2015) apresentam uma rede ("AttentionNet") baseada em cálculos iterativos dos gradientes direcionais na imagem para determinação de pontuações para determinação na detecção de objetos. Najibi *et al.* (2016) apresentaram uma rede que não gera uma região proposta para pesquisa. Utilizando uma matriz de caixas de tamanhos e posições fixas na imagem, esta rede realiza regressões de forma iterativa para determinar em cada elemento desta matriz quais são as partes são referentes a um objeto pesquisado. Redmon *et al.* (2016) apresenta uma rede chamada "YOLO"que combina a proposta de uma matriz de caixa de análise com a formulação de um mapa de pontuações de probabilidade do pertencimento do conteúdo daquele elemento da matriz a uma classe de objeto. O modelo "YOLO"tem dificuldade em lidar com pequenos objetos em grupos, dada sua capacidade limitada de análise da imagem em resoluções diferentes. Baseado neste fato, Liu *et al.* (2016) propuseram uma rede (SSD - *Single Shot Multibox Detector*) baseada no conceito "MultiBox"e também RPN (*Region Proposed Network*) apresentado anteriormente para realizar detecções. A partir de uma região proposta, a rede gera âncoras de possíveis objetos detectados. O processo é repetido iterativamente para escalas diferentes. Para detectar objetos com tamanhos diferentes a rede funde mapas de características obtidos com diferentes resoluções para apresentar um resultado final.

Neste trabalho serão estudadas redes do método de proposta de região (Faster R-CNN e UNet) e também redes baseadas em Regressão e Classificação (SSD e YOLO). A escolha destas redes para estudo foi baseada na disponibilidade de documentação e bibliotecas para desenvolvimento em *Python* de cada uma destas redes e em função de serem redes com bons desempenhos durante a execução em equipamentos de recursos de processamento limitado, comuns em ambiente industrial. O trabalho de (HUANG *et al.*, 2017) faz uma comparação de desempenho entre redes convolucionais e as relações entre precisão na classificação e localização com a resolução e tamanho das imagens, tipos classificadores

e tempo de processamento. As redes Faster R-CNN, SSD e YOLO tem desempenhos relativos bons e são representantes de formas diferentes de organizar as redes convolucionais para a aprendizagem, conforme explorado anteriormente. A intenção de utilizar redes diferentes com a mesma base de dados é comparar os métodos e resultados, utilizando o mesmo ambiente de testes. Já a rede U-Net, que utiliza um método diferente de aprendizado de características, é uma estrutura de relativa simplicidade na implementação e com resultados adequados para o tipo de problema proposto Ronneberger *et al.* (2015).

Os algoritmos de aprendizagem profunda, (citados em diante como *deep learning*) utilizam uma grande quantidade de operações matemáticas para a tarefa de treinamento para estabelecimento de correlações entre entradas e saídas. No contexto de classificação de imagens e detecção de objetos, tais algoritmos utilizam os descritores citados anteriormente como etapa no processo de aprendizagem e não como ferramenta para identificação. Ou seja, tais características são analisadas automaticamente pelos modelos, em função de uma análise pixel-por-pixel da imagem e não baseada em elementos físicos dos objetos registrados nas imagens (ROSEBROCK, 2017).

3.5.1. Redes neurais convolucionais para detecção de objetos e segmentação de imagens

O aprendizado profundo (*deep learning*) é um tipo de aprendizado automático sendo uma especialidade dos métodos de aprendizagem de máquina que modelam problemas com alto nível de abstração utilizando um diverso número de camadas intermediárias, com transformações lineares e não-lineares entre elas para realizar o processo de aprendizagem (GOODFELLOW *et al.*, 2016).

Uma implementação típica de *deep learning* para identificação de objetos é apresentada na Figura 3.6, com as etapas de extração de características destacadas em vermelho e a etapa de classificação de imagem destacadas em roxo. Diferente dos métodos discutidos na Seção 3.5, os métodos aqui discutidos cumprem as etapas de seleção de região e extração de características de forma, com menor custo computacional e de maneira mais robusta em relação à interferências como luminosidade, característica do objeto, plano de fundo, etc (ZHAO *et al.*, 2019).

Zhao *et al.* (2019) citam em seu artigo que a rede neural convolucional (*Convolutional Neural Network* - CNN) é o modelo mais representativo de *deep learning*. Cada camada de uma CNN é chamada de mapa de características (*feature map*). Este mapa é uma matriz em três dimensões com a intensidades dos pixels para os diferentes canais de cor (como por exemplo o modelo de canais de cor RGB). O mapa de qualquer camada interna é uma imagem induzida, composta pelos canais de cores, e é interpretada como uma característica (ou *feature* como descrito em seções anteriores). Cada neurônio é conectado a uma pequena porção de de neurônios adjacentes da camada anterior. Transformações

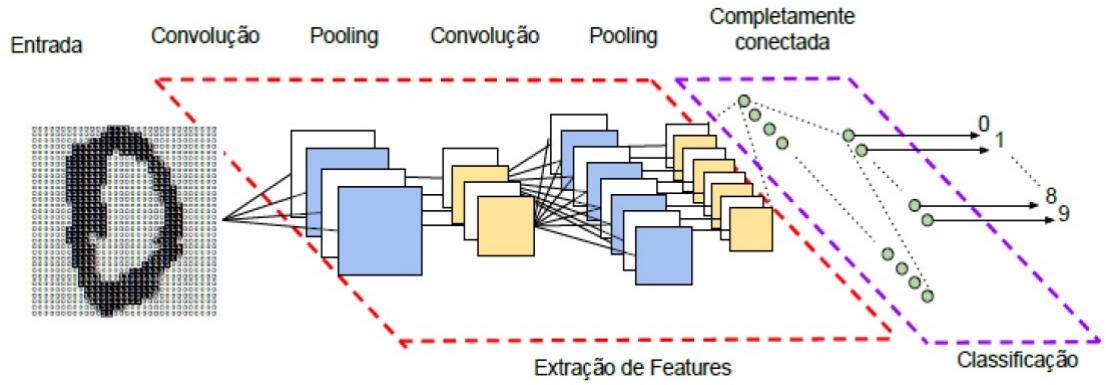


Figura 3.6: Exemplo de estrutura *deep learning* para identificação de imagens composta por rede neural convolucional e rede neural para classificação (VARGAS *et al.*, 2016).

através de funções matemáticas de diversos tipos são realizadas como filtros e *pooling*. A filtragem é uma operação de convolução que utiliza uma matriz com pesos obtidos através de treinamento, com valores das camadas anteriores. Este filtro usualmente tem dimensões menores, porém mesma profundidade do que a imagem analisada e passa por toda a extensão espacial (altura e comprimento) da mesma. Um produto desta operação imagem x matriz é registrado como uma imagem induzida. Filtros diferentes que detectam características (*features*) são convoluídas com a imagem analisada e formam um conjunto de mapas de ativação, que são passadas para a próxima camada da CNN (Figura 3.7).

Após a etapa de convolução, tais mapas de ativação são processados por uma função de ativação. A função de ativação é um nó na rede entre neurônios ou no final da rede que é uma transformação não-linear que executa um processo decisório se aquela imagem induzida é representativa o bastante para continuar o processo de classificação e detecção ou se o processo deve ser interrompido. O resultado da transformação não-linear é encaminhado para a próxima camada de neurônios como entrada.

Após as camadas de convolução e as operações da função de ativação, a operação de *pooling* é realizada. Esta operação é um tipo de subamostragem da Figura ou do mapa de características gerado em etapas anteriores, produzindo um novo mapa com um resumo das informações. O objetivo é reduzir o número de informações para processamento e tornar o processo mais eficiente (HIJAZI *et al.*, 2015). Algumas alternativas para reduzir o tamanho de um mapa (ou imagem) são possíveis como: selecionar o valor máximo (*max pooling*), valor médio (*average pooling*), norma do conjunto (L2-*pooling*), normalização do contraste local entre outras conforme descrito em Zhao *et al.* (2019).

Após as operações de filtragem, transformações não-lineares das funções de ativação e *pooling*, as CNN comumente apresentam camadas do tipo "totalmente conectadas"(*fully connected*). Elas conectam todos os neurônios da camada anterior com os neurônios de saída da rede, que representam as classes a serem identificadas pelas camadas responsáveis pela etapa de classificação, que vem a seguir. Hijazi *et al.* (2015) demonstram em seu

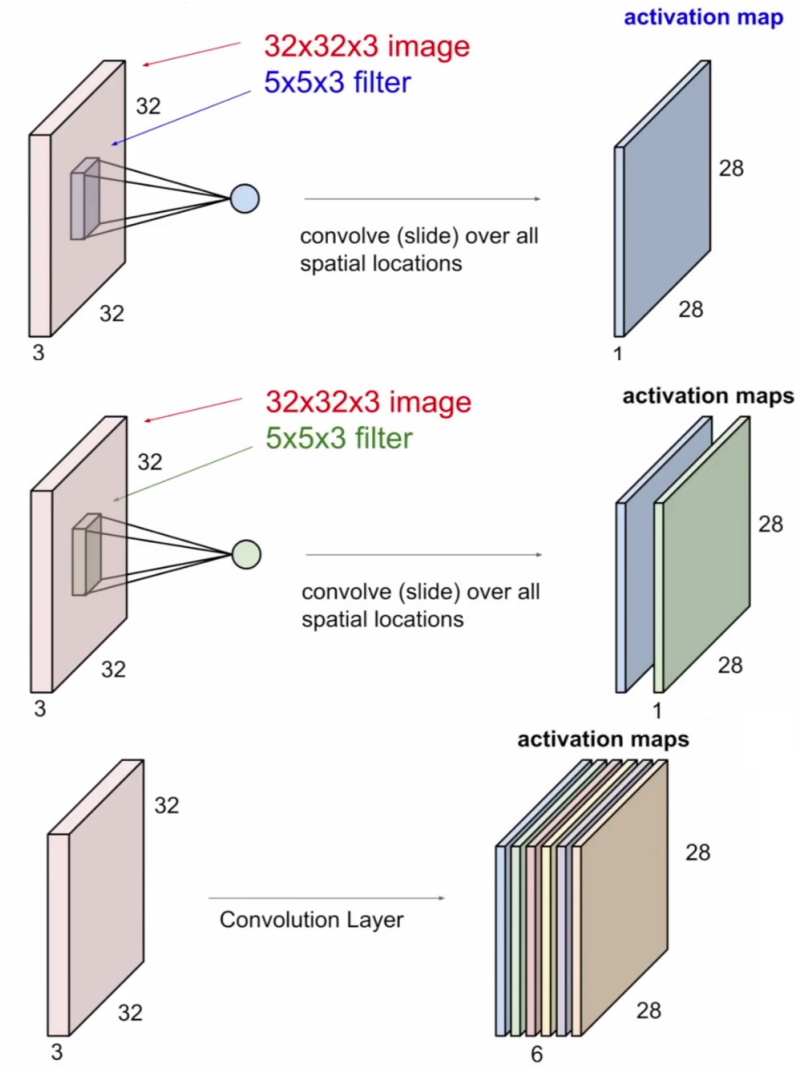


Figura 3.7: Operação de filtro convolucional em camadas em rede neural convolucional (LI *et al.*, 2016). No exemplo, imagem analisada tem formato de 32x32 e 3 camadas referentes aos canais de cores RGB.

artigo um tipo de sumarização de pesos que uma camada totalmente conectada realiza, a partir das camadas anteriores. Na Figura 3.8, dada uma camada totalmente conectada "L", representada pelas caixas azuis, ela realiza a multiplicação e o somatório do mapa de características $Y^{(L-1)}$ da camada anterior e dos pesos $w^{(L)}$ da camada atual. Esta camada realiza o produto e a soma entre todas os mapas da camada anterior, gerando mapas de classes $Y^{(L)}$.

A CNN portanto finaliza na camada totalmente conectada e entrega um mapa de classificação para uma outra rede, com objetivo especifico de classificação de imagens. Um classificador deve atribuir um rótulo a uma imagem, sendo este rótulo pertencente à uma biblioteca de comparação.

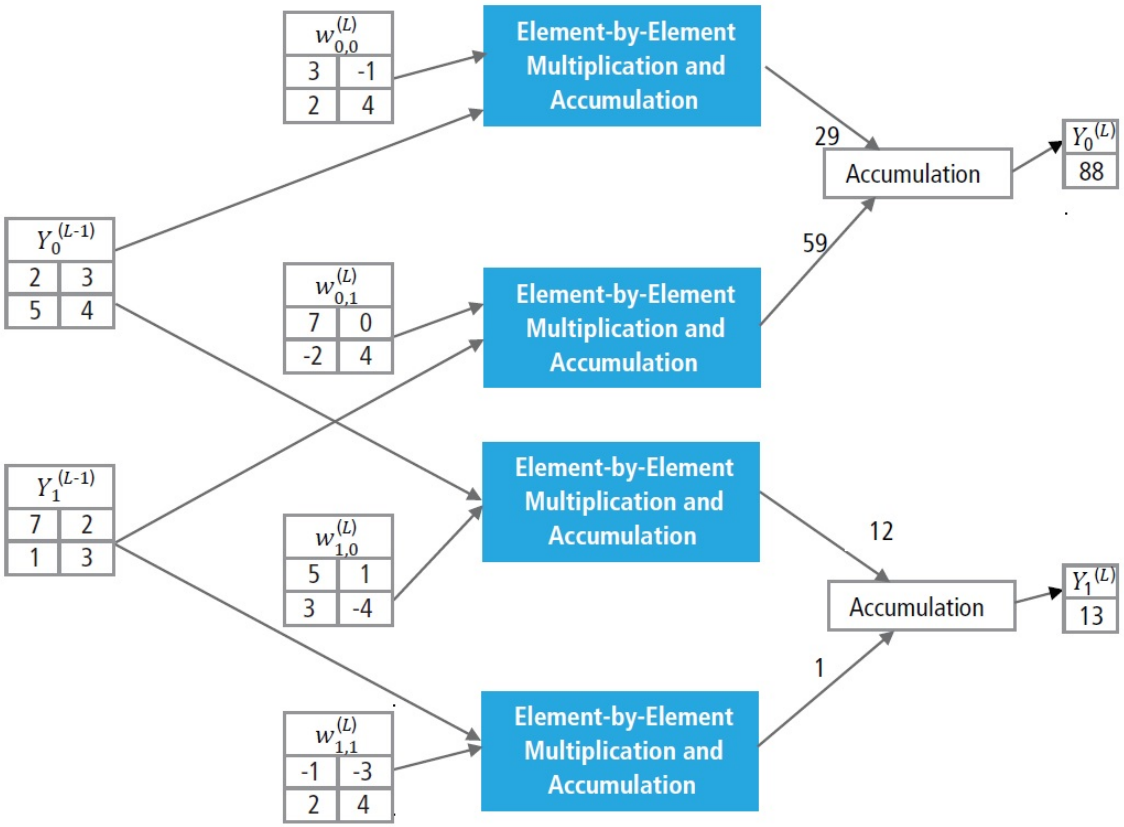


Figura 3.8: Processamento de uma camada totalmente conectada "L"(HIJAZI *et al.*, 2015)

3.5.2. Extratores de *features*

Uma arquitetura típica para uma rede para classificação, conforme a Figura 3.6, é primeiramente a etapa de extração de características (*features*) e em seguida a etapa de classificação. Na primeira parte são utilizados filtros com pesos específicos para as operações de convolução e coleta dos detalhes pertinentes a cada objeto e na segunda fase camadas conectadas fazem o trabalho de comparação dos pesos obtidos com aqueles armazenados em biblioteca de treinamento. Como resultado de cada passo do treinamento são apresentadas as probabilidades de pertencimento de uma imagem a uma determinada classe. Ocorre também o cálculo do erro na comparação da classe obtida na predição com a classe identificada como verdadeira na imagem de entrada. Para os modelos que realizam também a localização do objeto, etapas adicionais são necessárias para determinar as coordenadas que delimitam o mesmo.

As CNNs para extração de *features* funcionam conforme descrito na Seção 3.5.1 e entregam o resultado para as camadas de detecção de objeto ou de classificação, dependendo da função desejada. Desta forma, a estrutura para extração de *features* é chamada de espinha dorsal (ou *backbone*). São arranjos de estruturas de operações como convoluções com filtos, *pooling* e ativações. Algumas estruturas *backbone* como Inception-v2

(SZEGEDY *et al.*, 2016), Darknet-53 (REDMON e FARHADI, 2018), VGG (SIMONYAN e ZISSERMAN, 2014), ResNet (HE *et al.*, 2016) são construídas e otimizadas para melhor aproveitamento de recursos computacionais e com melhores resultados em acurácia na classificação e menores tempos de treinamento.

Neste trabalho serão utilizadas as estruturas *backbone* Inception-v2 para as redes *Faster R-CNN* e *SSD* e Darknet-53 para a rede *YOLOv3*.

Inception-v2

Esta CNN é uma adaptação da rede Inception-v1, que tinha por objetivo resolver o problema de extrair características a partir de imagens não-uniformes em uma biblioteca (objetos centralizados ou deslocados, tamanhos diferentes, etc). Além disso, uma CNN sem uma devida lógica na disposição das camadas e das operações convolucionais pode ter alto custo computacional e tender ao *overfitting*. A proposta portando foi criar uma CNN com filtros de tamanhos diferentes, operando em paralelo para as operações de convolução e concatenar as saídas destas operações logo em uma camada posterior. As operações de *pooling* também são realizadas em paralelo e concatenadas na saída. Esta estrutura modular é repetida 9 vezes na CNN apresentada no trabalho (Figura 3.9). Para reduzir os efeitos do problema de "desaparecimento do gradiente"("*vanishing gradient*") (GOODFELLOW *et al.*, 2016), os autores propõem o uso de classificadores auxiliares em regiões intermediárias da rede (representados pelas caixas azuis na Figura 3.9), de forma que a função de perda total da rede seja composta por uma soma ponderada das perdas intermediárias e da perda encontrada no classificador ao final da arquitetura (SZEGEDY *et al.*, 2015).

A CNN Inception-v2 trouxe como melhoria da versão anterior otimizações para melhorar a eficiência e o custo computacional das mesmas. As redes tem desempenho melhor quando as operações de convolução não alteram drasticamente a dimensão das imagens de entrada. Grandes reduções podem causar perda de informação e consequentemente um gargalo de representação do objeto a ser classificado. A solução apresentada foi realizar operações consecutivas de convolução com filtros de ordens menores até se atingir a ordem desejada para redução, conforme Figura 3.10 (SZEGEDY *et al.*, 2016).

A arquitetura final da rede Inception-v2 é apresentada na Tabela 3.1, com as camadas apresentadas em ordem de implementação da entrada da imagem até a saída com a classificação resultante.

Darknet-53

A CNN proposta por Redmon e Farhadi (2018) utiliza um *backbone* denominada Darknet-53, que tem 53 camadas convolucionais e é uma adaptação da rede Darknet-19 utilizada na CNN *YOLOv2*.

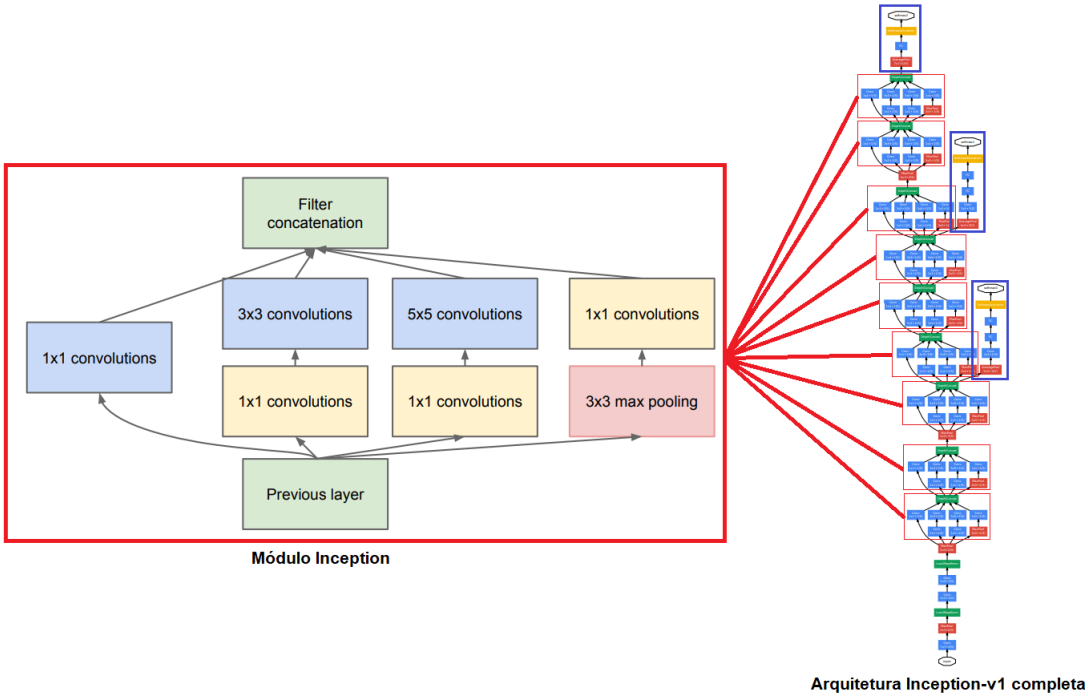


Figura 3.9: Módulo da rede Inception-v1 e arquitetura completa da rede. Adaptado de Szegedy *et al.* (2015).

A arquitetura da CNN é apresentada na Tabela 3.2. Estruturas com filtros 3x3 para redução da imagem e 1x1 para operações *zero padding* são utilizadas de forma recorrente, dobrando a quantidade de filtros a cada nova ocorrência. As camadas finais são para uma operação final de *pooling* médio global de toda a estrutura e em seguida uma camada totalmente conectada para classificação, utilizando o método *softmax*.

3.5.3. Classificação, Regressão e Otimização

As CNN podem ser utilizadas para resolver problemas de classificação ou regressão. Os problemas do primeiro tipo são tipicamente encontrar a qual classe pertence uma entrada (no caso deste trabalho, uma imagem) dentro de um conjunto de classes pré-definidas, sendo portanto um problema de natureza discreta. Os problemas de regressão consistem em encontrar uma função real que se aproxima ao máximo de outra, cujos coeficientes, ordens e operações são desconhecidos e só se conhece uma lista de pares entrada/saída.

Li *et al.* (2016) demonstram que a primeira etapa é definir uma função de pontuação que mapeia os valores dos pixels em uma imagem em uma escala de confiabilidade para cada classe comparada em uma biblioteca prévia, ou seja, a probabilidade de pertencimento a uma classe. Existem algumas implementações para esta função de pontuação para classificação e as mais simples e mais utilizadas para as tarefas de detecção de objetos são os classificadores lineares como *Softmax* ou *Support Vector Machine* (SVM),

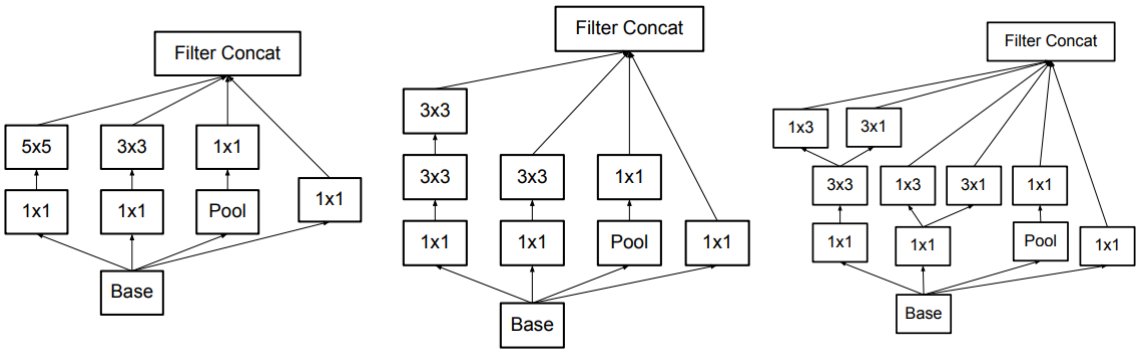


Figura 3.10: Mudança do módulo Inception da versão 1 para versão 2: na primeira Figura, o módulo original. Na segunda, o modelo equivalente, com operações em série substituindo filtros 5x5. Na terceira Figura, uma otimização do filtro 3x3, para dois filtros em paralelo 1x3 e 3x1 que tem melhor desempenho computacional. Adaptado de Szegedy *et al.* (2016).

que utilizam funções lineares para atribuírem pontuação para as imagens durante um treinamento.

Uma abordagem para *Softmax* descrita em Goodfellow *et al.* (2016) é utilizar um vetor de saída com as probabilidades de pertencimento a uma classe, de tal forma que a soma dos termos deste vetor seja igual a 1. O elemento com maior valor é o que melhor representa a classe associada a entrada.

Considerando uma rede com $i = 1,...,N$ camadas, k o número de classes da biblioteca, h o fator de ativação do último nó e W o peso resultante da última camada antes da primeira camada de classificação *Softmax*, o resultado de saída de cada camada de classificação será conforme a equação 3.1 (TANG, 2013):

$$a_i = \sum_k h_k W_{ki} \tag{3.1}$$

A probabilidade de pertencimento de uma entrada a uma classe é dada pela equação 3.2:

$$p_i = \frac{exp(a_i)}{\sum_j^N exp(a_j)} \tag{3.2}$$

A classe predita $\hat{i}$ pela rede então é dada pela equação 3.3:

$$\hat{i} = arg(max(p_i)) \tag{3.3}$$

Dada a função de classificação, o problema passa a ser a comparação do desempenho da classificação com a identificação verdadeira, medida por uma função de erro. O objetivo da CNN passa a ser minimizar este erro através de iterações em uma abordagem de otimização objetiva.

