RASTREAMENTO DE POSES HUMANAS A PARTIR DE ENTRADAS RGB

RICARDO ROSSITER BARIONI

Trabalho de Graduação

RECIFE/2018

Ricardo Rossiter Barioni

RASTREAMENTO DE POSES HUMANAS A PARTIR DE ENTRADAS RGB

Trabalho apresentado ao Curso de Graduação em Ciência da Computação do Centro de Informática da Universidade Federal de Pernambuco como requisito parcial para obtenção do grau de Bacharel em Ciência da Computação.

Orientador: Veronica Teichrieb
Co-Orientador: Lucas Silva Figueiredo, Kelvin Batista da Cunha

RECIFE
2018

Trabalho de graduação apresentado por **Ricardo Rossiter Barioni** ao curso de Graduação em Ciência da Computação do Centro de Informática da Universidade Federal de Pernambuco, sob o título **Rastreamento de Poses Humanas a Partir de Entradas RGB**, orientado pelo **Prof. Veronica Teichrieb** e aprovado pela banca examinadora formada pelos professores:

Prof. Hansenclever de França Bassani
Centro de Informática/UFPE

Prof. Veronica Teichrieb
Centro de Informática/UFPE

RECIFE
2018

Dedico este trabalho a meus pais, Maria de Fátima e Ricardo, e a meu irmão, Daniel, os quais estiveram ao meu lado sempre.

Agradecimentos

Gostaria de agradecer primeiramente à minha família: a Fátima, Ricardo e Daniel, por estarem do meu lado durante esta jornada. Muitas vezes, cheguei tarde em casa, e compreenderam o esforço que investi neste projeto.

Também agradeço à professora Veronica, a Lucas, a Kelvin e a Willams, por me darem um imenso suporte na orientação e execução deste projeto.

Agradeço imensamente à família Voxar Labs, por estarem me dando um grande apoio, tanto tecnicamente quanto emocionalmente. Um agradecimento especial a Eduardo e Caio.

Agradeço a meus amigos e amigas, por me deixarem tranquilo nos momentos de maior ansiedade, e por me alegrarem nos momentos mais difíceis. Em especial, um agradecimento aos "Porteiros": Hilton, Janjo, Xinho, Miranda, Freitas, Buba, Brandão e Migge.

The only thing we have to fear is fear itself.

—FRANKLIN D. ROOSEVELT

Resumo