Tipo de camada/ módulo	**Tamanho do filtro/ passo da operação**	**Tamanho da entrada**
Convolução 1	3x3/2	299x299x3
Convolução 2	3x3/1	149x149x32
Convolução para redução	3x3/1	147x147x32
Pooling	3x3/2	147x147x64
Convolução 3	3x3/1	73x73x64
Convolução 4	3x3/2	73x73x80
Convolução 5	3x3/1	35x35x192
3x Inception	Conforme Figura 3.10b	35x35x288
5x Inception	Conforme Figura 3.10c	35x35x288
2x Inception	Conforme Figura 3.10c	8x8x1280
Pooling	8x8	8x8x2048
Linearização		1x1x2048
Softmax	Classificador	1x1x1000

Tabela 3.1: Lista de camadas da rede Inception-v2 em ordem de implementação da entrada para a saída, com o resultado da classificação (SZEGEDY *et al.*, 2016).

A classificação também envolve um problema de regressão, ao avaliar os valores da saída da CNN comparados à biblioteca de treinamento. A função de erro para esta tarefa é usualmente o erro médio quadrático, definido pela equação 3.4, onde y é a saída do modelo e t são os valores de referência obtidos na biblioteca de treinamento:

$$l(y,t) = ||y-t||^2 \tag{3.4}$$

As CNNs estudadas neste trabalho são destinadas a resolver um problema multi-objetivo que envolve a classificação e a regressão, além da tarefa de localização (com a predição das coordenadas que o objeto se encontra na imagem) a partir da abordagem da análise do Gradiente Descendente, com a utilização de derivadas em cada ponto para a predição da direção que a função de erro deve seguir para atingir seu ponto mínimo.

Neste trabalho serão estudadas as CNN do tipo SSD (Seção 3.5.5), R-CNN, Fast R-CNN e Faster R-CNN (Seção 3.5.6), YOLO (Seção 3.5.7), Mask R-CNN (Seção 3.5.8) que utilizam redes CNN para predição de localização e redes CNN para classificação. Os modelos implementados discutidos nos Capítulos 4 e 5 utilizam redes de classificação implementadas após as redes para localização, conforme a Figura 3.6.

3.5.4. Funções de ativação

A função de ativação é um nó de uma rede neural que é colocado ao fim de uma rede ou entre camadas da mesma. Sua utilidade é ser uma ferramenta de decisão para selecionar os dados mais representativos para seguir em frente no processo de treinamento de uma rede, dado o produto de uma entrada com os pesos da camada da rede em questão

	Tipo de camada	**Quantidade de filtros**	**Tamanho do filtro**	**Tamanho da entrada**
	Convolucional	32	3x3	256x256
	Convolucional	64	3x3/2	128x128
1x	Convolucional	32	1x1	
1x	Convolucional	64	3x3	
1x	Residual			64x64
	Convolucional	128	3x3/2	64x64
2x	Convolucional	64	1x1	
2x	Convolucional	128	3x3	
2x	Residual			64x64
	Convolucional	256	3x3/2	32x32
8x	Convolucional	128	1x1	
8x	Convolucional	256	3x3	
8x	Residual			32x32
	Convolucional	512	3x3/2	16x16
8x	Convolucional	256	1x1	
8x	Convolucional	512	3x3	
8x	Residual			16x16
	Convolucional	1024	3x3/2	8x8
4x	Convolucional	512	1x1	
4x	Convolucional	1024	3x3	
4x	Residual			8x8
	Avgpool		Global	
	Totalmente Conectada		1000	
	Softmax			

Tabela 3.2: Lista de camadas da rede Darknet-53 em ordem de implementação da entrada para a saída, com o resultado da classificação (REDMON e FARHADI, 2018).

(LI *et al.*, 2016).

As transformações não lineares realizadas pela função de ativação utilizam um o resultado do produto $entrada * peso$ como variável para alguma operação matemática fixa limitada a um domínio disponível. Existem algumas funções de ativação comumente utilizadas, conforme a Figura 3.11.

- **Sigmoid**: A função sigmoidal tem a forma $\sigma(x) = 1/(1 + e^{-x})$. A função pega um dado valor real x e o transporta para os limites da função sendo 0 (em caso de valores negativos grandes) ou 1 (em caso de números positivos grandes). Esta função apresenta uma grande desvantagem: quando o resultado do nó satura em 0 ou 1, o gradiente nestas regiões é próximo de 0. Na realimentação da rede neural do tipo *backpropagation* a multiplicação este gradiente é multiplicado pelo gradiente da saída da rede para o objetivo geral, praticamente zerando o resultado e levando ao final do treinamento, com um grande valor de erro. Outro ponto é que a função não é simétrica em 0 e é sempre positiva e isso causa um impacto na avaliação dos gra-

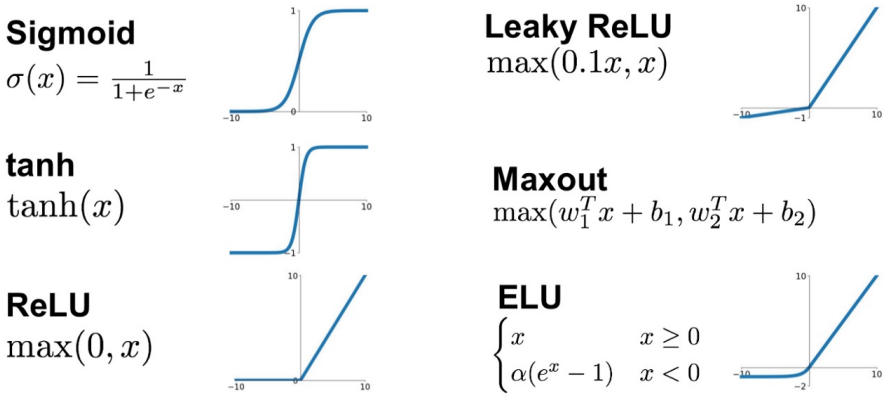


Figura 3.11: Funções de ativação (UDOFIA, 2018).

dientes durante a etapa de *backpropagation* sendo os mesmos sempre positivos para $x > 0$ ou negativos para $x < 0$, introduzindo uma grande oscilação na atualização dos pesos durante as iterações. (LI *et al.*, 2016).

- **tanh**: A função de tangente hiperbólica leva o valor da função para limites na faixa $[-1, 1]$. Como na função *sigmoid*, sua ativação satura, porém sua saída é centrada em torno do valor 0, o que elimina a questão da oscilação dos valores atualizados dos pesos nas camadas, conforme citado anteriormente.

- **ReLU**: a função, cujo nome é uma abreviação para *Rectified Linear Unit* (ou unidade linear retificada), computa a função $f(x) = max(0, x)$. Em outras palavras, a ativação é um corte para valores maiores do que 0 apresentados pela multiplicação dos valores da camada pelo seu peso. De acordo com o estudo apresentado por Krizhevsky *et al.* (2012) esta função apresenta uma aceleração até 6 vezes maior do que a função *tanh* ou *sigmoid* na convergência dos valores do gradiente, em função da sua forma linear e não de saturação conforme as demais funções. Também é uma função simples, que replica o resultado da entrada, ao contrário das anteriores que utiliza exponenciais ou funções trigonométricas que tem um valor de custo computacional mais elevado. Porém a função $ReLU$ pode acabar ativando poucos neurônios e causando o encerramento precoce do treinamento de uma rede neural. Isto pode ocorrer caso o gradiente de saída apresente um valor negativo e após o *backpropagation* o valor permanecerá sempre negativo e esta função sempre levará o resultado para 0, desativando o neurônio.

- **Leaky ReLU**: esta função é uma tentativa para contornar o problema apresentado pela função $ReLU$. Ao invés da função ser igual a 0 para $x < 0$, a função *Leaky* $ReLU$ tem uma pequena inclinação negativa (por exemplo, 0,1 ou menores) para valores nesta faixa.

- **ReLU6**: Uma variação da função $ReLU$ proposta por Krizhevsky e Hinton (2010).

Esta função é dada por $f(x) = min(max(x, 0), 6)$, saturando para valores de x maiores do que 6. Conforme a conclusão do autor, esta função leva ao aprendizado de características que estejam mais esparsas na imagem de forma mais rápida, uma vez que a função tem um limite de saturação definido ao invés de infinito comparado à $ReLU$ e assim provoca a convergência do gradiente no *backpropagation*.

- **Maxout**: esta função é uma generalização do caso apresentado pela função $ReLU$ e a versão *Leaky* $ReLU$. Esta função calcula o máximo entre duas funções lineares do tipo $w_1^T x + b$ e $w_2^T x + b$ onde w_1^T é a matriz de pesos associada a camada 1 e w_2^T para a camada 2 e b é um valor de constante. No exemplo, para $ReLU$, $w_1^T = 0, b = 0$ e $w_2^T = 1, b = 0$. Na função *Maxout* existem os beneficios da linearidade sem saturação e também do tratamento para valores menores do que 0, como na função *Leaky* $ReLU$. A desvantagem é o grande número de parâmetros para cada neurônio, levando a um grande número geral de operações necessárias para toda a rede.

- **ELU**: a função *exponential linear units* (ou unidades lineares exponenciais) tem a característica de acelerar o treinamento de redes neurais comparado a $ReLU$ e suas variações. O fator exponencial para valores menores do que 0 torna a convergência nesta faixa de valores mais rápida do que a *Leaky* $ReLU$ e sem o problema de zerar o neurônio conforme a $ReLU$ apresenta. Uma característica negativa da ELU é que ela não é centralizada em torno de 0 e apresenta o mesmo problema da função *sigmoid*, causando oscilação na atualização dos pesos da camada durante as iterações. Para corrigir este problema, existem variações desta função com escalares parametrizáveis para tornar a função centralizada em 0 (NWANKPA *et al.*, 2018).

Conforme o texto de Nwankpa *et al.* (2018), as funções de ativação são um componente chave para o treinamento e otimização de redes neurais, implementadas em diferentes camadas de arquiteturas de *deep learning* e é utilizada em diversos tipos de resolução de problemas como processamento de linguagens, detecção de objetos, classificação e segmentação de imagens entre outros.

Nwankpa *et al.* (2018) realizam um estudo comparativo entre as implementações de redes neurais para detecção e classificação de objetos e quais funções de ativação foram utilizadas. Na grande maioria dos casos a função $ReLU$ apresentou desempenho superior conforme comparação entre algoritmos e também das implementações vencedoras de competições de desafios de reconhecimento de imagens como o *Image Large Scale Visual Recognition Challenge* (ILSVRC), que disponibiliza uma grande biblioteca de imagens catalogadas em classes conhecida como *ImageNet*. A principal razão é a sua simples implementação computacional e baixo custo de processamento o que leva a um treinamento com maior eficiência.

3.5.5. *Single Shot Detector* (SSD) - Detector por única imagem

O método de detecção de objetos *Single Shot Detector* (SSD) é baseado em criar hipóteses de caixas de detecção de tamanhos diferentes para percorrer uma determinada imagem, realizar amostras de *pixels* desta caixa de detecção e aplicar um classificador de imagens na seleção realizada, da mesma forma que outros métodos *estado-da-arte* para detecção de objetos, como YOLO (Seção 3.5.7) ou R-CNN (Seção 3.5.6). A característica diferencial do SSD é encontrar objetos similares ao de uma classe pesquisada em toda extensão da imagem fornecida em uma única iteração (LIU *et al.*, 2016).

O modelo SSD tem dois componentes típicos: uma camada de classificações de imagens, que utiliza uma rede neural pré-treinada e outra camada específica com o modelo SSD que faz as operações de detecção de objetos através de redes neurais convolucionais. Na Figura 3.12 as camadas representadas pelas caixas brancas representam as camadas de redes neurais com modelos para classificação de imagens e as caixas azuis representam as camadas de operações para detecção de objetos.

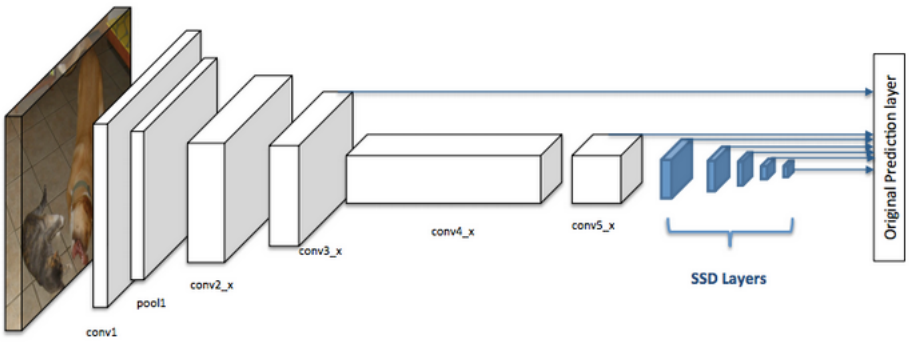


Figura 3.12: Arquitetura de uma rede neural convolucional com um detector de objetos do tipo SSD. Adaptado de Liu *et al.* (2016).

O método SSD divide a imagem em diversos setores formando uma matriz de regiões para análise. Após o estabelecimento desta matriz, o algoritmo de classificação é executado e naquelas regiões que existe uma probabilidade de ocorrência de um objeto, são encaminhadas para a execução do algoritmo de detecção de objetos SSD, com uma área de pesquisa menor. Na fase de detecção de objetos são utilizadas as "caixas-âncora"(ou *anchor boxes*) são aquelas que mais se adequam ao tamanho do objeto em análise. As caixas-âncora tem tamanhos variáveis que são definidas durante a fase de treinamento, com relação largura x altura variáveis, escalas e localizações nas imagens, variando ampliação e redução das mesmas (*"aspect ratios"*). Durante o treinamento são realizadas identificações dos objetos em várias destas escalas, com o objetivo de percorrer toda a imagem e detectar o objeto em diferentes regiões e profundidades. Conforme descrito em Huang *et al.* (2017), durante o treinamento, para uma âncora a que melhor se relaciona a uma área delimitada em uma imagem como identificação positiva de um objeto b, esta é

separada para um conjunto de âncoras de identificação positiva. Em caso negativo, esta é armazenada em um conjunto de âncoras de identificação negativa.

Õ artigo de Liu *et al.* (2016) descreve a função de perda para o modelo de identificação é determinado pela equação 3.5 que é a soma ponderada da função de função de perda por localização L_{loc} com a função de perda de confiabilidade L_{conf}. Sendo x um indicador em que ocorreu uma identificação de uma caixa de tamanho padronizado à uma imagem definida como verdadeira para um dado objeto.

$$L(x, c, l, g) = \frac{1}{N}(L_{conf}(x, c) + \alpha L_{loc}(x, l, g)) \tag{3.5}$$

Nesta equação, N é o número de caixas com identificação correta positiva. Se $N = 0$, a perda é igual a 0. A função de localização é uma suavização da linearização de primeira ordem entre a diferença entre a caixa de utilizada para predição l e a caixa que determina um objeto identificado. A função de confiabilidade é uma relação entre as classes de identificação utilizadas no treinamento.

3.5.6. *Region Convolutional Neural Network* - (R-CNN) - Rede Neural Convolucional por Região e derivações

Para os métodos para detecção de objetos em estudo neste trabalho, foram selecionados aqueles baseados no conjunto classificação e localização. Ou seja, é necessário identificar inicialmente na imagem analisada uma imagem semelhante às comparadas à bibliotecas pré-estabelecidas, ou seja, à classes. Isto é feito através da detecção por um conjunto de características (ou *features*) como o HOG e classificação por redes como máquinas de vetores-suporte (*Support Vector Machines* - SVM), redes neurais convolucionais (CNN) ou similares (REN *et al.*, 2015).

O sucesso portanto depende da eficiência do processo de detecção para posteriormente encaminhar para etapa de classificação. A detecção utilizando HOG não é tão eficiente quanto as comparadas utilizando CNN. Mas o custo computacional da utilização de CNN é alto, uma vez que é necessário executar o algoritmo para cada extrato de imagem, na técnica de "janelas deslizantes"de busca, discutida na Seção 3.5.5. O método de rede neural convolucional por região (*Region Convolutional Neural Network* - R-CNN), proposto por Girshick *et al.* (2014), trata este problema através de um algoritmo específico chamado busca seletiva, que reduz o número de caixas de detecção. Este algoritmo utiliza características como texturas, intensidade, cores para gerar as regiões candidatas para a classificação.

Inicialmente são geradas regiões de interesse (*Regions of Interest* - RoI) através de um método de busca seletiva proposto por Uijlings *et al.* (2013). O método consiste em realizar segmentação semântica da imagem e assim determinar áreas com possíveis objetos para identificação. No R-CNN são geradas aproximadamente 2000 RoI para cada

imagem para serem analisadas na próxima etapa, conforme implementação de Girshick *et al.* (2014). Em seguida as RoI são encaminhadas para uma CNN para realizar a extração de *features*. Com o mapa de *features* construído, este é encaminhado para a classificação em uma rede do tipo SVM e para a geração de *bounding boxes* para localização dos objetos (Figura 3.13).

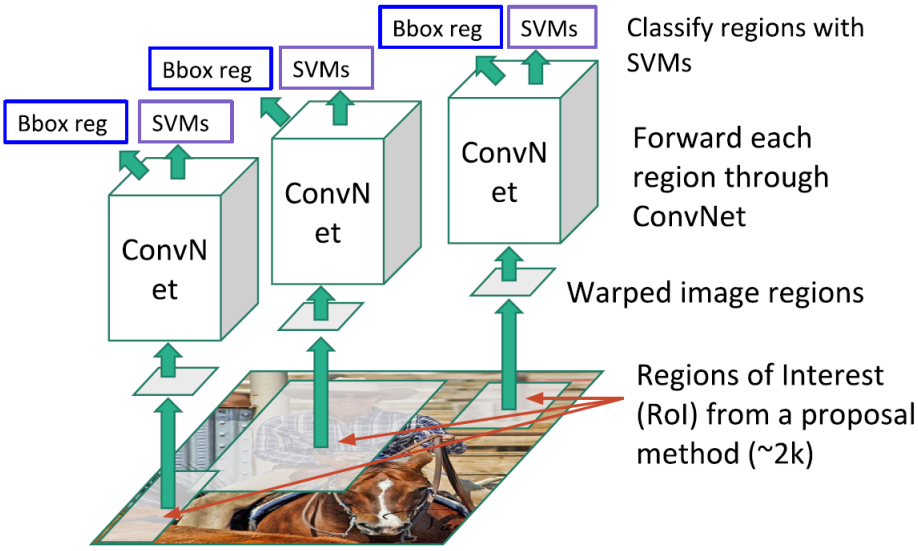


Figura 3.13: Rede neural modelo *R-CNN* (LI *et al.*, 2016).

A R-CNN tem problemas como custo computacional excessivo (geração de 2000 RoI para análise para cada imagem) e grande tempo necessário para treinamento. A detecção utilizando as redes já treinadas também gasta um longo tempo para detecção (GIRSHICK, 2015).

O mesmo autor propôs uma melhoria do método, a Fast R-CNN. Em Girshick (2015), o autor substitui utiliza a etapa de geração de RoI pela busca seletiva e passa todas as RoI geradas por uma rede convolucional específica para criação de um mapa de *features* ao contrário de passar cada RoI em uma rede convolucional separadamente. Em seguida este mapa com os RoI propostos é encaminhado para camadas convolucionais com filtros mais específicos para a detecção de objetos e uma matriz de mapas de características com os objetos devidamente detectados é encaminhada para etapas de classificação e geração de *bounding boxes* para localização (Figura 3.14). Este método é mais rápido na comparação ao R-CNN em aproximadamente 10 vezes nas etapas de treinamento e detecção e também apresenta maior precisão (GIRSHICK, 2015).

Ren *et al.* (2015) propõem em seu artigo uma adaptação das R-CNN otimizada para melhorias de desempenho computacional. Chamado Faster R-CNN, este método é composto de dois módulos. O primeiro é uma rede convolucional para proposição de regiões, que elimina a busca seletiva com alto custo computacional, e o segundo é um classificador de imagens que utiliza as regiões selecionadas na primeira etapa. A primeira rede de proposição de região orienta a segunda rede em qual local da imagem iniciar a

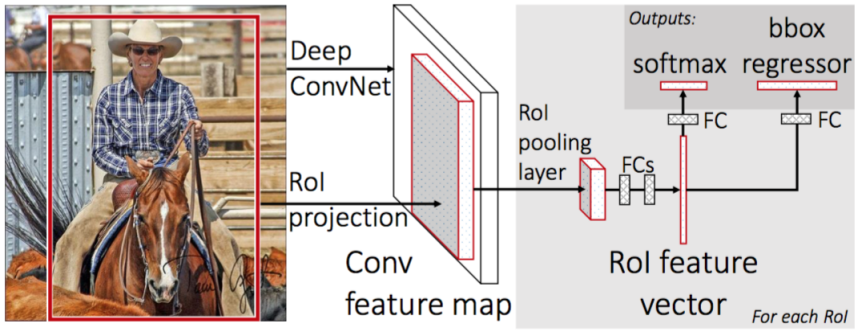


Figura 3.14: Rede neural modelo *Fast R-CNN* (GIRSHICK, 2015).

busca (Figura 3.15).

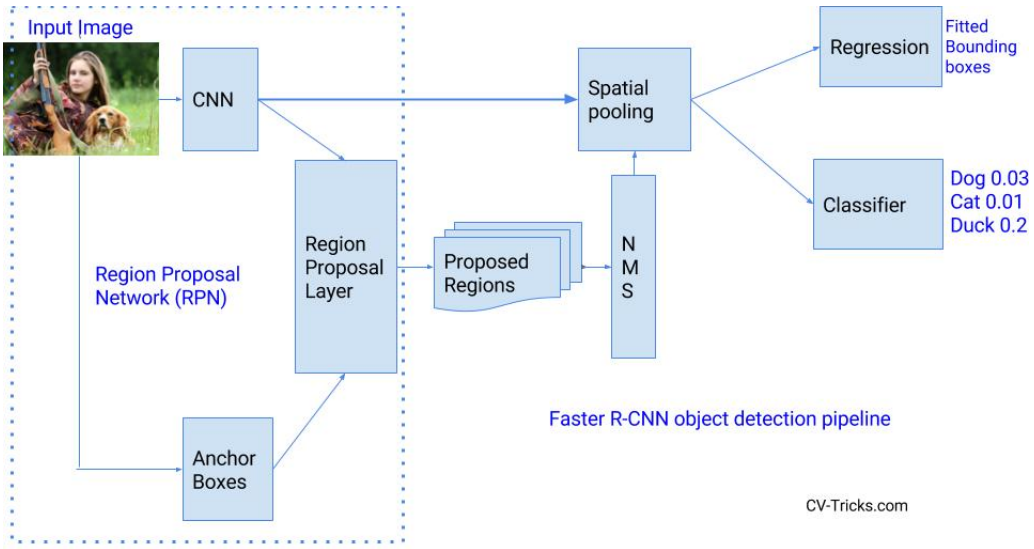


Figura 3.15: Rede neural modelo *Faster R-CNN*

A função de perda durante a fase de treinamento é definida pela equação 3.6. O modelo Faster R-CNN avalia a identificação entre cada "âncora"e uma imagem identificada como verdadeira na biblioteca com o maior resultado possível para a interseção sobre a união (IoU) entre a âncora e a imagem verdadeira. A função também considera resultados de IoU acima de 0,7. A segunda condição já seria suficiente para determinar amostras positivas, porém pode resultar em nenhuma amostra selecionada, para casos onde o treinamento tem resultados com índices baixos de confiabilidade e detecção. Detecções com valores do IoU menores do que 0,3 contribuem com valor negativo na função. Âncoras cuja identificação estiver na faixa entre 0,3 e 0,7 não contribuem na função de perda objetiva.

$$L(p_i, t_i) = \frac{1}{N_{cls}} \sum_i L_{cls}(p_i, p_i^*) + \lambda \frac{1}{N_{reg}} \sum_i p_i^* L_{reg}(t_i, t_i^*) \tag{3.6}$$

Na equação 3.6 o índice i é referente ao índice de uma âncora de identificação e

p_i é a probabilidade da âncora ser um objeto identificado positivamente. O parâmetro p_i^* é referente à imagem com identificação positiva da biblioteca, sendo 1 para identificação positiva em relação à âncora e 0 com identificação negativa. Já t_i é um vetor com as 4 coordenadas (x,y) da caixa de detecção proposta. t_i^* é um vetor com as coordenadas do objeto na imagem com identificação positiva na biblioteca utilizada no treinamento. A função de perda na classificação L_{cls} é uma função binária entre identificação positiva (1) ou negativa (0) das classes do objetos registrados na biblioteca de treinamento. A função de perda na regressão L_{reg} é uma suavização da linearização em primeira ordem da função $t_i - t_i^*$, como definido em Girshick (2015). O termo $p_i^* L_{reg}$ significa que a função de perda na regressão somente é ativada para âncoras positivas ($p_i^* = 1$). Os dois termos da equação são normalizados por N_{cls} e N_{reg} e ponderados pelo fator λ. N_{cls} representa o tamanho da batelada de operações para classificação e N_{reg} o número definido para localizações propostas pelo modelo de predição inicial. Segundo Ren *et al.* (2015), o fator λ não afeta os resultados em uma faixa mais ampla de iterações. Neste artigo é recomendado utilizar um fator que mantenha os pesos das duas parcelas aproximadamente igual. Este valor é obtido empiricamente.

3.5.7. *You Only Look Once* - (YOLO) - Você olha somente uma vez

Redmon *et al.* (2016) propõem em seu artigo um método para identificação de objetos em uma nova abordagem, em comparação aos modelos SSD, R-CNN e Faster R-CNN. Ao invés de realizar inicialmente a classificação e depois a localização do objeto, no novo método a etapa de predição de caixas de identificação é realizada através de uma rede neural convolucional na imagem em toda a sua extensão, em uma única iteração. O objetivo é otimizar o custo computacional e o tempo para identificação de objetos. O método denominado *YOLO* (*You only look once* - "Você olha somente uma vez"em tradução livre)

O método YOLO divide a imagem em uma matriz S x S e para cada célula é executada uma operação com uma rede neural convolucional para predição de N caixas de detecção naquele pedaço de imagem. A confiabilidade da predição reflete a precisão da caixa de detecção e quando uma caixa de detecção contém um objeto ou não, independente do número de classes utilizadas na biblioteca de treinamento, ou seja, sem classificar este extrato da imagem. É realizada também uma operação para prever uma pontuação de classificação nestas caixas de detecção (Figura 3.16).

A diferença entre os métodos SSD, R-CNN e também Faster R-CNN é que o método YOLO realiza apenas uma iteração para uma rede neural convolucional para classificação e outra para predição de caixas de detecção. Os ganhos em desempenho são expressivos conforme citado em Huang *et al.* (2017). Porém, pelo fato do método

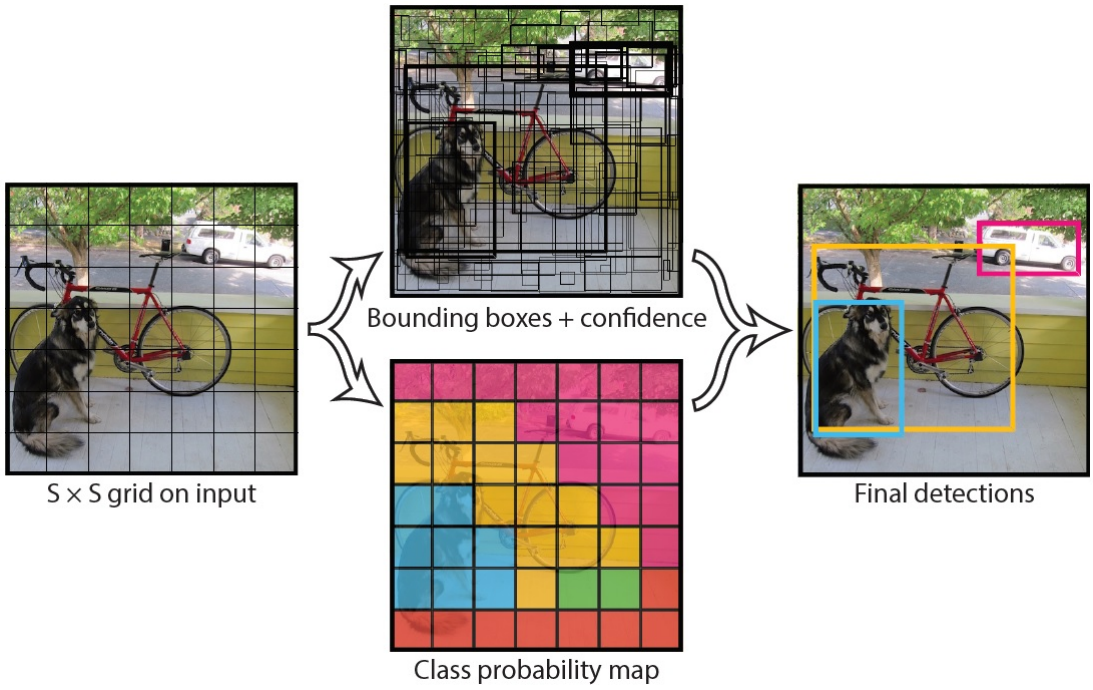


Figura 3.16: Modelo YOLO para detecção de objetos. (REDMON *et al.*, 2016)

original não utilizar a técnica de variação de escalas da imagem para classificar e detectar objetos em profundidades e tamanhos variados, para objetos em escala menor nas imagens o método YOLO tem desempenho inferior aos demais métodos comparados (REDMON *et al.*, 2016). Na Figura 3.17 é exibida a arquitetura padrão do método YOLO, com as camadas da rede convolucional para classificação e localização do objeto.

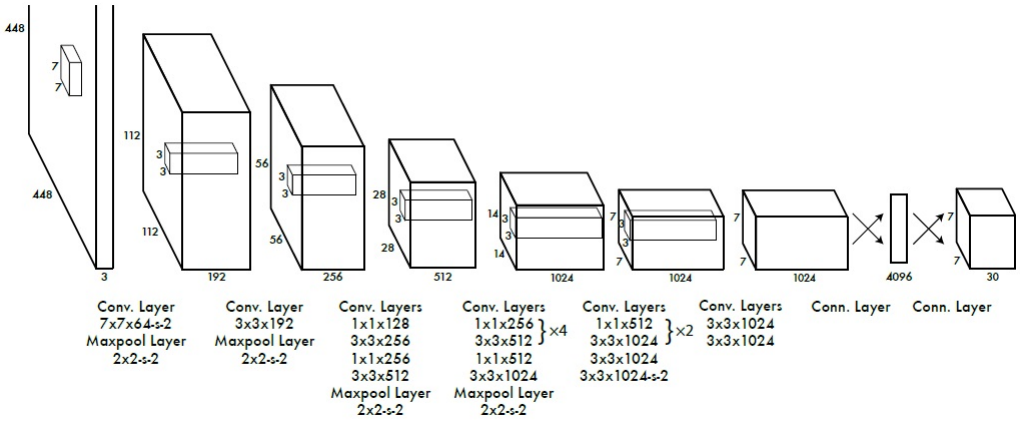


Figura 3.17: Arquitetura da rede neural convolucional para classificação e localização de imagens do método YOLO. (REDMON *et al.*, 2016)

Essencialmente o método YOLO versão 3 (YOLOv3), proposto no artigo de Redmon e Farhadi (2018), processa uma imagem de dimensões pré-definida em *pixels* e a coloca como entrada de uma rede treinada de modelo tipo *Darknet-53* e produz detecções em três escalas. A cada escala, o número de detecções é do formato enumarado a seguir

e representada na Figura 3.18 (CHIAN, 2019):

1. tamanho da batelada (*batch size*): tamanho da quantidade de imagens processada a cada iteração na rede.

2. número de "caixas-âncora"ou número de caixas estabelecidas para detecção (*num of anchor boxes*).

3. número de divisões da matriz estabelecida na imagem na dimensão "X"(grid size).

4. número de divisões da matriz estabelecida na imagem na dimensão "Y"(grid size).

5. Fator multiplicador referentes a parâmetros do treinamento prévio da rede em um conjunto de imagens. A composição deste multiplicador é feita de 4 para as coordenadas das caixas de detecção, 1 para um índice de confidência e mais a quantidade de classes utilizadas para detecção. O conjunto de imagens *COCO* tem 80 classes e esta foi utilizada no treinamento desta rede. O fator multiplicador total, portanto é igual a 85.

O produto do número de caixas para detecção pelo fator multiplicador representa a dimensão de uma matriz de características utilizadas para detecção. Como exemplo, para 3 caixas de detecção e fator igual a 85, a matriz de características tem tamanho igual a 255 (KATURIA, 2018).

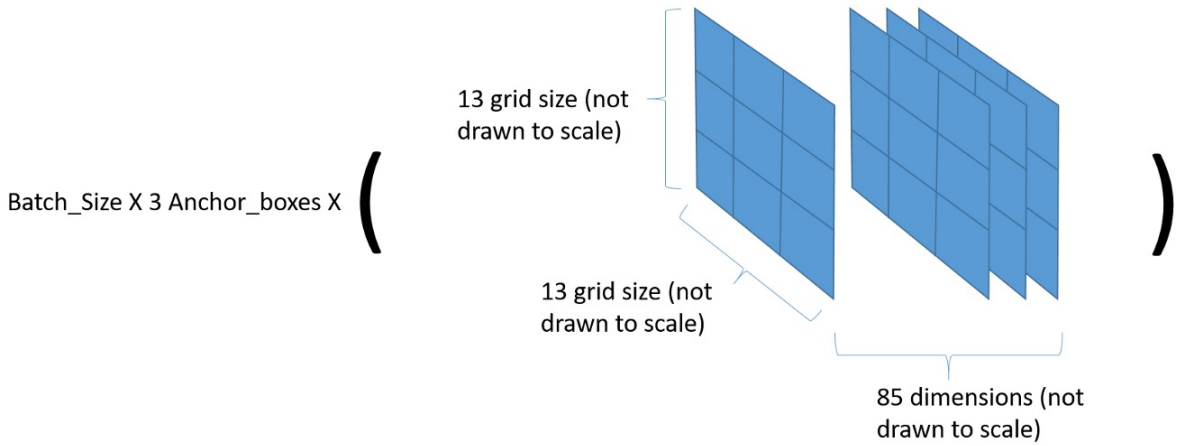


Figura 3.18: Arquitetura da rede YOLOv3 para processamento da imagem. (CHIAN, 2019)

O modelo YOLOv3 realiza as predições em 3 escalas, que são feitas reduzindo a resolução da imagem de entrada pelos fatores de 32, 16 e 8. A primeira detecção é realizada na camada de número igual a 82. Nas primeiras 81 camadas a imagem é reduzida de tamanho. Como exemplo, para uma imagem de 416x416 *pixels*, as reduções pelo fator de 32 resultam em uma imagem de 13x13 *pixels*. Portanto, o mapa de características

utilizado para detecção tem dimensões de 13x13x255. As etapas seguintes, com uma ampliação pelo fator de 2 e em seguida por um fator de 2 novamente e em seguida operações convolucionais, formam um mapa final de características para detecção de dimensões iguais a 52x52x255 *pixels*. Portanto o mapa de detecção é descrito pela equação $S * S * (B * (5 + C))$, onde S é a dimensão final da imagem após reduções e ampliações, B é o número de caixas de detecção aplicadas, 5 é um fator descrito anteriormente como o as coordenadas das caixas de detecção mais um fator de confidência e C é o número de classes utilizadas para treinamento do modelo, diante de um conjunto de imagens (KATURIA, 2018).

Utilizando a mesma notação da equação do mapa de descrição de objetos descrita no parágrafo anterior, a função de perda do modelo YOLOv3 é composta pela Equação 3.7.

$$\begin{aligned} L = \lambda_{coord} \sum_{i=0}^{S^2} \sum_{j=0}^{B} 1_{ij}^{obj} [(x_i - \hat{x}_i)^2 + (y_i - \hat{y}_i)^2] \\ + \lambda_{coord} \sum_{i=0}^{S^2} \sum_{j=0}^{B} 1_{ij}^{obj} [(\sqrt{w_i} - \sqrt{\hat{w}_i})^2 + (\sqrt{h_i} - \sqrt{\hat{h}_i})^2] \\ + \sum_{i=0}^{S^2} \sum_{j=0}^{B} 1_{ij}^{obj} (C_i - \hat{C}_i)^2 + \lambda_{noobj} \sum_{i=0}^{S^2} \sum_{j=0}^{B} 1_{ij}^{noobj} (C_i - \hat{C}_i)^2 \\ + \sum_{i=0}^{S^2} 1_i^{obj} \sum_{c \epsilon classes} (p_i(c) - \hat{p}_i(c))^2 \end{aligned} \tag{3.7}$$

Conforme descrito em Redmon *et al.* (2016), os termos da equação são compostos por:

- x_i, y_i, que são as coordenadas do ponto centróide da caixa de detecção,
- w_i, h_i, são as distâncias da largura (w) e altura (h) da caixa de detecção,
- C_i que representa uma pontuação de confiança se há um objeto identificado ou não em um determinado quadrante analisado,
- $p_i(c)$, que é a função de perda na classificação,
- 1_{ij}^{obj} e 1_{ij}^{noobj} são fatores penalizadores da função de perda. Ou seja, quando há objeto identificado na célula, o fator 1_{ij}^{obj} é igual a 1 e quando não há é igual a 0. O fator 1_{ij}^{noobj} é o oposto, recebendo valor igual a 0 quando nenhum objeto é reconhecido e 1 quando não há detecção,
- 1_i^{obj} é igual a 1 quando ocorre uma detecção de uma determinada classe i dentro do conjunto de classes da biblioteca de treinamento.

Todos os fatores de perda são calculados através de erros médios quadráticos, com a diferença entre um valor de referência de uma caixa determinada como detecção verdadeira na biblioteca de treinamento e o valor encontrado a cada iteração. Os valores com um chapéu sobre o termo ($\hat{.}$) são os reais lidos da referência de coordenadas passadas como áreas de detecção verdadeira na biblioteca de imagens. Os valores sem o chapéu são os valores previstos pela rede.

É necessário realizar este cálculo para caixa de detecção, sendo que o termo $\sum_{j=0}^{B} 1_{ij}^{obj}$ representa isto. Para cada célula da detecção é necessário calcular a função de perda, sendo o somatório $\sum_{i=0}^{S^2}$ o responsável por esta iteração. Os multiplicadores λ são constantes e utilizados para atribuir pesos aos termos da equação. Como o objetivo da função de perda é ser minimizada após as iterações da rede e o objetivo geral do modelo é ser eficiente em detecção de objetos, o peso λ_{coord} tem maior valor associado para valorizar os termos associados às coordenadas das caixas de detecção.

3.5.8. Mask R-CNN

Conforme apresentado na Seção 3.5, uma alternativa para o problema de detecção de objetos (classificação e localização de várias instâncias em uma imagem), é utilizar a segmentação como ferramenta para interpretação de pixels afim de delimitar as fronteiras das regiões referentes a um determinado objeto. A Seção 3.2 traz os conceitos sobre como a segmentação pode ser utilizada para identificação, classificação e localização de objetos. Conforme a Figura 3.2, algumas redes convolucionais foram desenvolvidas para tratar o problema de segmentação semântica e também detecção de objetos utilizando segmentação.

He *et al.* (2017) apresentam em seu trabalho a rede *Mask R-CNN* que utilizam a segmentação para extração de características de objetos na imagem. Conforme descrito na Seção 3.5.6, a rede *Faster R-CNN* tem duas saídas para cada candidato a objeto detectado: um rótulo de classe de objetos que aquela imagem apresenta uma maior probabilidade de pertencer e também as coordenadas para uma *bounding box* que determinam a localização deste objeto na imagem. Os autores do artigo sobre a *Mask R-CNN* propõem adicionar uma saída adicional com a máscara que determine a região dos pixels que representam o objeto detectado. Esta característica torna o método *Mask R-CNN* como ideal para obter informações mais detalhadas do objeto como a forma, perímetro e áreas mais aproximadas do que os detectados através dos retângulos propostos pelas *bounding boxes*.

A rede *Faster R-CNN* consiste em dois estágios. O primeiro é chamado de *Region Proposal Network - RPN* (rede de proposição de região), que delimita uma área maior, candidata a ter um objeto para detecção. O segundo estágio extrai as *features* de cada região candidata e executa a classificação e determinação da *bounding box* melhor ajustada ao tamanho do objeto detectado (HE *et al.*, 2017).

A extração de *features* utiliza a técnica denominada *Region of Interest Pooling (RoIPooling)* proposta e descrita no artigo de Girshick (2015). É uma camada de rede neural e que tem duas entradas: um mapa de *features* de tamanho fixo obtido de uma CNN e uma matriz com uma lista de regiões de interesse obtida na etapa da rede RPN. Esta rede relaciona cada região de interesse com a área equivalente no mapa de *features* e gera uma série de *bounding boxes* de acordo com a comparação com o mapa de *features*.

Na rede *Mask R-CNN* ao contrário de utilizar a técnica de *RoIPooling* é utilizado o conceito de *RoIAlign*. O mapa de *features* é uma sequência de filtros aplicados na imagem que geram resultados de correlação para cada filtro. No *RoIPooling* a combinação da pontuação em cada um destes filtros gera uma pontuação geral sobre quão pertinente uma região de interesse é semelhante ao mapa. No *RoIAlign* estas pontuações em cada camada do filtro são "alinhadas", ou seja, cada resultado em cada característica é armazenado em memória. O resultado prático é que tem-se a informação de forma do objeto após a combinação dos filtros do mapa de *features*, que é o equivalente a uma máscara gerada em uma segmentação semântica, ou seja, uma máscara com um rótulo específico de uma classe associada. O esquema da rede completa do modelo *Mask R-CNN* é exibido na Figura 3.19. O resultado final da detecção de objetos e máscaras de segmentação é exibido na Figura 3.20.

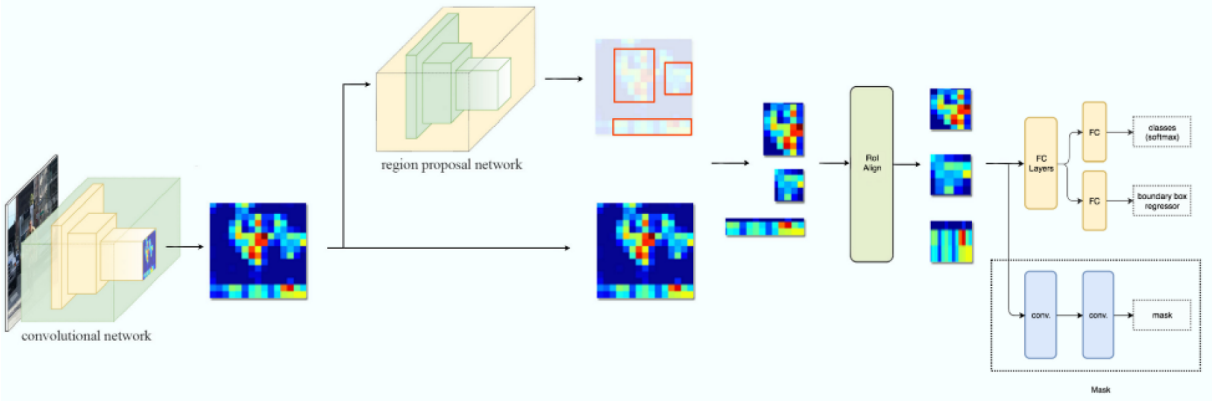


Figura 3.19: Arquitetura da rede *Mask R-CNN* com as camadas *RoIAlign* (GIRSHICK, 2015).

3.5.9. U-Net

As redes estudadas anteriormente são baseadas em proposição de caixas de detecção (*bounding boxes*) ou ainda na avaliação de uma função de similaridade em extratos da imagem à biblioteca de treinamentos para a tarefa de detecção e para a tarefa de classificação utilizam camadas de redes convolucionais especializadas em analisar um mapa de *features* relacionando a uma biblioteca de classes utilizada em treinamento. A rede *U-Net* proposta inicialmente por Ronneberger *et al.* (2015) é uma rede do tipo FCN (*Fully Connected Network*) e se dedica à tarefa de classificação e também de segmentação semântica,

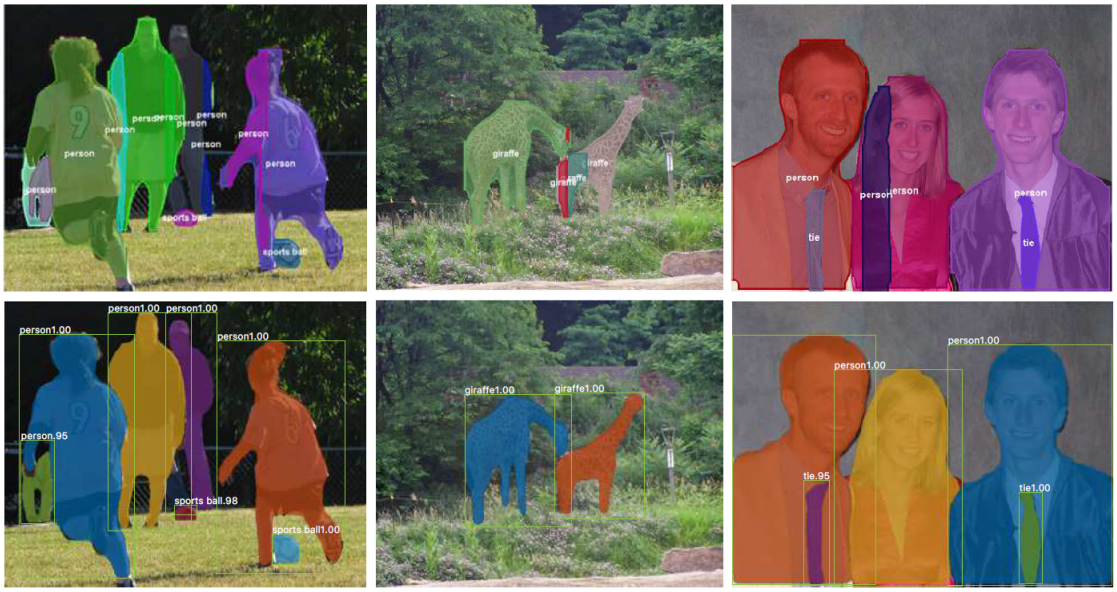


Figura 3.20: Resultados da rede *Mask R-CNN* com as máscaras de segmentação e as caixas de detecção de objetos (GIRSHICK, 2015).

ou seja, de determinação de máscaras que referenciem regiões equivalentes a objetos na imagem e a associação destas máscaras a classes de objetos armazenados inicialmente em biblioteca para treinamento.

A rede U-Net não só realiza a interpretação do mapa de *features* em classes como também necessita reconstituir a partir deste mapa os objetos detectados para formar as máscaras de detecção. Para utilizar a rede U-Net com o objetivo de detecção de objetos ou segmentação de instâncias, é necessário algoritmos auxiliares para realizar a diferenciação entre áreas identificadas e delimitação em *bounding boxes*.

A arquitetura da U-Net consiste em etapas de contração da imagem (*downsampling*) para obtenção de informações de características da imagem e contorno e camadas de expansão da imagem (*upsampling*) para determinação de localização (RONNEBERGER *et al.*, 2015).

A arquitetura da rede é ilustrada na Figura 3.21. O lado esquerdo consiste no caminho de contração da imagem e o lado direito no caminho de expansão. A primeira metade é uma arquitetura típica de uma rede convolucional que consiste na repetida aplicação de duas convoluções de matrizes 3x3 pixels, ao longo da extensão da imagem. Os resultados passam por funções de ativação do tipo ReLU e uma operação de *max-pooling* para obtenção de valores máximos com um passo (*stride*) de 2. O objetivo é coletar os valores mais representativos e realizar a redução da imagem (*downsampling*). A cada passo de *downsampling*, o número de canais de *features* é dobrado, ou seja, mais camadas de filtros para extração de características são adicionados. Tipicamente o formato da imagem para entrada é de 512x512 pixels e é reduzida até o formato de 32x32. A cada *downsampling* existe uma etapa correspondente no caminho de expansão da imagem e

os resultados da etapa anterior são a alimentação da etapa seguinte. Cada passo do caminho de expansão consiste na interpolação da imagem através de uma convolução de uma matriz 2x2 com mapa de *features* gerando valores a serem anexados no vetor anterior. A cada etapa o número de canais do mapa de *features* é reduzido pela metade. Também ocorre uma operação de concatenação do mapa de *features* da etapa correspondente no *downsampling* e duas operações de convolução 3x3, seguidos de uma operação com a função de transferência ReLU. A concatenação é necessária devido ao fato da perda dos pixels da borda em cada convolução (e a consequente perda de informação de localização). Na camada final, uma convolução com uma matriz 1x1 é realizada para mapear cada componente de um vetor de 64 posições referente às classes utilizadas para detecção. No total a rede tem 23 camadas convolucionais (RONNEBERGER *et al.*, 2015).

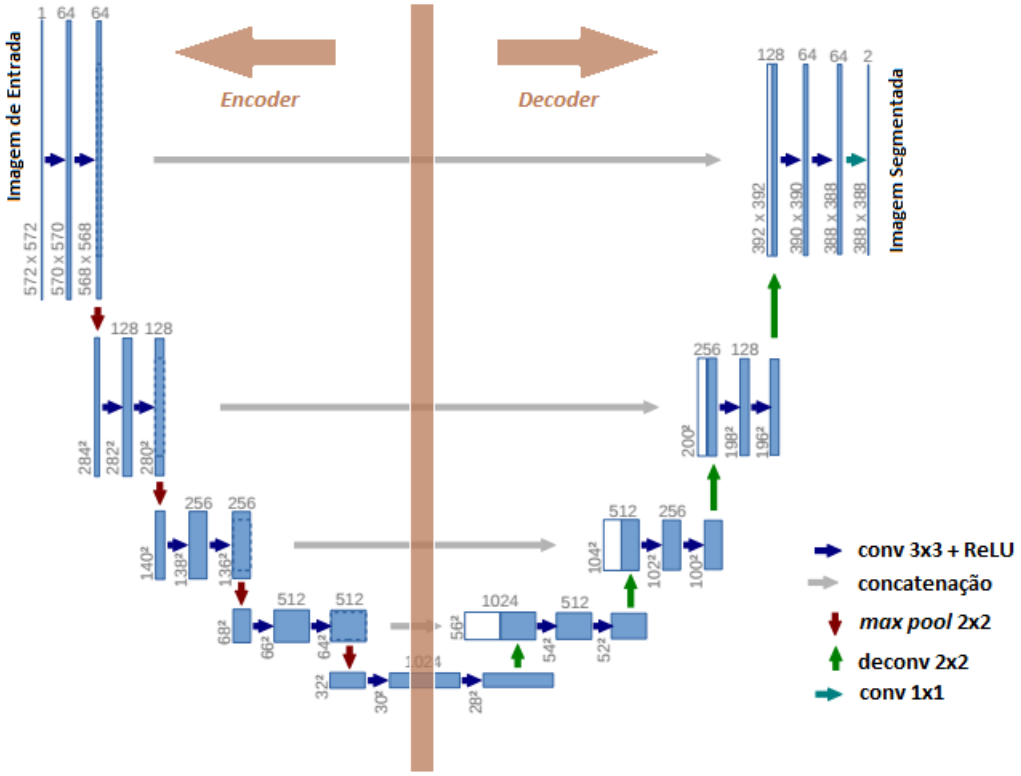


Figura 3.21: Arquitetura da rede U-net. Adaptado de Ronneberger *et al.* (2015).

3.6. Métodos de avaliação de desempenho de modelos de classificação, detecção de objetos e segmentação

Para validar e testar os modelos de reconhecimento de padrões, classificação, detecção de objetos e segmentação é necessário uma metodologia adequada relacionada à

visão computacional e aos parâmetros associados a qualidade da saída.

Tharwat (2020) em seu artigo cita que de acordo com o número de classes utilizadas nos métodos de classificação, existem dois tipos de avaliação: os problemas de classificação binária (uma classe de objeto e o fundo da imagem, por exemplo) e os de várias classes para identificação e as métricas de avaliação dependem do tipo de problema de classificação. A Figura 3.22 representa os conceitos de identificações Positivo Verdadeiro (*True Positive - TP*), Positivo Falso (*False Positive - FP*), Negativo Verdadeiro (*True Negative - TN*) e Negativo Falso (*False Negative - FN*).

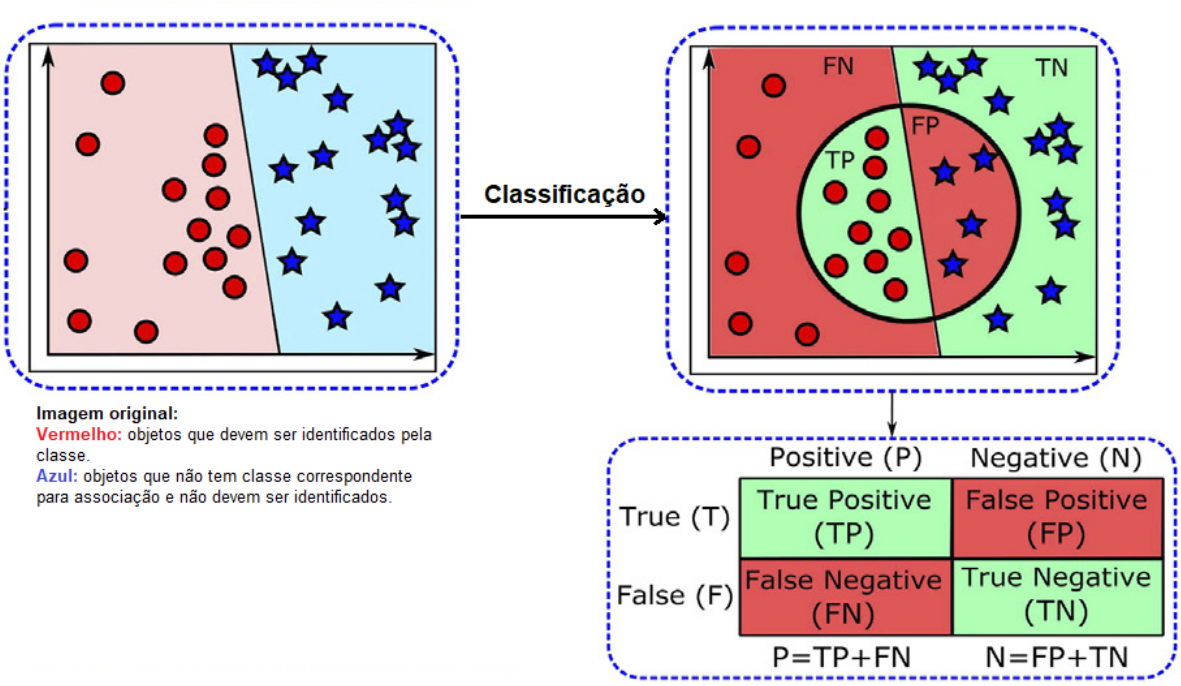


Figura 3.22: Tipos de identificações em modelos de classificação. Adaptado de Tharwat (2020).

A acurácia (A) é uma das medidas mais comumente utilizadas para classificação de desempenho e é definida como a razão entre as amostras corretamente classificadas pelo número total de amostras, conforme a equação 3.8.

$$A = \frac{TP + TN}{TP + TN + FP + FN} \tag{3.8}$$

A sensitividade ou Taxa de positivos verdadeiros (TPR) ou *recall* de um classificador representa as amostras positivas corretamente cadastradas em relação ao número total de amostras positivas (P = TP + FN), conforme a equação 3.9. A especificidade ou Taxa de negativos verdadeiros (TNR) ou *inverse recall* é expressa pela razão das amostras negativas verdadeiras pelo número total de amostras negativas (N = FP + TN) expresso pela equação 3.10. A precisão (P) é a razão entre as amostras verdadeiras positivamente

identificadas sobre a some das identificações(verdadeiras e falsas) conforme a equação 3.11. Assim o *recall* representa a proporção de amostras positivas corretamente classificadas e também é uma medida de precisão (THARWAT, 2020). O acompanhamento destes indicadores durante a fase de treinamento demonstra se o modelo está convergindo para uma resultados eficientes e servem para modelos que utilizam tanto caixas de detecção quanto segmentação, por se tratarem da análise de identificações, sejam de objetos ou de máscaras de segmentação.

$$\text{recall} = \frac{TP}{TP + FN} \tag{3.9}$$

$$\text{inverse recall} = \frac{TN}{FP + TN} \tag{3.10}$$

$$\text{precisão (P)} = \frac{TP}{TP + FP} \tag{3.11}$$

Outra métrica utilizada é a Interseção sobre União (*Intersection over Union - IoU*) que é uma estatística para estimar a similaridade entre dois conjuntos de amostras: uma *bounding box* produzida por uma predição e a sua relação com a caixa que delimita um objeto real, conforme a Figura 3.23(a). O indicador é a razão entre a área de interseção entre as duas caixas e a união entre as duas áreas. A qualidade do indicador é demonstrada na Figura 3.23(b) em que quanto maior a interseção entre as duas caixas e quanto menor for a área da união entre as mesmas representa maior precisão na detecção.O mesmo conceito pode ser aplicado na segmentação, onde ao contrário de se analisar as caixas de detecção, são comparadas as máscaras que determinam a região real de um objeto e a máscara gerada após o modelo realizar a predição desta área (VON WANGENHEIM, 2019b).

Outro indicador de desempenho é a precisão média (AP) proposta no desafio de classificação e detecção de objetos *PASCAL Visual Object Classes Challenge* (VOC). No documento do kit de desenvolvimento do edital do concurso, o indicador é definido da seguinte forma (EVERINGHAM e WINN, 2011):

1. Computar um gráfico entre os indicadores precisão/*recall*,
2. Calcular a área abaixo do gráfico através de integração numérica.

Como um indicador único, é utilizado o mAP que é a média do valor obtido para cada classe utilizado no treinamento ou detecção. A razão entre precisão e *recall* mostra

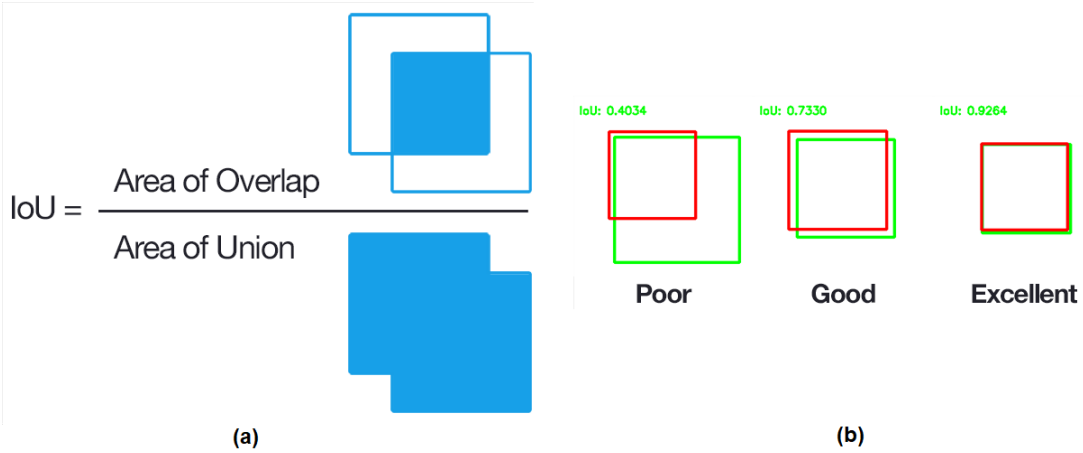


Figura 3.23: (a) Conceito gráfico do indicador IoU (b) Exemplos de qualidade dos resultados do IoU na comparação entre *bounding box* e caixa que determina o objeto real. Adaptado de von Wangenheim (2019b).

a evolução da confiança do modelo e a quantidade de passos necessária até atingir um ponto de estabilidade no *recall*, com o máximo de identificações positivas e mínimo de falsos negativos e com precisão adequada, ou seja, máximo de verdadeiros positivos e o mínimo de falsos positivos.

4. Materiais e Métodos

Para a medição de granulometria em circuito de britagem primária deverão ser estabelecidas algumas etapas para que a aplicação seja robusta o suficiente para o ambiente industrial e que o conceito de identificação, delimitação e medição das partículas seja aplicável para condições mais diversas possíveis tanto em fatores ambientais como iluminação, disposição das partículas na área útil da imagem, posicionamento da câmera, quanto fatores ligados aos métodos escolhidos para implementação, como estratégias para amostragem e filtragem de dados, adaptação, treinamento e auto-calibração dos dados tanto para fase de modelagem quanto para a validação da operação, velocidade de processamento e resposta.

4.1. Caracterização da área de estudos

O objetivo inicial deste trabalho seria implantar um sistema de aquisição e processamento de imagens na operação de britagem primária da unidade de beneficiamento de minério de Vargem Grande, na cidade de Itabirito, Minas Gerais, Brasil, conforme descrito no Capítulo 2. Porém é necessário estabelecer um teste de conceito para verificar a viabilidade da solução e para elaborar cenários diferentes de testes. Em função da limitação de possibilidades de intervenção em processos produtivos que estejam em operação, será realizado um roteiro de testes em laboratório para prova de conceito e posteriormente a verificação da implantação em área operacional. Serão realizados alguns testes também com vídeos da operação da britagem primária nas condições descritas no Capítulo 2 para processamento e testes dos modelos.

Para teste do conceito, serão coletadas imagens de fragmentos de massa mineral sobre uma superfície em situação de ambiente controlado de iluminação, distância até o material de interesse e configurações da câmera para aquisição de dados. Será utilizada uma câmera monocular para gravação das imagens para testes.

Para processamento das imagens será utilizado o pacote *OpenCV*, com programa implementado em linguagem *Python* para leitura do arquivo de imagem, vídeo ou do fluxo de vídeo em tempo real. A partir deste pacote e da implementação do código utilizando bibliotecas de redes neurais convolucionais dos pacotes *TensorFlow* e *Keras* para execução em um microcomputador ou servidor, poderão ser desenvolvidas estratégias para tratamento das imagens quadro a quadro, filtragens e separações de áreas de interesse, técnicas para identificação e delimitação das partículas em relação ao restante da imagem do quadro e procedimentos para realização das medidas lineares do tamanho das partículas.

4.2. Métodos para Detecção de Partículas

Serão utilizadas as técnicas abordadas no Capítulo 3 para detecção e classificação de objetos para identificação das partículas de minério:

1. *Single Shot Detector* (SSD);
2. Faster R-CNN;
3. YOLOv3;
4. U-Net para geração de máscaras de segmentação de objetos, associadas com algoritmos para geração de caixas de detecção.

Para validação dos resultados serão utilizadas as métricas discutidas na Seção 3.6.

4.3. Métodos para Medição de Partículas

Uma vez detectadas as partículas em cada imagem, é necessário realizar o processo de medição dos objetos de interesse, ou seja, dos fragmentos rochosos identificados. Para tanto é necessário realizar um processo de calibração com uma referência na imagem que seja de tamanho conhecido. Para o caso da análise do fluxo de vídeo na britagem primária, devem ser selecionadas referências fixas na imagem, como algum orifício da grelha da peneira vibratória, ou alguma dimensão linear do equipamento que seja constante. Posteriormente deve ser utilizada uma forma específica para comparação. Como o interesse é na dimensão linear de comprimento e largura das partículas, será utilizada a forma retangular para obtenção de valores de largura e comprimento.

4.4. Metodologia

Conforme citado na Seção 4.1, os testes serão realizados com imagens de fragmentos rochosos em bancada, com iluminação e condições de ambientes controladas e também utilizando vídeos de operações da britagem primária de Vargem Grande. Os fragmentos rochosos selecionados para os testes foram coletados na unidade de Vargem Grande e são exemplares similares aos encontrados nas áreas operacionais em sua composição e características físicas como coloração, porosidade e superfície. Estes fragmentos diferem no tamanho, sendo que as amostras coletadas tem tamanho que variam de 4,8 mm à 19 mm, sendo que na área operacional, a característica dos elementos rochosos é de tamanho maior, superiores à 200 mm.

Os métodos escolhidos para identificação dos elementos de interesse, ou seja, dos fragmentos rochosos, conforme descrito nas seções anteriores e em especial na Seção 3.5,

serão os baseados em *deep learning* e aprendizado de máquina, em função das características operacionais para que se deseja projetar uma solução de identificação e medição. Estes métodos tem um fator especial de poderem abranger condições como: iluminação em condições desfavoráveis, elementos na área de análise que são diferentes dos objetos de interesse e possibilidade de readaptação em função de mudanças do material ou objeto de análise.

Uma possível abordagem seria a de utilização de métodos como de janelas deslizantes ("*sliding windows*") como método de busca dos objetos de interesse, associado a técnicas de processamento de imagens como discutido na Seção 3.4. Porém, tal método de busca e suas variações não determinam a exata posição de um objeto de interesse, assim como os limites da área de interesse para análise. Os algoritmos de busca necessitariam de várias iterações em parâmetros de buscas diferentes para cobrir todas as possibilidades, o que torna o custo computacional inviável quando comparado a métodos como o de detecção de objetos por aprendizado de máquina. Os métodos que utilizam aprendizado de máquinas utilizam combinações de métodos com conceitos de classificação de imagens e rotinas de buscas setoriais nas imagens originais.

Conforme mencionado no Capítulo 3, as CNNs selecionadas para estudo e experimento serão SSD, Faster R-CNN, YOLOv3 e UNet. As etapas do experimento consistem em:

- Estabelecer inicialmente uma biblioteca de imagens referente a estes fragmentos rochosos;
- Realizar o treinamento de redes neurais ou estruturas de aprendizado de máquina, à partir da biblioteca montada;
- Avaliar o modelo gerado, através de análise de indicadores de erro em etapas de treinamento, testes e validação;
- Escolha de imagens para detecção de fragmentos rochosos de um conjunto diferente do selecionado para a etapa de elaboração de biblioteca e treinamento de modelo;
- Treinamento de cada modelo, utilizando como entrada as imagens e os rótulos de classe, ou seja, uma delimitação com coordenadas espaciais dos locais nas imagens correspondentes à objetos de interesse. No caso, os rótulos gerados são para classe "pedra", que representa os fragmentos rochosos analisados. Registrar os indicadores de desempenho de classificação e localização de objetos nas etapas de treinamento e validação.
- Realizar testes com conjunto de dados diferente do utilizado na etapa de treinamento e validação. Registrar indicadores de desempenho de classificação e localização de objetos.

- Selecionar a CNN com o melhor desempenho na etapa de construção de modelos com fragmentos rochosos em ambiente controlado e treinar um novo modelo, agora utilizando imagens do ambiente da Britagem Primária de Vargem Grande, com condições operacionais similares às encontradas durante a formulação do problema.

- Elaborar biblioteca com imagens da operação da Britagem Primária. e realizar dois tipos de treinamento com bibliotecas diferentes:

  I) Rotular apenas os objetos de interesse (fragmentos rochosos);

  II) Rotular todos os elementos da imagem de forma a ter diversas classes distintas para classificação;

  Dividir o conjunto de imagens entre treinamento e validação.

  Utilizar técnica de *data-augmentation* para aumentar variabilidade do conjunto de imagens para treinamento e assim aumentar a robustez do modelo.

  Iniciar treinamento do modelo colocando como entrada as imagens e rótulos gerados.

  Validar o modelo com um conjunto de imagens e rótulos diferentes do utilizado para treinamento.

  Registrar indicadores de desempenho de classificação e localização de objetos.

- Realizar testes com conjunto de dados diferentes dos utilizados no treinamento e validação. Registrar indicadores de desempenho de classificação e localização de objetos.

- Executar algoritmo de medição de dimensões dos fragmentos rochosos identificados;

- Comparar o número de identificações com o conjunto de fragmentos disponibilizado para testes. Comparar as medidas obtidas com medidas realizadas manualmente do conjunto de fragmentos.

A Figura 4.1 representa um fluxograma com as etapas da metodologia em sequência e a divisão entre as etapas de testes em ambiente de laboratório para teste do conceito e das etapas do experimento utilizando imagens da Britagem Primária de Vargem Grande.

4.4.1. Elaboração de biblioteca para treinamento de modelos

Para treinar os modelos selecionados é necessário montar uma biblioteca com imagens de um grupo de objetos similares aos de interesse no momento da identificação. Portanto, foram utilizados fragmentos rochosos em condições diferentes de iluminação, cores de fundo, agrupamento para simular condições de análise de imagens que estão presentes no momento de execução dos algoritmos de detecção. Como os testes serão

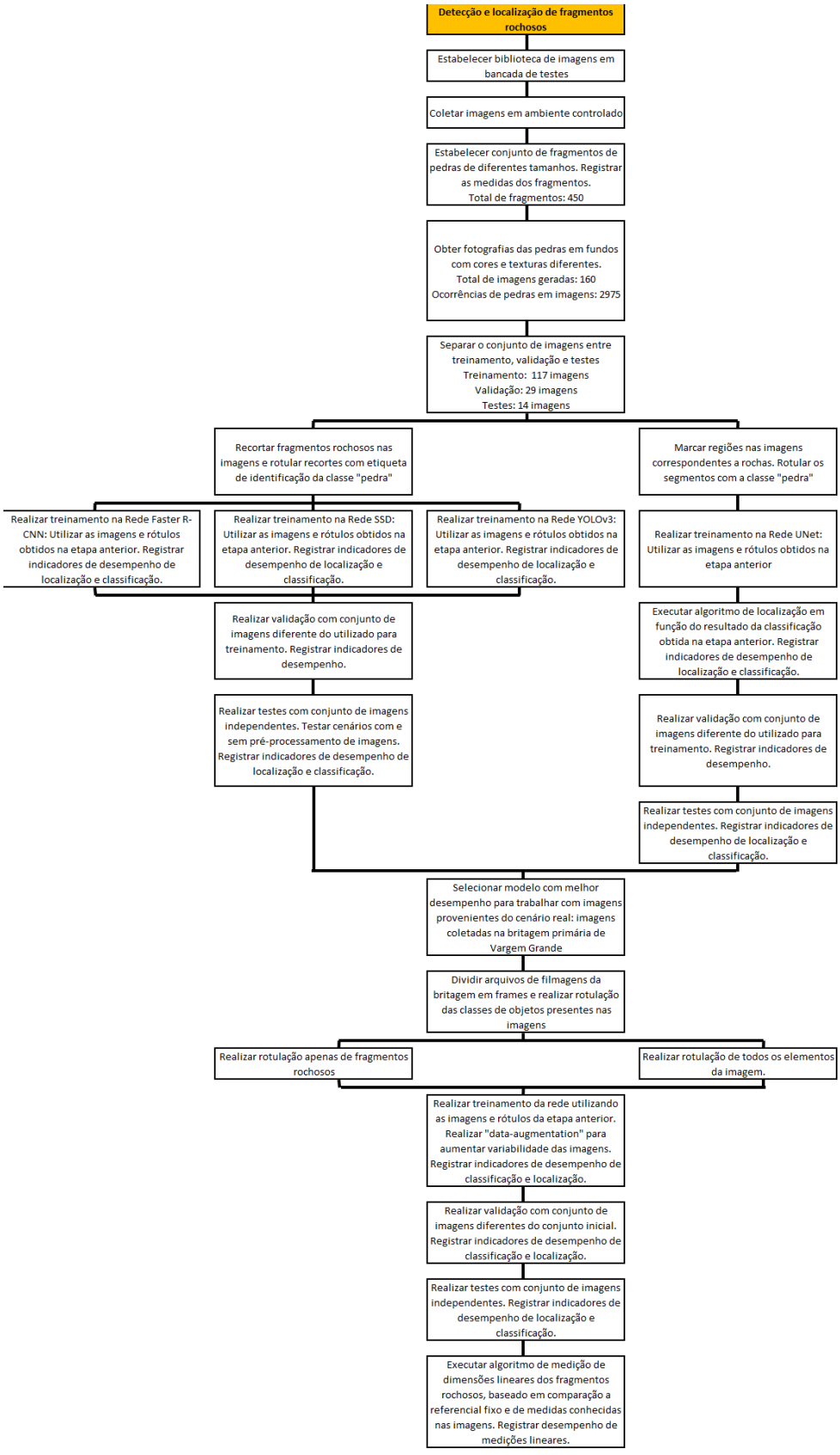


Figura 4.1: Fluxograma de etapas para elaboração de bibliotecas, treinamento e validação de modelos, testes de detecção e localização de objetos e finalmente, medição de dimensões lineares de objetos.

realizados em condições controladas, a quantidade de cenários e condições diferentes dos fragmentos rochosos é limitado. Foram utilizados 450 fragmentos rochosos para criação de uma biblioteca de análise. O tamanho dos fragmentos utilizados para o teste em bancada foi em média de 2,0 cm na sua maior dimensão linear. Os fragmentos da operação da Britagem Primária em Vargem Grande tem medidas acima de 20,0 cm na maior dimensão linear, pois são materiais que não passaram pela primeira peneira de separação, conforme descrito na Seção 2.2. Para um teste em bancada, fragmentos com tais tamanhos inviabilizariam os testes e portanto, foram escolhidos materiais semelhantes, porém com dimensões reduzidas.

Após a coleta, é elaborada uma base de dados em que para cada imagem, com a marcação de um quadrilátero com a posição delimitada que aparece o fragmento rochoso. Este quadrilátero deve ser o mais próximo dos limites de cada fragmento, para que a matriz de *pixels* limitada seja a mais próxima da representação do objeto desejado (Figura 4.2). São atribuídos rótulos a estes quadriláteros, que identificam o objeto com uma classe. Estes rótulos são utilizados como entrada na CNN para treinamento, juntamente com as imagens a que eles referenciam.

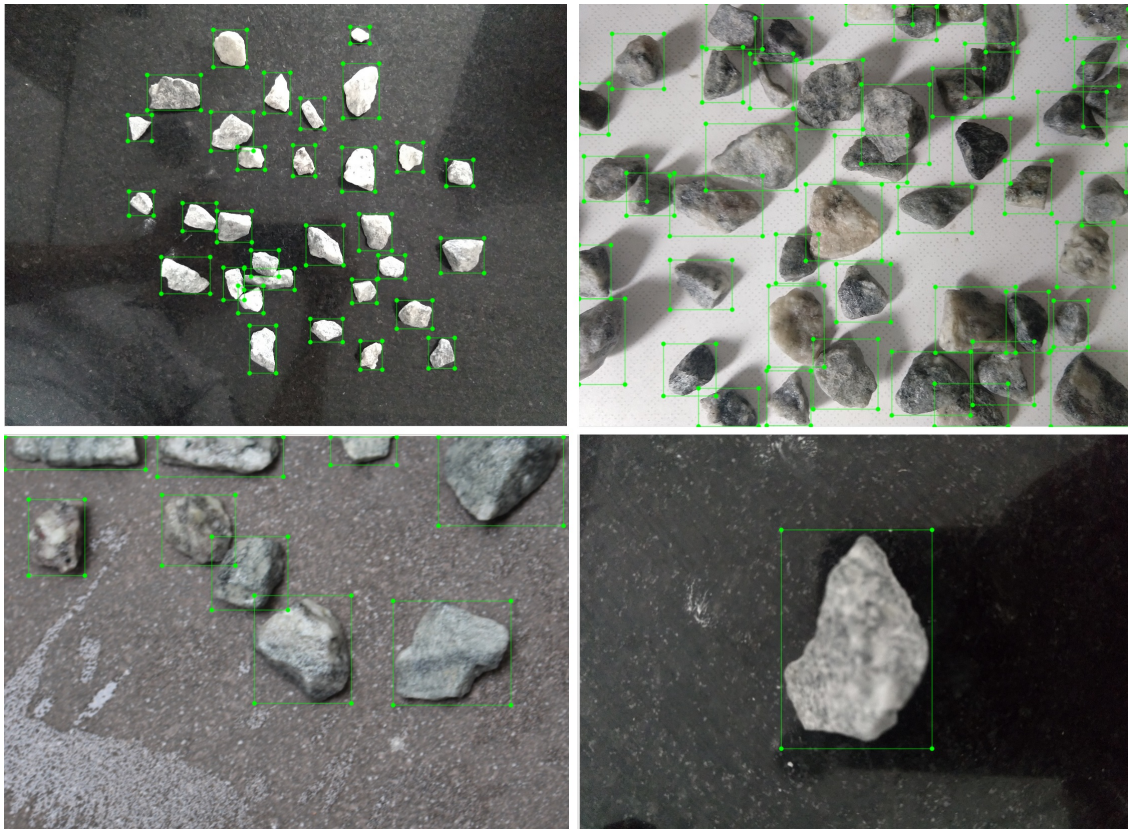


Figura 4.2: Imagens utilizadas para criação de biblioteca para treinamento, testes e validação de modelos.

Os 450 fragmentos foram organizados em conjuntos para treinamento e testes dos modelos. 117 arquivos de imagens foram separados para treinamento, com 1803 amostras

de fragmentos. Para a biblioteca de testes foram separados 29 arquivos de imagens, com 1172 amostras. Foi utilizado a aplicação *LabelImg* para organização das imagens, com a determinação dos quadriláteros de área de interesse das imagens (Figura 4.3a). Posteriormente este *software* gera arquivos do tipo *xml* com as coordenadas de cada área de interesse. (Figura 4.3b).

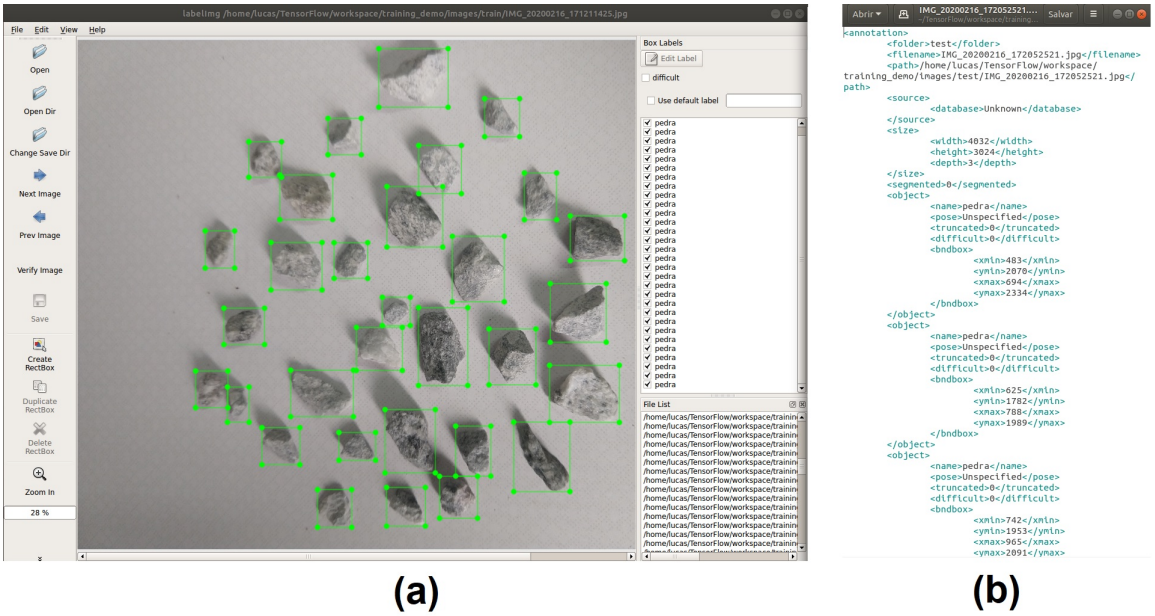


Figura 4.3: a) *Software LabelImg* utilizado para organização das imagens. b) Arquivo *XML* gerado com as informações de quadrilátero definido para identificação de fragmento.

Com o objetivo de se utilizar como entrada nos modelos de redes neurais para treinamento, foi criada uma classe denominada "pedra"para rotular cada fragmento rochoso demarcado como região de imagem identificada como verdadeira ou positiva. Esta classe será o rótulo utilizado na classificação de imagens e identificação de objetos como saída das redes neurais utilizadas e descritas a seguir.

4.4.2. Preparação de biblioteca para treinamento do modelo U-Net

Para a formulação da biblioteca de imagem para o treinamento da rede U-Net, o processo para delimitação da área que cada fragmento rochoso ocupa na imagem é realizada de forma diferente, com a criação de um polígono com a área aproximada (Figura 4.4).

Para a rede U-Net foi utilizado o software *LabelMe* para a geração dos polígonos e delimitação das máscaras referentes as áreas dos objetos nas imagens. Em seguida é gerado um arquivo do tipo *JSON* com as coordenadas dos pontos que fazem parte destes polígonos criados. A partir destas áreas os modelos realizaram as operações para extrações de características e treinamentos para obtenção dos pesos de cada camada das redes.

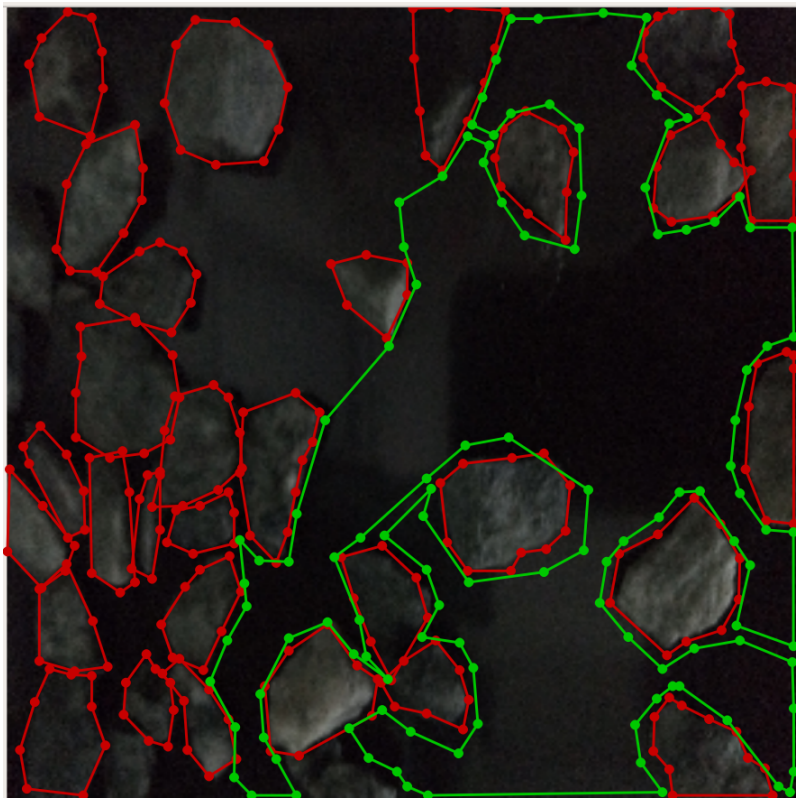


Figura 4.4: Geração de polígonos para biblioteca de modelo de segmentação.

Para as imagens extraídas dos vídeos gravados da operação da britagem de Vargem Grande serão realizadas as anotações dos demais elementos da imagem, além dos fragmentos rochosos (denominados como a classe "pedra") para verificar a eficiência do treinamento da rede com o mapeamento de todos os elementos da imagem (Figura 4.4).

Figura 4.5: Máscaras geradas para delimitação de objetos e classes demarcadas para utilização no treinamento da rede U-Net.

Após a criação das bibliotecas com as imagens e máscaras de detecção de objetos, é realizada a criação de uma imagem com a conversão das máscaras para escala de cinza em que cada classe tem um tom equivalente para diferenciação no momento de identificação pelas camadas da rede neural (Figura 4.6).

Ronneberger *et al.* (2015) em seu artigo cita que a biblioteca para treinamento do modelo U-Net pode ser obtida à partir de poucas amostras e utilizando técnicas de

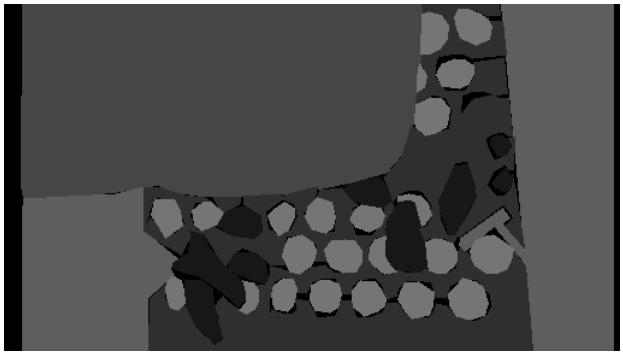


Figura 4.6: Conversão das máscaras de segmentação para escala de cinza

aumento da variância das características das imagens (*augmentation*). Na construção da biblioteca em questão, foram utilizadas as transformações de espelhamento horizontal e vertical da imagem e também uma variação randômica do brilho da imagens. Sendo assim, para a biblioteca de treinamento, foram utilizadas 16 imagens originais e criadas as máscaras de segmentação equivalentes. Após as operações de aumento de variância, a biblioteca de testes foi composta por 128 imagens sendo as 16 originais mais 112 cópias das imagens originais alteradas com as operações de transformação. Para a biblioteca de testes foram utilizadas 4 imagens originais e após as transformações a biblioteca de testes passou para 32 amostras. As imagens das máscaras de segmentação também foram devidamente transformadas.

4.4.3. Ambiente computacional

Para o processamento de imagens foram utilizados um ambiente Linux, com bibliotecas específicas para operações com imagens. Seguem as configurações de *hardware* utilizadas:

- Máquina Virtual Linux - Distribuição Ubuntu 18.04.4 LTS - 64 Bits
- Memória RAM: 9,6 GB
- Processadores: 4 x Intel Core i7-8550U @ 1,80 GHz

O desenvolvimento foi realizado em ambiente *Python*. Para a implementação dos modelos de detecção de objetos foi utilizado o pacote de bibliotecas de estruturas de treinamento e operações com redes neurais e aprendizado de máquina *TensorFlow*, a bilioteca de *deep learning* para *Python Keras*, juntamente com as bibliotecas para processamento de imagens *OpenCV*.

A implementação dos modelos *Faster R-CNN* e SSD foram realizadas utilizando a API de detecção de objetos disponibilizada pela equipe de desenvolvimento do *TensorFlow*. O modelo *YOLOv3* foi implementado com *TensorFlow* utilizando o modelo

apresentado por Redmon e Farhadi (2018). O modelo U-Net foi implementado utilizando o pacote *Keras*, utilizando a estrutura de CNN proposta por Ronneberger *et al.* (2015).

4.4.4. Implementação do modelo *Single Shoot Detector* (SSD) - Detector de única imagem

As configurações para treinamento da rede SSD utilizadas para treinamento no *TensorFlow* foram:

- Modelo de Rede Neural Convolucional para classificação pré-treinada utilizada como base: Inception V2
- *Dataset* de imagens utilizado com base da rede de pré-treinamento: COCO
- Taxa de aprendizagem do treinamento: constante
- Número de imagens para treinamento: 117
- Número de bateladas: 1
- Número de imagens por batelada: 1
- número de passos por época: 117
- Número total de passos: 200.000 (limite empírico de convergência da função de erro para rede modelo Inception V2 utilizando o dataset de imagens COCO)
- Função de ativação: RELU6
- Escalas de avaliação da imagem (*"aspect ratios"*): 1,0x, 2,0x, 0,5x, 3.0x, 0,3333x

4.4.5. Implementação do modelo *Faster R-CNN*

Similar ao modelo SSD, o modelo *Faster R-CNN* tem as seguintes características:

- Busca seletiva para gerar resultados de possíveis objetos identificados (camadas de classificação).
- Realimentação da rede neural convolucional (CNN) com tais resultados.
- Treinamento separado para definição de "caixas-limite"de detecção de objetos (*bouding boxes* que delimitam a região do objeto identificado (camadas de detecção de objetos).

A diferença em relação ao modelo SSD é que as redes Faster R-CNN utilizam as "caixas-âncora"em mais camadas convolucionais, tornando o modelo mais complexo, recursivo e preciso. O modelo SSD é mais rápido por estimar as caixas de detecção de objetos com menos camadas convolucionais, mais precisamente, uma camada para cada resolução definida durante o treinamento.

As configurações para treinamento da rede Faster R-CNN utilizadas para treinamento no *TensorFlow* foram:

- Modelo de Rede Neural Convolucional para classificação pré-treinada utilizada como base: Inception V2
- *Dataset* de imagens utilizado com base da rede de pré-treinamento: COCO
- Taxa de aprendizagem do treinamento: constante
- Número de imagens para treinamento: 117
- Número de bateladas: 1
- Número de imagens por batelada: 1
- número de passos por época: 117
- Número de passos total: 200.000 (limite empírico de convergência da função de erro para rede modelo Inception V2 utilizando o dataset de imagens COCO)
- Função de ativação: RELU6
- Escalas de avaliação da imagem (*"aspect ratios"*): 1,0x, 2,0x, 0,5x, 3.0x, 0,3333x

4.4.6. Implementação do método *YOLOv3 - You Only Look Once*

O método YOLO descrito na Seção 3.5.7 também foi utilizado para treinamento e identificação de objetos. Os parâmetros utilizados para o treinamento são descritos abaixo.

- Modelo de Rede Neural Convolucional para classificação pré-treinada utilizada como base: Darknet-53
- *Dataset* de imagens utilizado com base da rede de pré-treinamento: COCO
- Taxa de aprendizagem do treinamento: inicia em $1 * e^{-4}$. Esta taxa foi configurada para decair a cada 5 épocas do treinamento. O limite inferior da taxa de aprendizagem foi configurado para $1 * e^{-6}$

- Número de imagens para treinamento: 117
- Número de bateladas: 1
- Número de imagens por batelada: 1
- número de passos por época: 117
- Número máximo de épocas de treinamento: 100
- Número de passos total: 11700
- Função de ativação: linear do tipo *Leaky ReLU*, seguindo a equação 4.1:

$$\phi(x) = \begin{cases} x & \text{se } x > 0 \\ 0,1x & \text{caso contrário} \end{cases} \tag{4.1}$$

- Escalas de avaliação da imagem (*"aspect ratios"*): (1/32)x, (1/16)x, (1/8)x

4.4.7. Implementação do método U-Net

Conforme a Seção 3.5.9, o modelo U-Net realiza a geração de máscaras de segmentação semântica a partir do treinamento de uma rede neural convolucional. A seguir as configurações de treinamento da rede.

- Número de imagens para treinamento: 128
- Número de bateladas: 1
- Número de imagens por batelada: 1
- número de passos por época: 512
- Número máximo de épocas de treinamento: 40
- Número de passos total: 20480
- Função de ativação: linear do tipo *ReLU*

Após o treinamento, para utilização do modelo U-Net para detecção de objetos é necessário um pós processamento com a detecção das bordas das máscaras de segmentação geradas para validação.

4.4.8. Execução de experimentos

Para teste do conceito, foram coletadas imagens de fragmentos de massa mineral sobre uma superfície em situação de ambiente controlado de iluminação, distância até o material de interesse e configurações da câmera para aquisição de dados. Será utilizada uma câmera monocular para gravação das imagens para testes. Também serão utilizadas imagens de gravações de vídeo da operação da Britagem Primária de Vargem Grande para teste de aplicação do modelo com melhor desempenho na etapa de testes em bancada.

Para processamento das imagens será utilizado o pacote *OpenCV*, com programa implementado em linguagem *Python* para leitura do arquivo de vídeo ou do fluxo de vídeo em tempo real. Foram utilizadas duas estratégias para avaliar o desempenho dos métodos de detecção: enviar as imagens para a CNN para detecção sem pré-processamento ou realizar operações nas imagens para limitar a área de busca das CNNs, com objetivo de otimizar a busca por objetos de interesse para serem detectados. As operações de pré-processamento utilizadas foram descritas à seguir:

- Aplicar filtro do tipo *blur* (ou borrar) para reduzir ruídos na imagem.
- Aplicar conversão da imagem para escala de cinza, para reduzir influência de demais cores na análise e aproximar das cores das imagens da biblioteca de treinamento.
- Aplicar filtro do tipo *threshold* para eliminar regiões claras na imagem.
- Inverter as cores da imagem após aplicação do filtro *threshold*, para marcar as áreas que não são claras e que não são as áreas de interesse para análise.
- Obter o complemento de cores da imagem original. O resultado são apenas as regiões claras da imagem que tem tonalidade próxima das imagens utilizadas na biblioteca de treinamento.

No Capítulo 5 serão abordados os resultados para as etapas de treinamento e também para os testes de detecção utilizando os modelos após a fase de treinamento.

5. Resultados

5.1. Treinamento, validação e testes de detecção de objetos dos modelos SSD, Faster R-CNN, YOLOv3, UNet - Testes em bancada

5.1.1. Rede SSD

Conforme citado na Seção 4.4.4, o número total de passos do treinamento foram de 106.301. O número de épocas do treinamento foi igual a 908. A Figura 5.1 exibe o gráfico da função de perda para a massa de dados de treinamento e também para a massa de dados de validação, conforme equação 3.5. Os dados foram apresentados no gráfico da Figura 5.1 de forma suavizada com a média móvel exponencial (EMA - *Exponential Moving Average*) em função da grande quantidade de pontos.

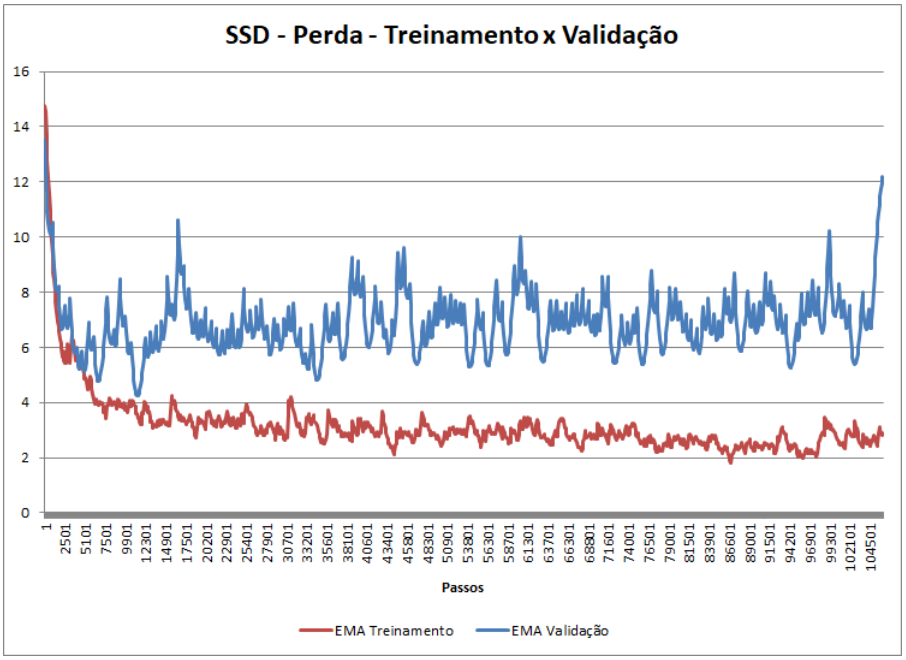


Figura 5.1: Função de perda para a rede modelo SSD, para as massas de dados de treinamento e validação.

A Figura 5.2 exibe o gráfico da métrica mAP para avaliação do modelo. Foram plotados os dados originais e também uma série com uma média móvel exponencial dos dados originais.

A Figura 5.3 exibe os erros do treinamento da CNN tipo SSD durante o treinamento separada pelas erros nas camadas de treinamento e localização, com os filtros de médias móveis exponenciais correspondentes.

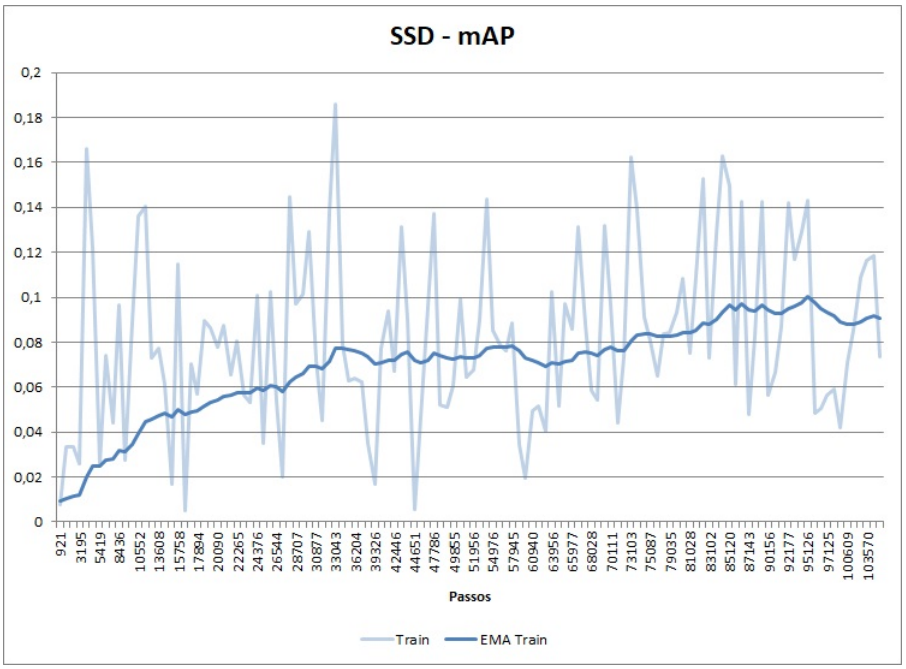


Figura 5.2: Precisão Média (mAP) para a rede modelo SSD, após treinamento

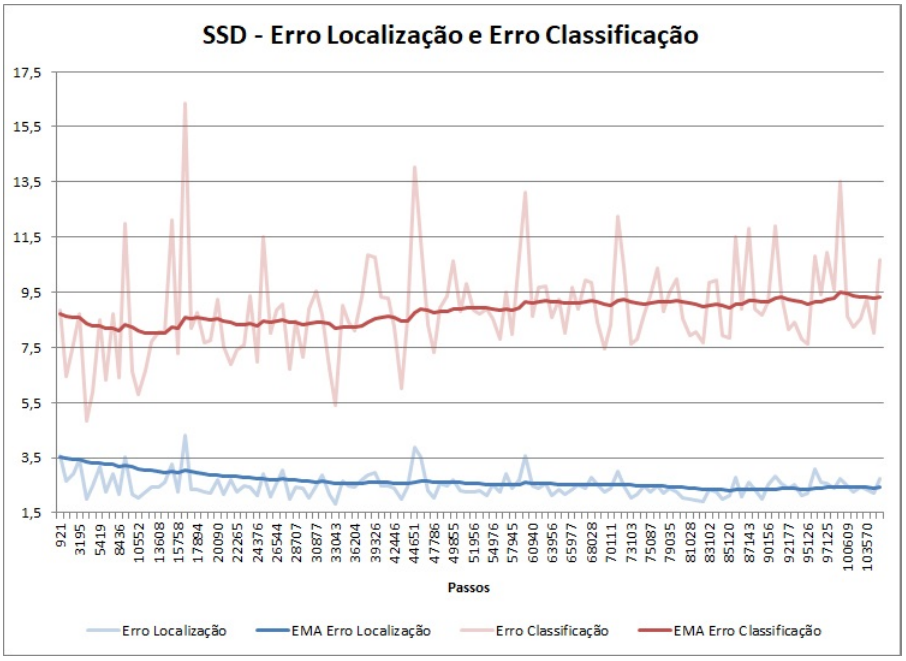


Figura 5.3: Perdas de Classificação e Localização para a rede modelo SSD.

5.1.2. Rede Faster R-CNN

O número total de passos do treinamento foram de 150.501. O número de épocas do treinamento foi igual a 1.286. A Figura 5.4 exibe o gráfico da função de perda para a massa de dados de treinamento e também para a massa de dados de validação.

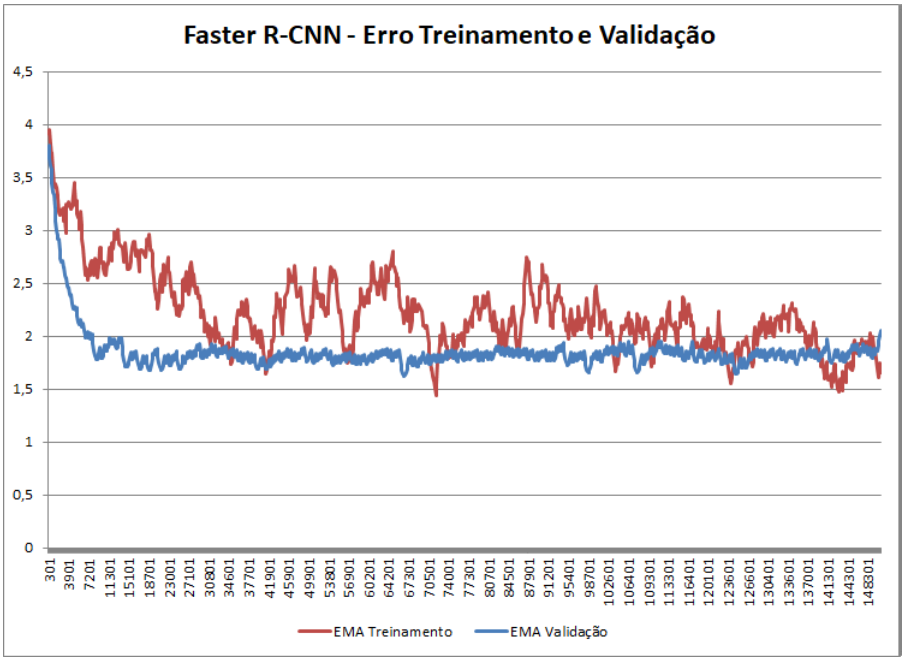


Figura 5.4: Função de perda para a rede modelo Faster R-CNN, para as massas de dados de treinamento e validação.

A Figura 5.5 exibe o gráfico da métrica mAP para avaliação do modelo. Foram plotados os dados originais e também uma série com uma média móvel exponencial dos dados originais.

A estrutura da rede Faster R-CNN é composta pela fase inicial de proposição de região (RPN) conforme explicitado na Seção 3.5.6. A Figura 5.6 apresenta dois tipos de erros na proposição de região: o primeiro é na assertividade da predição (se a região proposta contém objeto ou se a região selecionada é apenas fundo da imagem) e o erro de localização, dada a classificação previamente realizada. A Figura 5.7 apresenta os erros globais de classificação e localização da rede Faster R-CNN.

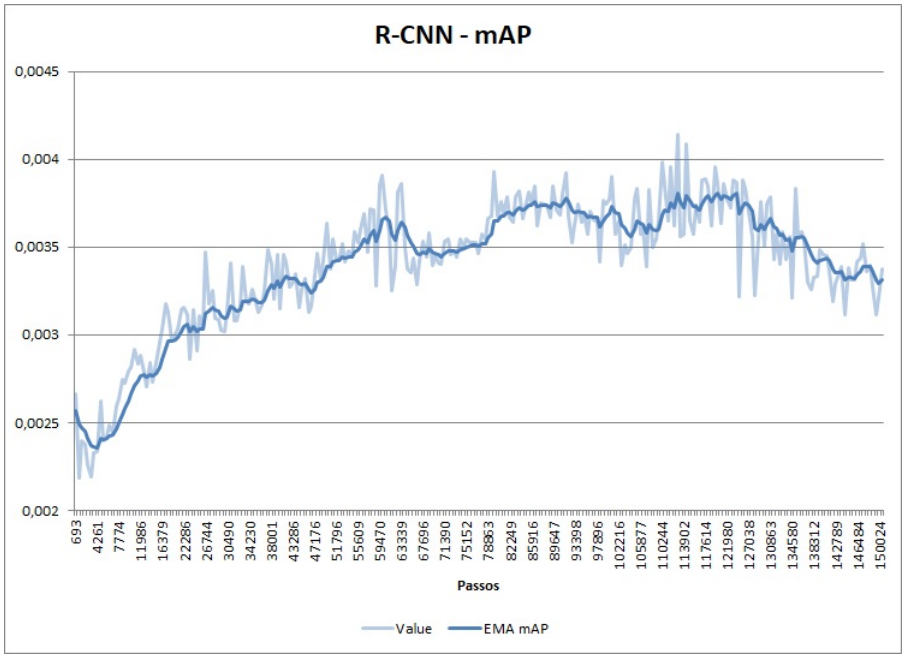


Figura 5.5: Precisão Média (mAP) para a rede modelo Faster R-CNN, após treinamento

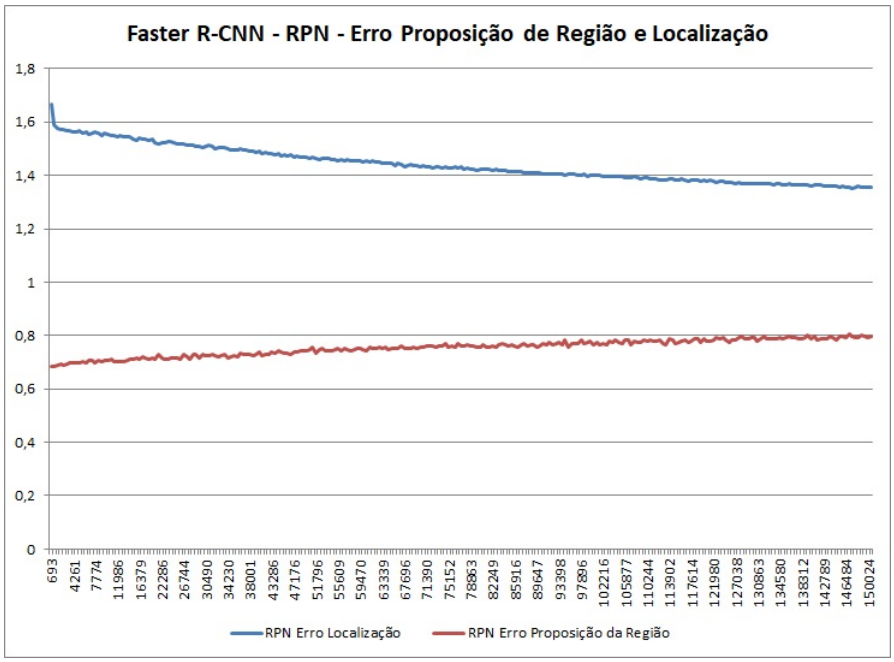


Figura 5.6: Erros na etapa de proposição de região (RPN) para o modelo Faster R-CNN na validação

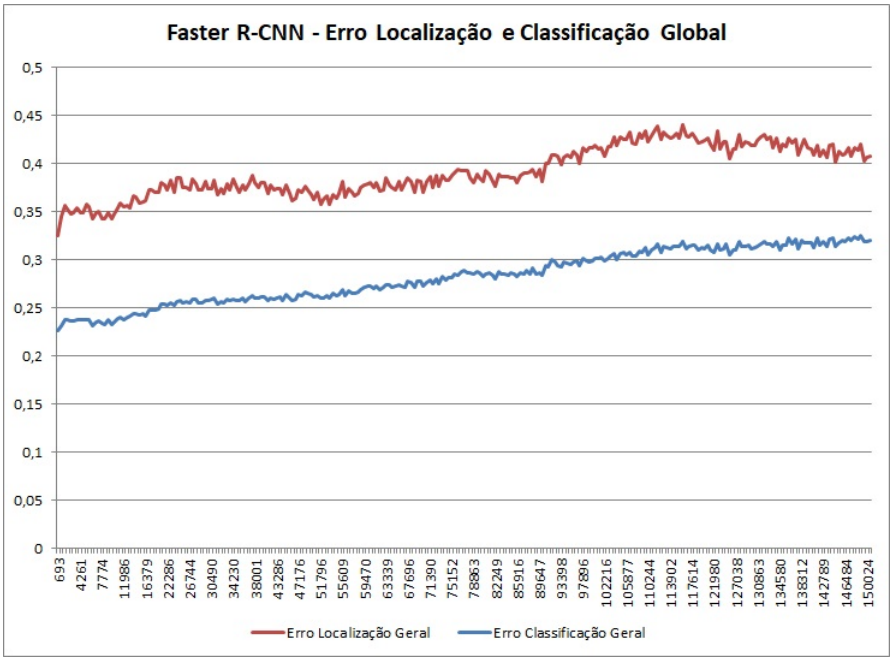


Figura 5.7: Erros de classificação e localização globais da rede Faster R-CNN

5.1.3. Rede YOLOv3

O número total de passos do treinamento foram de 11.700. O número de épocas do treinamento foi igual a 100. A Figura 5.8 exibe o gráfico da função de perda para a massa de dados de treinamento e também para a massa de dados de validação.

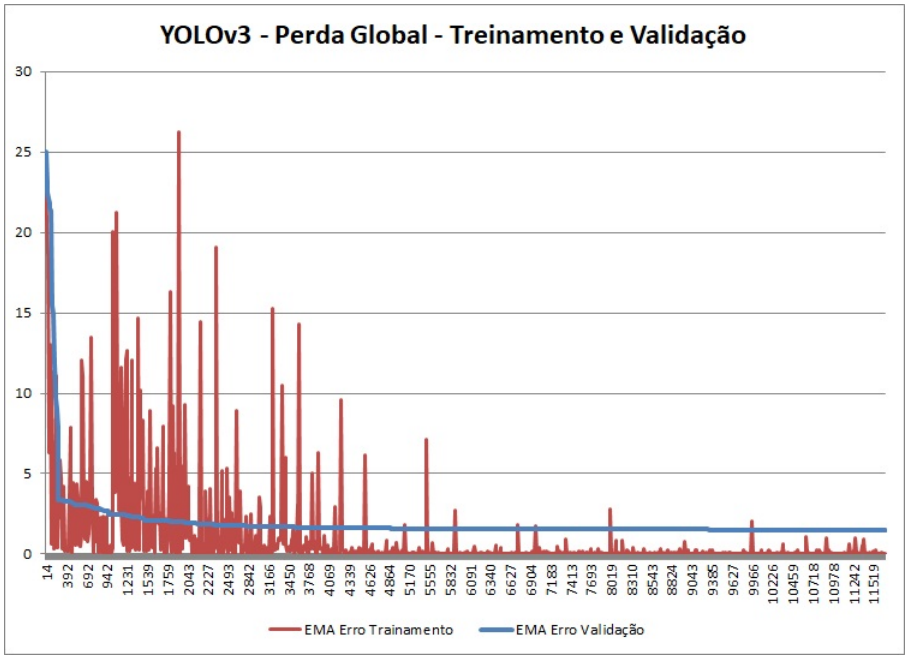


Figura 5.8: Função de perda para a rede modelo YOLOv3, para as massas de dados de treinamento e validação.

A Figura 5.9 exibe o gráfico da métrica mAP para avaliação do modelo.

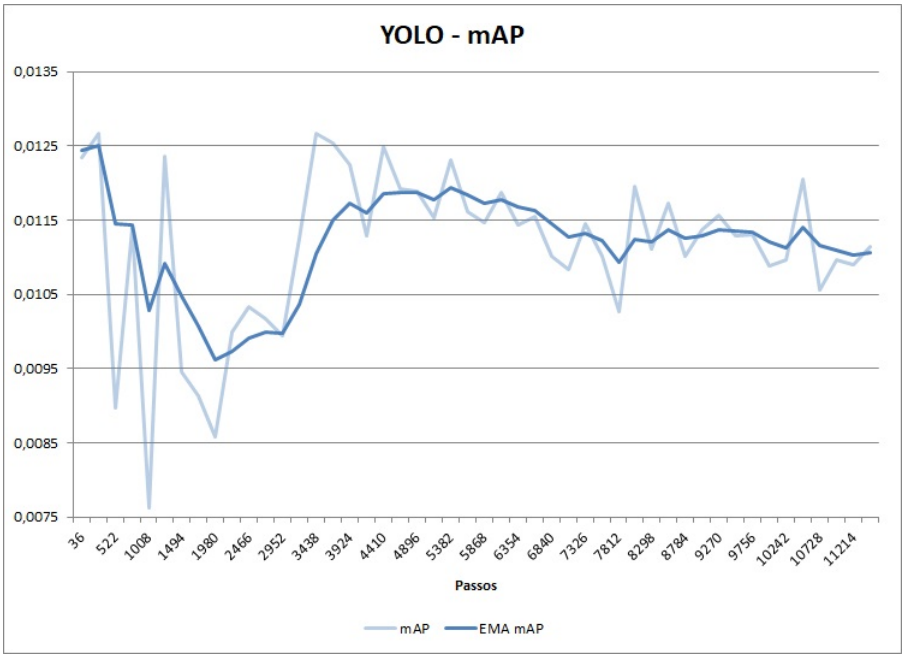


Figura 5.9: Precisão Média (mAP) para a rede modelo YOLOv3, após treinamento

Conforme a Seção 3.5.7, a função de perda do modelo YOLOv3 tem 4 partes referentes à localização do objeto (coordenadas x e y do centróide e altura h e largura w da *bounding box* predita), à classificação e à confiança da classificação. Esta função objetivo é minimizada ao longo do treinamento através do otimizador implementado na CNN. A Figura 5.10 exibe as perdas nas etapas de treinamento e validação para as 4 variáveis descritas.

5.1.4. Rede U-Net

O número total de passos do treinamento foram de 20992. O número de épocas do treinamento foi igual a 40. A Figura 5.11 exibe o gráfico da função de perda para a massa de dados de treinamento e também para a massa de dados de validação.

A Figura 5.12 exibe o gráfico da métrica mAP para avaliação do modelo. Foram plotados os dados para a massa de dados de treinamento e validação.

5.1.5. Testes de Detecção de Objetos

Após o treinamento foi realizado o teste de detecção de objetos, foram realizados os testes de detecção em dois tipos de cenário: imagens com fundo branco e imagens com o fundo preto. Além disso, foram utilizadas duas distâncias de aproximação para verificar a capacidade de detecção em função do nível de detalhe captado nas imagens. Além do posicionamento das imagens, também foram utilizadas duas versões de testes com e sem pré-processamento das imagens antes do processamento pela CNN, conforme descrito pela Seção 4.4.5. Foi utilizado um algoritmo para medição do tamanho dos objetos detectados,

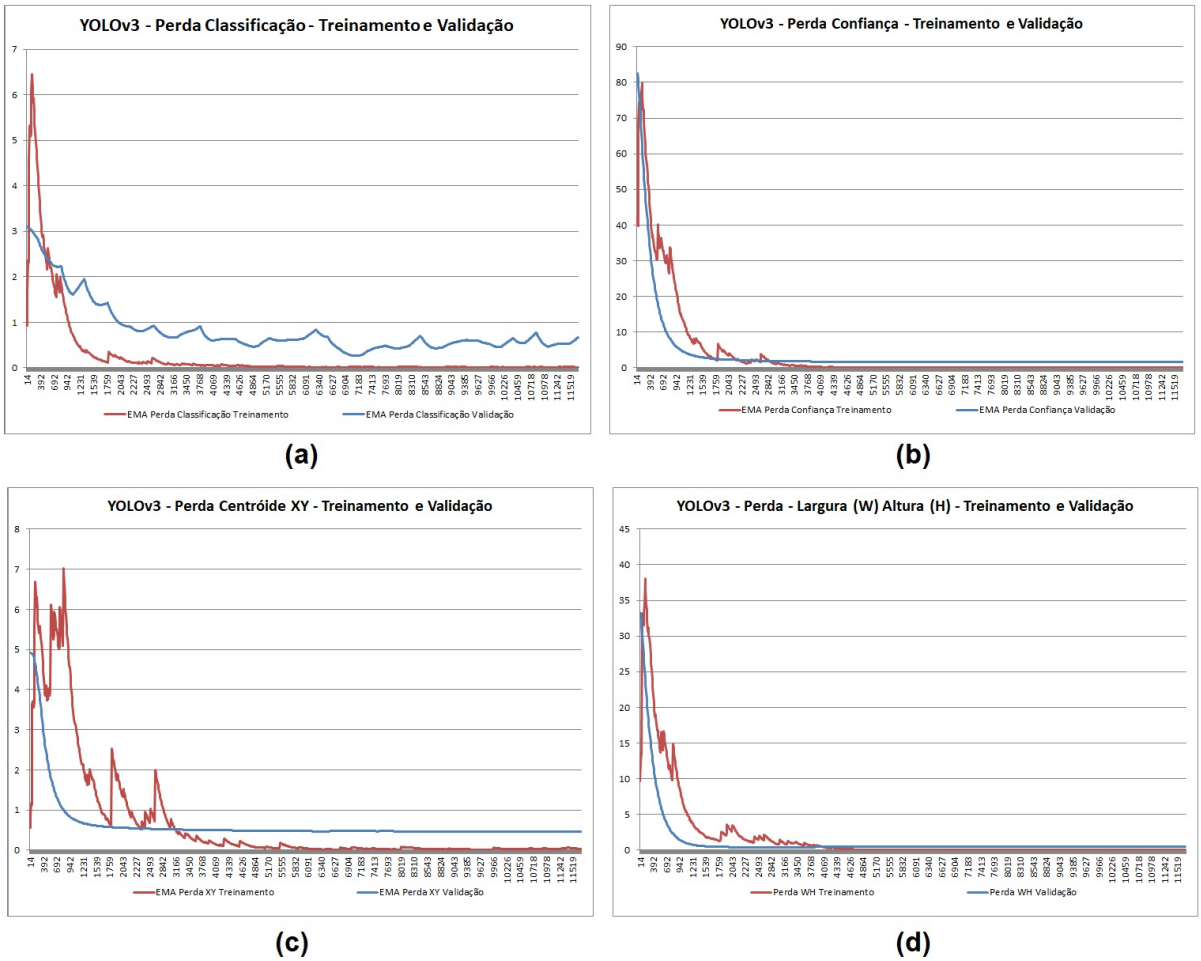


Figura 5.10: Perdas específicas YOLOv3 - a) Classificação b) Confiança c) Coordenadas x e y do centróide da *bounding box* predia d) Largura (w) e altura (h) da *bounding box* predita

considerando o tamanho da largura das *bounding boxes* geradas na predição. A medida em milímetros é obtida à partir da conversão da distância em pixels entre dois pontos a uma referência com comprimento linear conhecido linear em cada imagem. Os exemplos dos testes de detecção de imagens estão na Seção A.1.

5.1.6. Discussão dos Resultados

Os resultados obtidos no treinamento das redes SSD, Faster R-CNN e YOLOv3 apontam para baixas medidas de precisão conforme os gráficos de mAP e detecções com baixo índice de assertividade conforme as figuras na sessão A.1. A Tabela 5.1 exibe os índices das referências bibliográficas de cada rede e as versões implementadas neste trabalho.

Expandindo a análise para os gráficos de perda, observa-se que as 3 redes tem comportamentos diferentes na comparação entre as etapas de treinamento e validação. Para a rede SSD, o erro na validação foi aproximadamente 3 vezes maior do que no

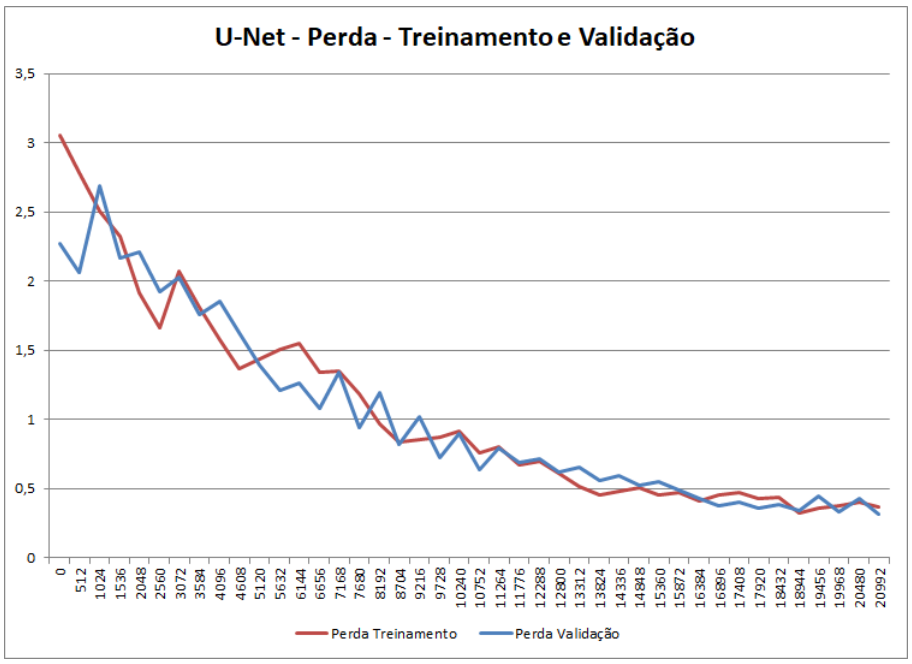


Figura 5.11: Função de perda para a rede modelo U-Net, para as massas de dados de treinamento e validação.

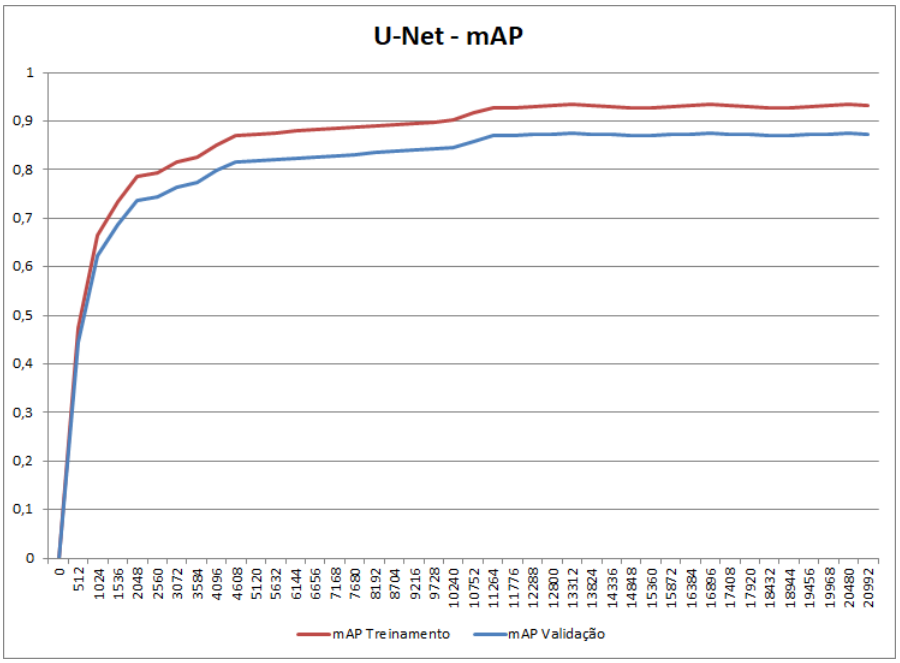


Figura 5.12: Precisão Média (mAP) para a rede modelo U-Net, após treinamento e validação

Rede	**Backbone**	**Dataset**	**Tamanho da entrada**	**mAP**
SSD (LIU *et al.*, 2016)	VGG16	PASCAL VOC2007	300x300	74,3
SSD	Inception-v2	Customizado	300x300	0,09
Faster R-CNN (REN *et al.*, 2015)	VGG16	PASCAL VOC2007	300x300	69,9
Faster R-CNN	Inception-v2	Customizado	300x300	0,003
YOLOv3 (REDMON e FARHADI, 2018)	Darknet-53	COCO	480x320	51,2
YOLOv3	Darknet-53	Customizado	480x320	0,01

Tabela 5.1: Comparação entre implementações dos modelos SSD, Faster R-CNN, YOLOv3 das referências bibliográficas e das realizadas neste trabalho.

treinamento, conforme Figura 5.1. Observando os erros da Figura 5.3, o erro na etapa de classificação é muito maior do que o erro na etapa de localização. A escolha de um *backbone* como a rede Inception-v2 utilizando a transferência de conhecimento do treinamento no dataset COCO não convergiu para os mesmos níveis de mAP descritos no artigo de Szegedy *et al.* (2015). As características de uma rede com muitas camadas e o problema do "desaparecimento do gradiente"contribuem para um baixo desempenho na tarefa de classificação. O desempenho melhor no treinamento em comparação à validação apontam para a ocorrência do "*overfitting*", que será discutido mais adiante.

A rede Faster R-CNN também apresenta baixo mAP (Figura 5.5) e valores de erros elevados (Figura 5.4, quando comparados ao trabalho de Ren *et al.* (2015), com treinamento no dataset COCO. Ao analisar a função de perda e seu componente para a localização e proposição de região (que é a entrada da rede Faster R-CNN), observa-se um erro elevado na etapa de localização, que significa que a rede gera regiões de pesquisa imprecisas para as próximas etapas de treinamento e assim compromete os resultados das próximas etapas de classificação da imagem e localização de objetos. Da mesma forma, a escolha do *backbone* Inception-v2 e sua estrutura sem o devido tratamento para a variação brusca de gradientes entre camadas é uma hipótese para o baixo desempenho na etapa de proposição de região e classificação.

A rede YOLOv3 apresenta baixo valor de mAP (Figura 5.9), baixo valor na função de perda na fase de treinamento e alto valor na etapa de validação, evidenciando o *overfitting*. Destaque para a análise dos erros específicos, com o alto valor de erro de confiança na classificação e alto erro na localização (Figura 5.10).

Um problema comum entre os 3 modelos é o baixo desempenho na precisão, com destaque para níveis elevados de erro na etapa de classificação. A variância e representatividade do *dataset* para o treinamento é vital para a qualidade da extração de *features* (GOODFELLOW *et al.*, 2016). Não foram utilizadas técnicas de aumento de variância

do *dataset* montado, apesar de terem sido escolhidas condições controladas como cor do fundo, iluminação e distância de obtenção das imagens.

O *overfitting* é caracterizado pelo bom comportamento no treinamento da evolução da função de erro e por valores elevados na função de erro na validação. Ou seja, o modelo é ineficiente na generalização dos casos e tem predições com baixa assertividade fora do treinamento. A taxa de aprendizagem para o treinamento foi escolhida para o treinamento das redes com objetivo de realizar um aprendizado suficiente durante o processo na minimização do erro. Todos os modelos foram treinados durante por uma quantidade de épocas acima do que a literatura referência de cada um deles indica para convergência na obtenção de pesos com níveis de desempenho adequados.

Os modelos implementados não tem estruturas para regularização, que combinados com a taxa de aprendizado, que buscam descartar valores elevados de pesos em cada passo do treinamento para privilegiar o aprendizado das *features*. Uma melhoria para buscar a evolução do mAP é utilizar pesos na parcela de normalização da função de erro e observar o comportamento da precisão e os impactos que ela causa na convergência da função de erro.

O trabalho de Abramovich e Pensky (2019) aborda a questão da precisão dos modelos de classificação em função da quantidade de classes utilizadas e o número de *features* significantes necessárias para determinar uma associação positiva a uma classe. O artigo descreve que modelos com maior número de classes tem maior precisão. Isso se deve ao fato de que *features* com pouca relevância para associação a uma classe podem ser importantes para determinação de outras e confirmação de uma associação positiva de uma imagem à uma classe. O *dataset* organizado para os experimentos basicamente contém duas classes: as partículas minerais (denominadas "pedras") e o fundo da imagem. Sendo assim e considerando os aspectos discutidos anteriormente, é necessário evoluir a implementação na obtenção de um *dataset* maior, com maior variância, maior diversidade de objetos e classes na perspectiva de se obter resultados melhores na classificação.

A rede U-Net tem melhor desempenho do que as demais redes tanto na convergência da função de perda, quanto no indicador mAP, estabilizando em aproximadamente 0,93 para a massa de dados de treinamento e 0,87 para a massa de dados de validação, se mostrando uma rede CNN mais eficiente para a tarefa de classificação com as condições similares às demais redes. A Seção 5.2 trás os resultados do treinamento, validação e testes com a rede U-Net utilizando a massa de dados da britagem primária de Vargem Grande, com as condições similares à formulação do problema.

5.2. Treinamento, validação e testes de detecção de objetos - U-Net

Conforme descrito na Seção 3.5.9, a rede U-Net é idealizada para realização da segmentação semântica de imagens, que realiza a separação de áreas de acordo com o pertencimento de um objeto a uma determinada classe. Para os testes, conforme metodologia descrita na Seção 4.4.1, a biblioteca foi criada a partir de imagens obtidas na gravação da operação da operação da britagem primária de Vargem Grande. Foram estabelecidas 5 classes distintas para classificação nas imagens e realizados os procedimentos de marcação das áreas pertinentes a cada classe. O modelo foi treinado por 40 épocas, sendo cada época composta por 512 passos.

A Figura 5.13 exibe o indicador de acurácia para o treinamento e validação do modelo.

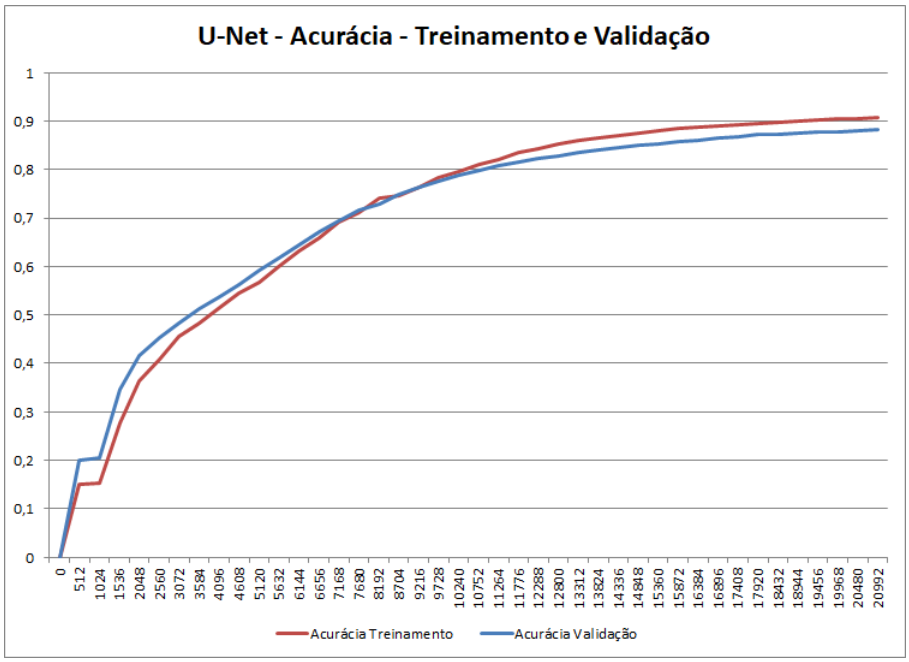


Figura 5.13: Acurácia rede U-Net - Treinamento e Validação

A Figura 5.14 exibe a evolução da função de perda nas etapas de treinamento e validação.

O modelo apresenta acurácia de 91,3% e convergência da função de erro em níveis semelhantes ao trabalho de Ronneberger *et al.* (2015) após treinamento e validação, que apresentou índice de 92,03% utilizando um *dataset* próprio.

5.2.1. Testes de Detecção de Objetos

A rede U-Net não tem camadas para detecção de objetos e geração de *bounding boxes* para localização. Após o treinamento do modelo, as imagens de validação são

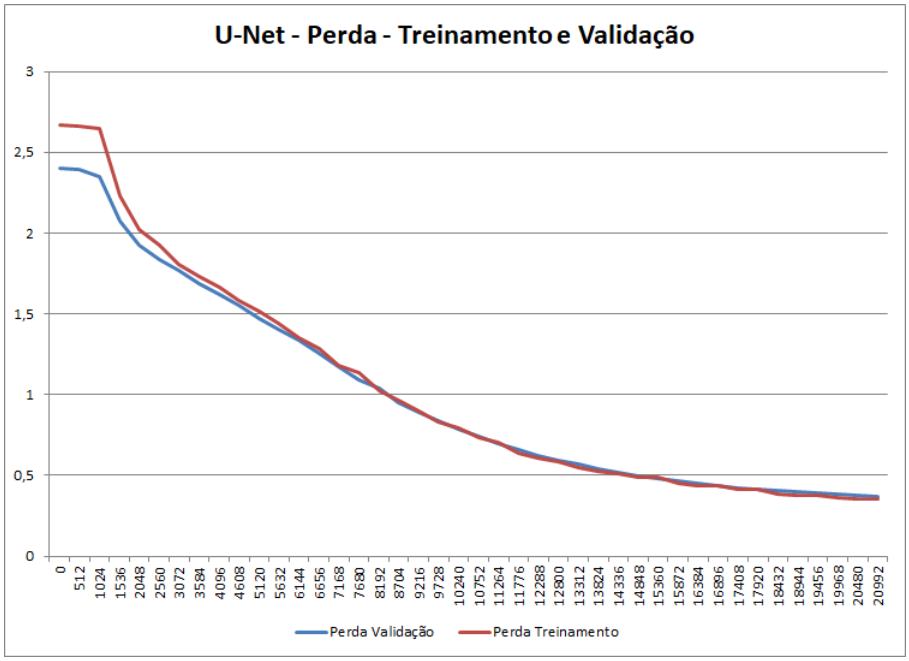


Figura 5.14: Perdas rede U-Net - Treinamento e Validação

processadas pelo modelo para gerar imagens equivalentes com o mapeamento das áreas de acordo com as classes criadas para o *dataset*. A Figura 5.15 exibe uma imagem utilizada como entrada da rede e o resultado da segmentação semântica utilizada.

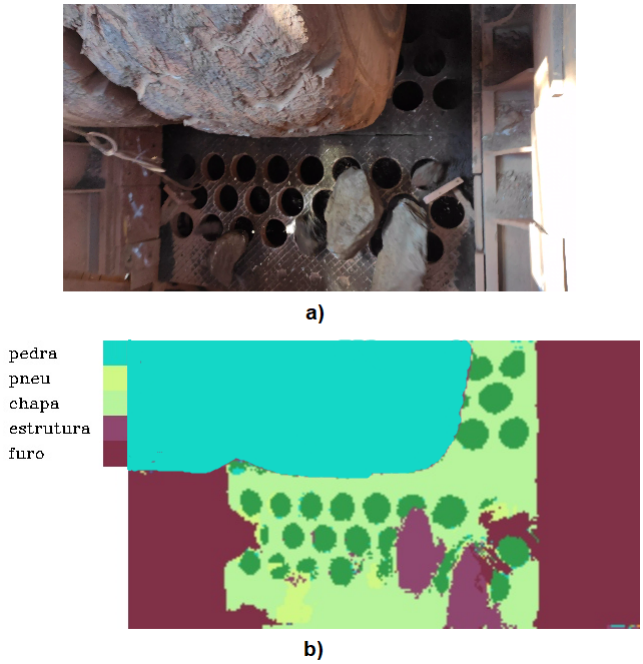


Figura 5.15: Segmentação produzida pela rede U-Net: a) imagem original fornecida como entrada da rede b) Segmentação gerada pela rede e classificação das áreas em classes.

Uma vez em posse das máscaras de identificação das áreas, para detecção de objetos foi desenvolvida uma rotina iterativa que percorre as áreas das máscaras e determina

os contornos equivalentes. Com os contornos definidos, é possível buscar as posições lineares das mesmas na imagem e estabelecer *bounding boxes* que delimitam cada objeto. A biblioteca de visão computacional *OpenCV* tem funções implementadas que executam tais tarefas e foram utilizadas neste trabalho. Com as *bounding boxes* definidas, a medição das dimensões lineares são realizadas com a contagem dos pixels das caixas de detecção e conversão em relação a medidas conhecidas na imagem. A Figura 5.16 exibe as etapas da segmentação até a detecção de objetos.

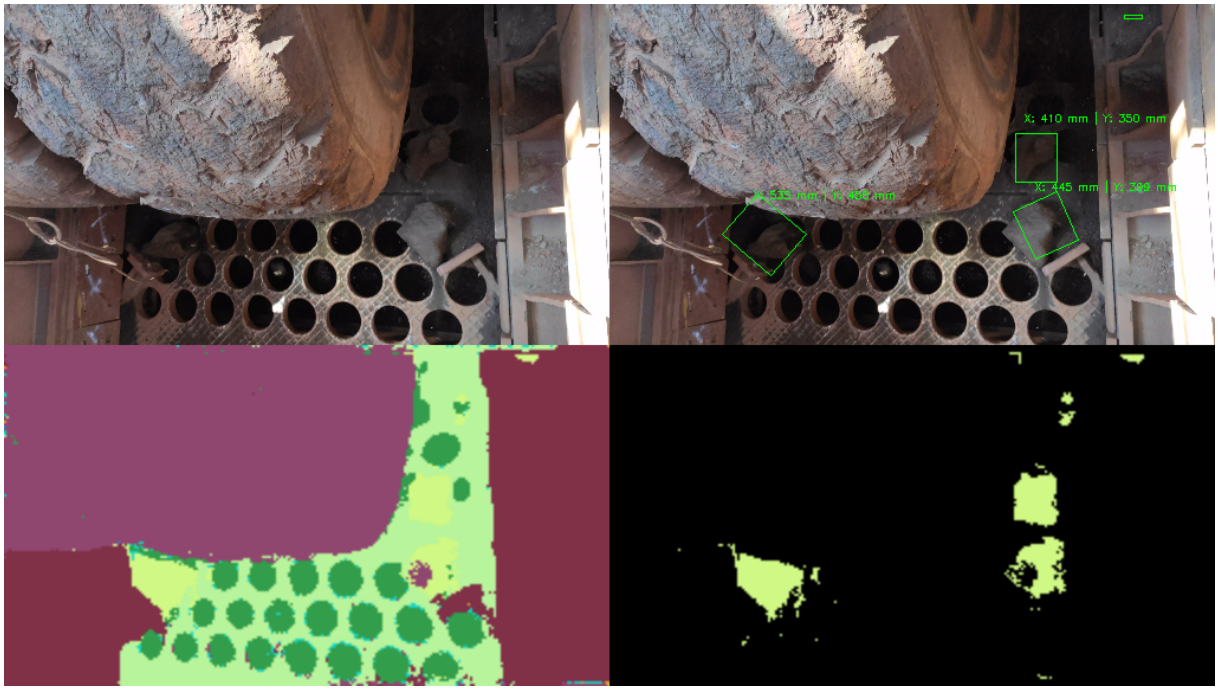


Figura 5.16: Resultados de detecção de objetos utilizando a rede U-Net. Da esquerda para direita em sentido anti-horário: 1) Imagem original fornecida como entrada da rede 2) Segmentação produzida pela rede U-Net 3) Classe "pedra"separada das demais 4) Criação de *bounding boxes* com as medidas equivalentes dos objetos.

A Tabela 5.2 mostra a distribuição estatística dos resultados de detecção de objetos da rede utilizando a base de dados de treinamento e validação. A métrica utilizada foi a de Intersecção sobre União (IoU). Os resultados mostram a média, mediana e desvio padrão dos resultados. A Figura 5.17 mostra um exemplo da forma como o cálculo é realizado com a comparação da caixa de detecção de objeto gerada via modelo (caixa verde) e uma caixa que representa a dimensão real do objeto detectado (caixa vermelha).

A Tabela 5.3 mostra a distribuição estatística dos resultados de detecção de objetos da rede utilizando novas imagens, diferentes do conjunto utilizado para treinamento e validação, utilizando o mesmo indicador IoU. A conclusão é que o desempenho do modelo para o conjunto de treinamento e validação é melhor do que um conjunto de imagens independente em função do número limitado de imagens utilizadas para construção do modelo e da baixa variabilidade. Poucas variações no cenário mostram um desempenho pior na capacidade do modelo em classificar corretamente as imagens e consequentemente

Imagem	Média	Mediana	Desvio Padrão
1	0,713	0,667	0,070
2	0,691	0,657	0,075
3	0,712	0,698	0,071
4	0,695	0,665	0,073
5	0,688	0,637	0,113
6	0,683	0,641	0,095
7	0,722	0,713	0,088
8	0,693	0,678	0,091
9	0,733	0,699	0,091
10	0,716	0,705	0,064
Geral	**0,704**	**0,678**	**0,081**

Tabela 5.2: Testes com rede U-Net utilizando dataset de treinamento e validação

Imagem	Média	Mediana	Desvio Padrão
1	0,642	0,648	0,041
2	0,643	0,667	0,060
3	0,657	0,668	0,071
4	0,624	0,623	0,078
5	0,689	0,703	0,120
6	0,625	0,602	0,094
7	0,624	0,622	0,088
8	0,619	0,604	0,084
9	0,510	0,480	0,118
10	0,557	0,549	0,043
Geral	**0,623**	**0,620**	**0,094**

Tabela 5.3: Testes com rede U-Net utilizando novo dataset, com imagens diferentes do conjunto de treinamento e validação

o algoritmo de geração das caixas de localização tem um desempenho prejudicado.

A Seção A.1.4 apresenta mais exemplos de detecção de objetos utilizando a rede U-Net.

5.2.2. Discussão dos Resultados

A rede U-Net apresenta resultados mais positivos em relação às demais redes estudadas e implementadas, especialmente pelo índice elevado de precisão no treinamento e validação. A característica mais simples da rede na profundidade da rede e a ausência de uma estrutura específica para classificação e outra para detecção de objetos torna a mesma mais eficiente na minimização do erro e formulação de um modelo eficiente para classificação.

Porém a qualidade do modelo é diretamente ligada a variância dos elementos que compõem o *dataset*. As poucas imagens analisadas e rotuladas apresentam características

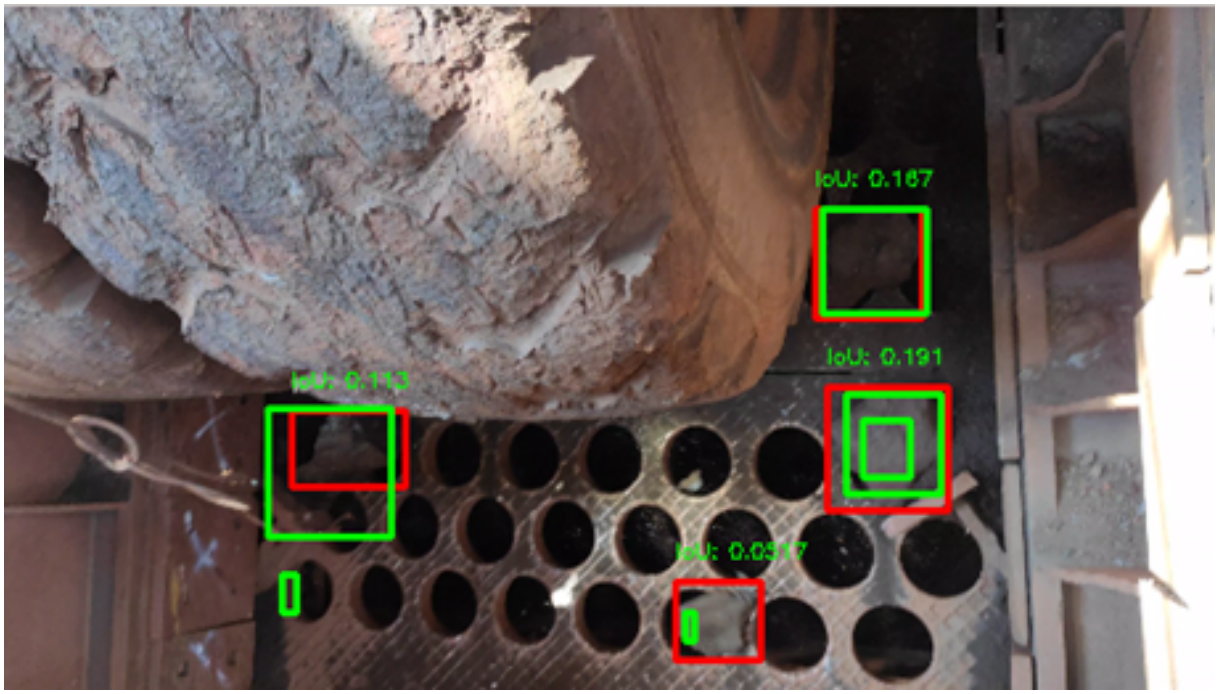


Figura 5.17: Resultados de detecção de objetos utilizando a rede U-Net e avaliação utilizando indicador IoU (Interseção sobre União).

similares entre massas de treinamento, validação e testes de detecção em quesitos como iluminação, posição e enquadramento e distribuição das cores. Para uma operação industrial, é necessário buscar diversas condições ambientais para elaborar um modelo robusto o suficiente para mapear corretamente as máscaras das classes. O trabalho elaborado por Ronneberger *et al.* (2015) foi idealizado para a classificação de imagens microscópicas de exames médicos ou mapas com variação limitada de classes. Pela característica construtiva da rede, uma mesma classe com maior variabilidade entre seus elementos pode gerar resultados menos eficientes na classificação.

Apesar dos resultados com níveis elevados de acurácia no treinamento, ocorreram diversas ocorrências de falsos-negativos e falsos-positivos na etapa de testes de detecção. Uma razão é a incapacidade da rede em detectar tons de cores muito semelhantes no contexto das imagens da britagem primária, onde os tons da própria estrutura do equipamento são semelhantes aos das partículas de minério. Uma biblioteca com mais amostras, com maior variância, podem levar a melhores resultados.

Para este modelo foi utilizada a regularização do tipo L2 (ou decaimento de pesos) que busca a redução dos pesos ao longo da minimização da função de perda. Os resultados encontrados na acurácia e a comparação entre treinamento e validação mostram um desempenho melhor do que os modelos anteriormente analisados.

5.3. Comparação entre modelos: utilização de recursos computacionais para treinamento e detecção de objetos

A Tabela 5.4 exibe uma comparação entre os tempos de treinamento entre os modelos implantados utilizando a estrutura definida na Seção 4.4.1. A rede YOLOv3 foi a rede mais rápida no tempo total. Coerente em função de ser a que teve o menor número de passos de treinamento. A rede U-Net foi a que gastou menos tempo por passo, o que é esperado pela estrutura mais simples da rede. A rede SSD foi a que levou o menor tempo para concluir uma época de treinamento, pois tem a estrutura mais simples em comparação a Faster R-CNN e YOLOv3.

	Tam. Dataset	**Passos**	**Épocas**	**Tempo de Trein. (h)**	**Segundos/ Época**	**Passos/ s**
SSD	117	106.301	909	24,3	96,24	1,2
Faster R-CNN	117	150.501	1286	46,3	129,6	0,9
YOLOv3	117	11.700	100	7,5	270	0,43
U-Net	128	20.480	40	13,5	1215	0,42

Tabela 5.4: Comparação entre tempos de treinamentos dos modelos SSD, Faster R-CNN, YOLOv3 e U-Net

A Tabela 5.5 exibe os tempos gastos na etapa de testes para detecção de objetos, utilizando o ambiente computacional descrito na Seção 4.4.1. A rede YOLOv3 apresenta o melhor desempenho nos testes de detecção de objeto. A rede U-Net mesmo utilizando algoritmos para criação das caixas de detecção fora de redes neurais convolucionais tem desempenho melhor do que as rede Faster R-CNN e SSD. Devem ser levadas em consideração as características de implementação e possíveis otimizações de código e também as configurações de *hardware* para treinamento e testes.

	Tempo para detecção (ms)
SSD	320
Faster R-CNN	404
YOLOv3	203
U-Net	278

Tabela 5.5: Comparação entre tempos de detecção dos modelos SSD, Faster R-CNN, YOLOv3 e U-Net

6. Conclusão

Neste trabalho foram apresentadas técnicas de classificação de imagens e detecção de objetos que podem ser aplicadas no contexto industrial para medição de granulometria de minério de ferro. Em comparação com as soluções atualmente comercializadas que exploram principalmente técnicas clássicas de processamento digital de imagens e necessitam de grande controle ambiental como iluminação, uniformidade e disposição espacial, os métodos apresentados se propõem ser robustos o bastante para interpretar condições variadas de ambiente e fornecer as saídas desejadas.

As redes SSD, Faster R-CNN e YOLOv3 não apresentaram resultados adequados para o contexto estudado, apesar de seu grande uso em aplicações em tempo real e em dispositivos móveis como identificação de pessoas e veículos. A implementação utilizando a API TensorFlow das redes SSD e Faster R-CNN pode ser otimizada para ter melhor uso de recursos computacionais. Além disso devem ser exploradas variações das estruturas *backbone* para buscar melhorias na acurácia e precisão na classificação e localização de objetos. Da mesma forma, devem ser exploradas as possibilidades de otimização na implementação da rede YOLOv3.

A rede U-Net é de simples implementação, com menor número de operações convolucionais e tem boa acurácia no *dataset* elaborado para este trabalho, com valores acima de 90%. Entretanto, como ponto comum em relação às demais redes, a elaboração de uma biblioteca com a variabilidade entre as amostras é vital para a qualidade na classificação e detecção. O processo de rotulação de imagens para construção das bibliotecas depende de grande análise manual, devido à necessidade de estabelecer as informações corretas para treinamento e da natural necessidade de utilizar um elevado número de amostras para coletar todas as características que são decisivas para uma correta classificação. Técnicas como a utilização de algoritmos de clusterização de superpixels para acelerar o processo de rotulação devem ser consideradas.

6.1. Trabalhos Futuros

Trabalhos futuros nas melhorias e otimizações do código são necessárias para reduzir o número de falsos-positivos e falsos-negativos e aumentar os índices de eficiência na classificação de imagens, conforme discutido ao longo do Capítulo 5 e na Seção anterior.

Além da medição de granulometria, a medição de volume de partículas também é igualmente importante e vários métodos abordados neste texto podem ser aproveitados para criação de um método para medição em 3 dimensões. Aliado às técnicas de aprendizado de máquina, o estudo sobre câmeras multi-oculares com diferenças de distâncias focais pode ser utilizado como coletor de dados para alimentar tais modelos.

Para medição da granulometria em tempo real na britagem primária na unidade

de Vargem Grande, deverá ser montada uma câmera para aquisição de imagens e implantado o código discutido neste trabalho em controladores industriais ou computadores para aquisição e tratamento dos dados. As características da geração da biblioteca discutidas anteriormente devem ser levadas em consideração de acordo com as características ambientais e dos equipamentos disponíveis na unidade operacional.

Aprimoramentos no tratamento de dados devem ser levados em consideração como análise da oclusão de objetos na imagem, tratamento adequado de rastreamento de objetos detectados, técnicas de amostragem de imagens para processamento em tempo real de forma a manter a representatividade da informação de material que passa pelo equipamento ao longo do tempo e tipos de câmeras e iluminação podem ser utilizadas em tal ambiente de forma a reduzir o impacto na complexidade da classificação ao uniformizar condições ambientais e trazer o maior nível de informações possíveis para o treinamento dos modelos.

Referências Bibliográficas

ABRAMOVICH, F., PENSKY, M. "Classification with many classes: Challenges and pluses", *Journal of Multivariate Analysis*, v. 174, pp. 104536, 2019.

BODLA, N., SINGH, B., CHELLAPPA, R., et al.. "Soft-NMS–Improving Object Detection With One Line of Code". Em: *Proceedings of the IEEE international conference on computer vision*, pp. 5561–5569, 2017.

CHIAN, E. "Calculating Loss of Yolo (v3) Layer". 2019. Disponível em: <https://towardsdatascience.com/calculating-loss-of-yolo-v3-layer-8878bfaaf1ff>.

CRUZ, J., SHIGUEMORI, E., GUIMARAES, L. "Comparação entre HOG SVM e Haar-like em cascata para a detecção de campos de futebol em imagens aéreas e orbitais", *Anais XVI Simpósio Brasileiro de Sensoriamento Remoto-SBSR, Foz do Iguaçu, PR, Brasil*, v. 13, pp. 6917–6922, 2013.

DAI, J., LI, Y., HE, K., et al.. "R-fcn: Object detection via region-based fully convolutional networks". Em: *Advances in neural information processing systems*, pp. 379–387, 2016.

DALAL, N., TRIGGS, B. "Histograms of oriented gradients for human detection". 2005.

DEL VILLAR, R., THIBAULT, J., DEL VILLAR, R. "Development of a softsensor for particle size monitoring", *Minerals Engineering*, v. 9, n. 1, pp. 55–72, 1996.

DI, Z., YANG, F., CAO, Y., et al.. "Optimization of particle size distribution in circulating fluidized beds via adjustment of crushers and tuning parameters of two-toothed roll crusher", *Powder Technology*, v. 352, pp. 151–158, 2019.

EVERINGHAM, M., WINN, J. "The pascal visual object classes challenge 2012 (voc2012) development kit", *Pattern Analysis, Statistical Modelling and Computational Learning, Tech. Rep*, v. 8, 2011.

FELZENSZWALB, P. F., GIRSHICK, R. B., MCALLESTER, D. "Cascade object detection with deformable part models". Em: *2010 IEEE Computer Society Con-*

ference on Computer Vision and Pattern Recognition, pp. 2241–2248. IEEE, 2010.

GIRSHICK, R. "Fast r-cnn". Em: *Proceedings of the IEEE international conference on computer vision*, pp. 1440–1448, 2015.

GIRSHICK, R., DONAHUE, J., DARRELL, T., et al.. "Rich feature hierarchies for accurate object detection and semantic segmentation". Em: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 580–587, 2014.

GOODFELLOW, I., BENGIO, Y., COURVILLE, A. *Deep Learning*. MIT Press, 2016. `http://www.deeplearningbook.org`.

GUPTA, A., YAN, D. S. *Mineral processing design and operations: an introduction*. Elsevier, 2016.

HAMZELOO, E., MASSINAEI, M., MEHRSHAD, N. "Estimation of particle size distribution on an industrial conveyor belt using image analysis and neural networks", *Powder technology*, v. 261, pp. 185–190, 2014.

HE, K., ZHANG, X., REN, S., et al.. "Spatial pyramid pooling in deep convolutional networks for visual recognition", *IEEE transactions on pattern analysis and machine intelligence*, v. 37, n. 9, pp. 1904–1916, 2015.

HE, K., ZHANG, X., REN, S., et al.. "Deep residual learning for image recognition". Em: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.

HE, K., GKIOXARI, G., DOLLÁR, P., et al.. "Mask r-cnn". Em: *Proceedings of the IEEE international conference on computer vision*, pp. 2961–2969, 2017.

HIJAZI, S., KUMAR, R., ROWEN, C. "Using convolutional neural networks for image recognition", *Cadence Design Systems Inc.: San Jose, CA, USA*, pp. 1–12, 2015.

HUANG, J., RATHOD, V., SUN, C., et al.. "Speed/accuracy trade-offs for modern convolutional object detectors". Em: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 7310–7311, 2017.

JEMWA, G. T., ALDRICH, C. "Estimating size fraction categories of coal particles on conveyor belts using image texture modeling methods", *Expert Systems with Applications*, v. 39, n. 9, pp. 7947–7960, 2012.

KAARTINEN, J., TOLONEN, A. "Utilizing 3D height measurement in particle size analysis", *IFAC Proceedings Volumes*, v. 41, n. 2, pp. 3292–3297, 2008.

KATURIA, A. "What's new in YOLO v3?" 2018. Disponível em: <https://towardsdatascience.com/yolo-v3-object-detection-53fb7d3bfe6b>.

KRIZHEVSKY, A., HINTON, G. "Convolutional deep belief networks on cifar-10", *Unpublished manuscript*, v. 40, n. 7, pp. 1–9, 2010.

KRIZHEVSKY, A., SUTSKEVER, I., HINTON, G. E. "Imagenet classification with deep convolutional neural networks". Em: *Advances in neural information processing systems*, pp. 1097–1105, 2012.

LI, F.-F., JOHNSON, J., YEUNG, S. "Cs231n: convolutional neural networks for visual recognition (2016)", *URL http://cs231n. stanford. edu*, v. 37, 2016.

LIAO, C., TARNG, Y. "On-line automatic optical inspection system for coarse particle size distribution", *Powder Technology*, v. 189, n. 3, pp. 508–513, 2009.

LIN, T.-Y., DOLLÁR, P., GIRSHICK, R., et al.. "Feature pyramid networks for object detection". Em: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2117–2125, 2017.

LIU, W., ANGUELOV, D., ERHAN, D., et al.. "Ssd: Single shot multibox detector". Em: *European conference on computer vision*, pp. 21–37. Springer, 2016.

NAJIBI, M., RASTEGARI, M., DAVIS, L. S. "G-cnn: an iterative grid based object detector". Em: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2369–2377, 2016.

NWANKPA, C., IJOMAH, W., GACHAGAN, A., et al.. "Activation functions: Comparison of trends in practice and research for deep learning", *arXiv preprint arXiv:1811.03378*, 2018.

QUIST, J., EVERTSSON, C. M. "Cone crusher modelling and simulation using DEM", *Minerals Engineering*, v. 85, pp. 92–105, 2016.

RAMÍREZ CERNA, L. "Fusão de descritores de histogramas de gradientes para a detecção de faces baseado em uma cascata de classificadores." 2014.

REDMON, J., FARHADI, A. "Yolov3: An incremental improvement", *arXiv preprint arXiv:1804.02767*, 2018.

REDMON, J., DIVVALA, S., GIRSHICK, R., et al.. "You only look once: Unified, real-time object detection". Em: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 779–788, 2016.

REN, S., HE, K., GIRSHICK, R., et al.. "Faster r-cnn: Towards real-time object detection with region proposal networks". Em: *Advances in neural information processing systems*, pp. 91–99, 2015.

RONNEBERGER, O., FISCHER, P., BROX, T. "U-net: Convolutional networks for biomedical image segmentation". Em: *International Conference on Medical image computing and computer-assisted intervention*, pp. 234–241. Springer, 2015.

ROSEBROCK, A. "Histogram of Oriented Gradients and Object Detection". 2014. Disponível em: <`https://www.pyimagesearch.com/2014/11/10/histogram-oriented-gradients-object-detection/`>.

ROSEBROCK, A. *Deep Learning for Computer Vision with Python: Starter Bundle*. Pyimagesearch, 2017.

SIMONYAN, K., ZISSERMAN, A. "Very deep convolutional networks for large-scale image recognition", *arXiv preprint arXiv:1409.1556*, 2014.

SZEGEDY, C., REED, S., ERHAN, D., et al.. "Scalable, high-quality object detection", *arXiv preprint arXiv:1412.1441*, 2014.

SZEGEDY, C., LIU, W., JIA, Y., et al.. "Going deeper with convolutions". Em: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1–9, 2015.

SZEGEDY, C., VANHOUCKE, V., IOFFE, S., et al.. "Rethinking the inception architecture for computer vision". Em: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2818–2826, 2016.

TANG, Y. "Deep learning using linear support vector machines", *arXiv preprint arXiv:1306.0239*, 2013.

TAYLOR, C. C., PIZLO, Z., ALLEBACH, J. P., et al.. "Image quality assessment with a Gabor pyramid model of the human visual system". Em: *Human Vision and Electronic Imaging II*, v. 3016, pp. 58–69. International Society for Optics and Photonics, 1997.

THARWAT, A. "Classification assessment methods", *Applied Computing and Informatics*, 2020.

THURLEY, M. J., ANDERSSON, T. "An industrial 3D vision system for size measurement of iron ore green pellets using morphological image segmentation", *Minerals engineering*, v. 21, n. 5, pp. 405–415, 2008.

UDOFIA, U. "Basic Overview of Convolutional Neural Network (CNN)". 2018. Disponível em: <https://medium.com/dataseries/basic-overview-of-convolutional-neural-network-cnn-4fcc7dbb4f17>.

UIJLINGS, J. R., VAN DE SANDE, K. E., GEVERS, T., et al.. "Selective search for object recognition", *International journal of computer vision*, v. 104, n. 2, pp. 154–171, 2013.

VALE. "Memorial Descritivo do Processo - Beneficiamento de Itabiritos - Projeto Vargem Grande Itabiritos", *Documento interno nº MF-2000VG-P-00101, Vale S.A.*, 2012.

VARGAS, A. C. G., PAES, A., VASCONCELOS, C. N. "Um estudo sobre redes neurais convolucionais e sua aplicação em detecção de pedestres". Em: *Proceedings of the XXIX Conference on Graphics, Patterns and Images*, v. 1, 2016.

VIOLA, P., JONES, M., OTHERS. "Rapid object detection using a boosted cascade of simple features", *CVPR (1)*, v. 1, n. 511-518, pp. 3, 2001.

VON WANGENHEIM, A. "Deep Learning::Segmentação con CNNs". 2019a. Disponível em: <http://www.lapix.ufsc.br/ensino/visao/visao-computacionaldeep-learning/deep-learningsegmentacao-semantica/>.

VON WANGENHEIM, A. "Avaliando, Validando e Testando o seu Modelo: Metodologias de Avaliação de Performance". 2019b.

WILLIAMS, R., LUKE, S., OSTROWSKI, K., et al.. "Measurement of bulk particulates on belt conveyor using dielectric tomography", *Chemical Engineering Journal*, v. 77, n. 1-2, pp. 57–63, 2000.

WILLS, B. A., FINCH, J. *Wills' mineral processing technology: an introduction to the practical aspects of ore treatment and mineral recovery*. Butterworth-Heinemann, 2015.

YANG, H., SHAO, L., ZHENG, F., et al.. "Recent advances and trends in visual tracking: A review", *Neurocomputing*, v. 74, n. 18, pp. 3823–3831, 2011.

YANG, M.-H., ROTH, D., AHUJA, N. "A SNoW-based face detector". Em: *Advances in neural information processing systems*, pp. 862–868, 2000.

YOO, D., PARK, S., LEE, J.-Y., et al.. "Attentionnet: Aggregating weak directions for accurate object detection". Em: *Proceedings of the IEEE International Conference on Computer Vision*, pp. 2659–2667, 2015.

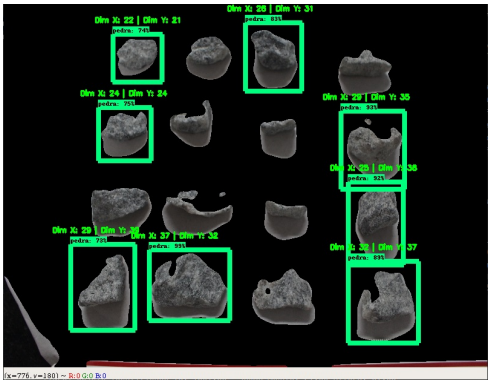
ZHAO, Z.-Q., ZHENG, P., XU, S.-T., et al.. "Object detection with deep learning: A review", *IEEE transactions on neural networks and learning systems*, v. 30, n. 11, pp. 3212–3232, 2019.

A. Testes de Detecção de Objetos

A.1. Testes de detecção de imagens com os modelos SSD, Faster R-CNN e YOLOv3

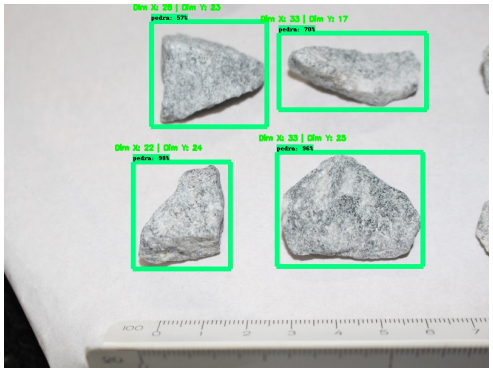
A.1.1. Detecção de objetos utilizado o modelo SSD

(a) Imagem com fundo branco afastada

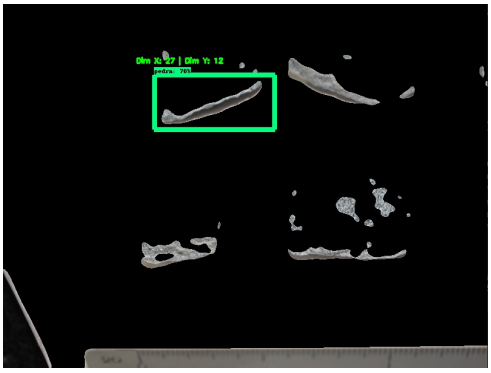


(b) Imagem com fundo branco afastada com pré pro-processamento

Figura A.1: Teste com modelo SSD - Fundo branco - Afastada



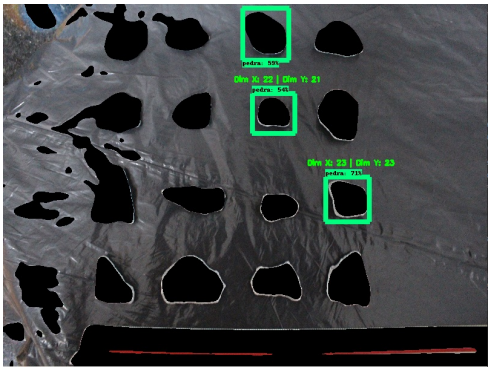
(a) Imagem com fundo branco aproximada



(b) Imagem com fundo branco aproximada com pré-processamento

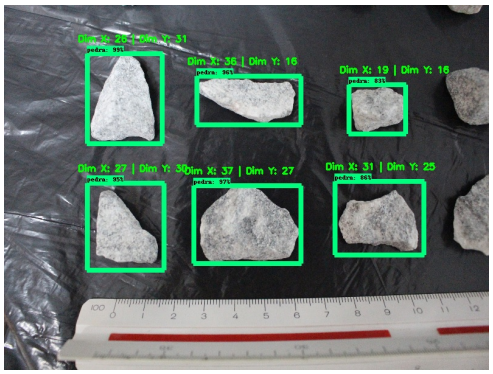
Figura A.2: Teste com modelo SSD - Fundo branco - Aproximada

(a) Imagem com fundo preta reduzida

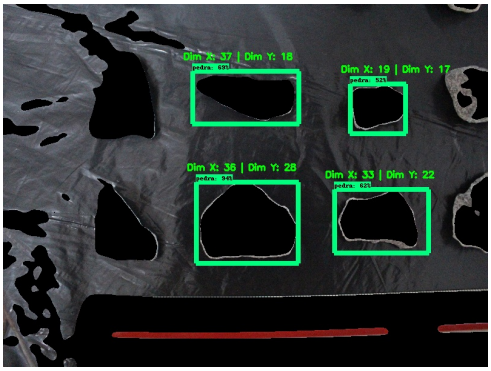


(b) Imagem com fundo preto reduzida com pré-processamento

Figura A.3: Teste com modelo SSD - Fundo preto - Reduzida



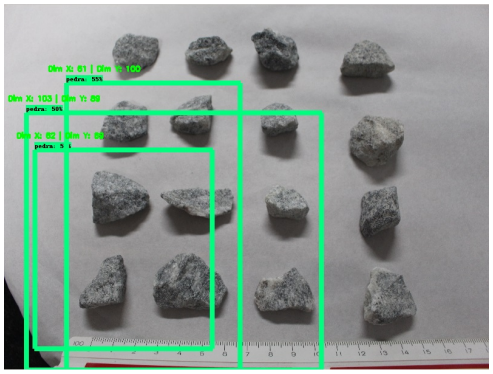
(a) Imagem com fundo preta ampliada



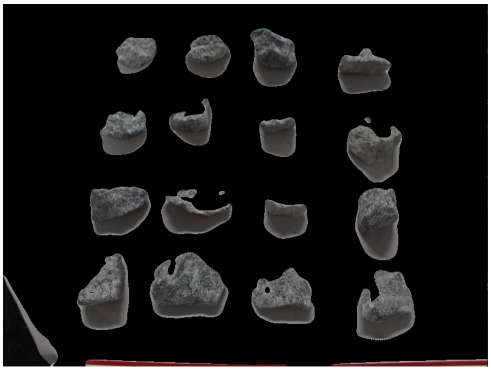
(b) Imagem com fundo preto ampliada com pré-processamento

Figura A.4: Teste com modelo SSD - Fundo preto - Ampliada

A.1.2. Detecção de objetos utilizado o modelo Faster R-CNN



(a) Imagem com fundo branco afastada



(b) Imagem com fundo branco afastada com pré pro-processamento

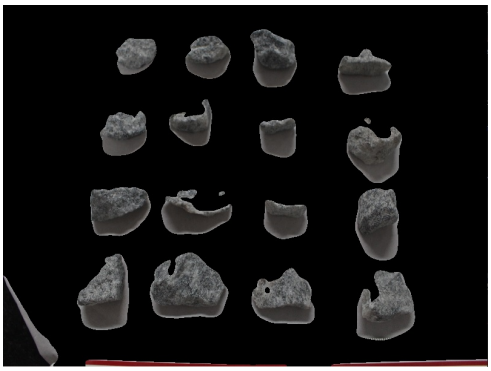
Figura A.5: Teste com modelo Faster R-CNN - Fundo branco - Afastada

(a) Imagem com fundo branco aproximada

(b) Imagem com fundo branco aproximada com pré-processamento

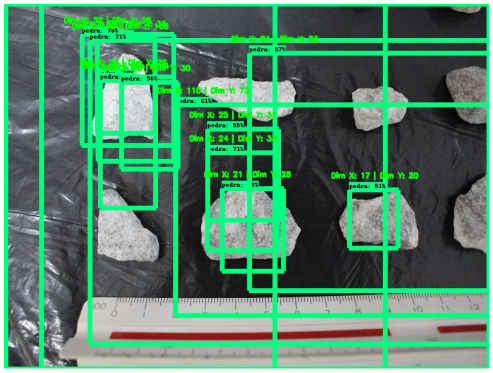
Figura A.6: Teste com modelo Faster R-CNN - Fundo branco - Aproximada

(a) Imagem com fundo preta reduzida

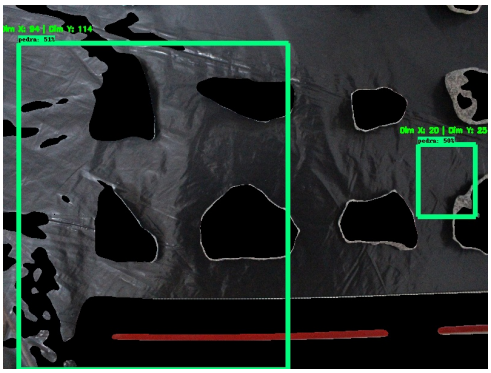


(b) Imagem com fundo preto reduzida com pré-processamento

Figura A.7: Teste com modelo Faster R-CNN - Fundo preto - Reduzida



(a) Imagem com fundo preta ampliada

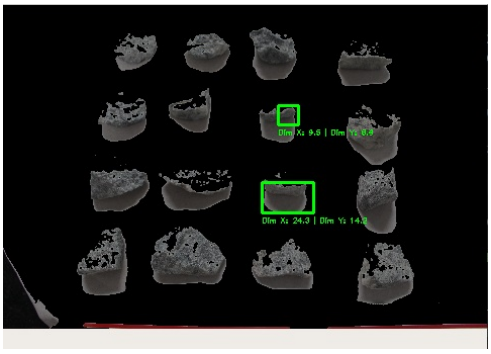


(b) Imagem com fundo preto ampliada com pré-processamento

Figura A.8: Teste com modelo Faster R-CNN - Fundo preto - Ampliada

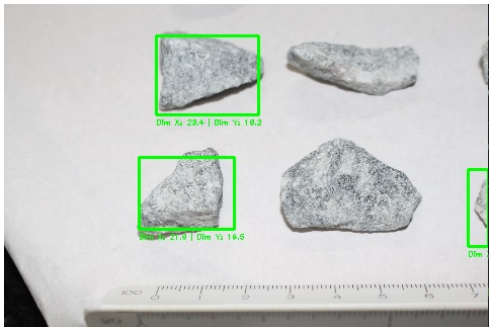
A.1.3. Detecção de objetos utilizado o modelo YOLOv3

(a) Imagem com fundo branco afastada

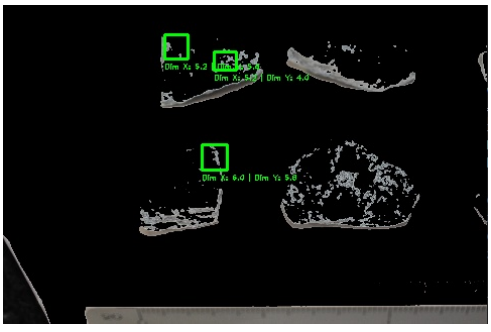


(b) Imagem com fundo branco afastada com pré pro-processamento

Figura A.9: Teste com modelo YOLOv3 - Fundo branco - Afastada



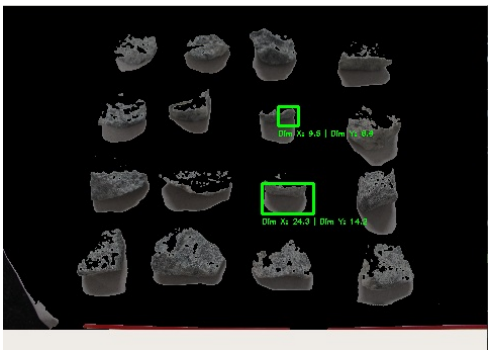
(a) Imagem com fundo branco aproximada



(b) Imagem com fundo branco aproximada com pré-processamento

Figura A.10: Teste com modelo YOLOv3 - Fundo branco - Aproximada

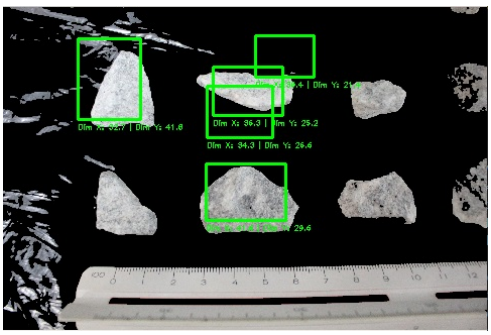
(a) Imagem com fundo preta reduzida



(b) Imagem com fundo preto reduzida com pré-processamento

Figura A.11: Teste com modelo YOLOv3 - Fundo preto - Reduzida

(a) Imagem com fundo preta ampliada



(b) Imagem com fundo preto ampliada com pré-processamento

Figura A.12: Teste com modelo YOLOv3 - Fundo preto - Ampliada

A.1.4. Detecção de objetos utilizado o modelo U-Net

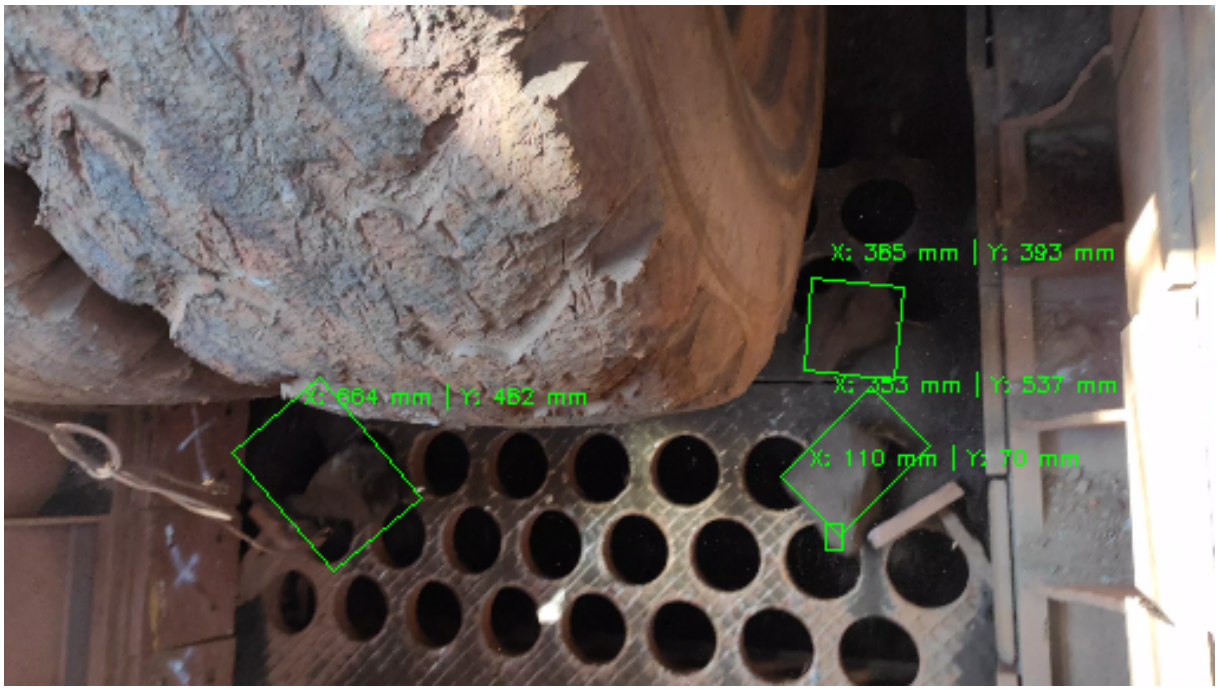


Figura A.13: Resultados de detecção de objetos utilizando a rede U-Net.

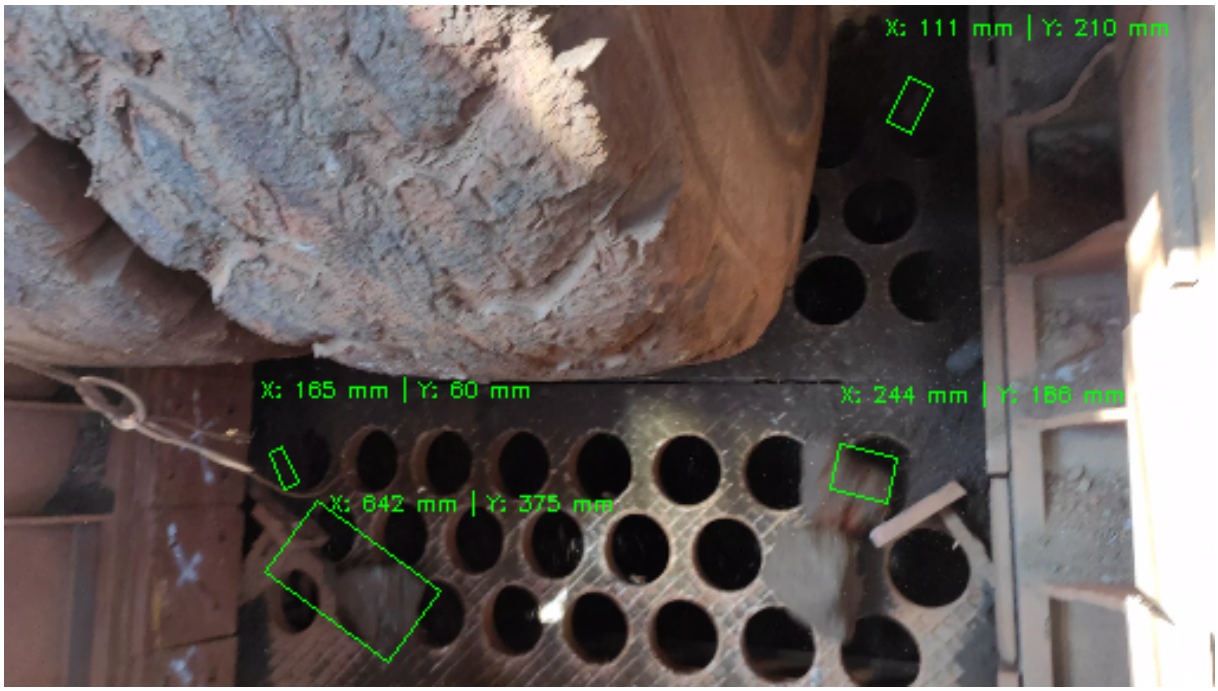


Figura A.14: Resultados de detecção de objetos utilizando a rede U-Net.

Figura A.15: Resultados de detecção de objetos utilizando a rede U-Net.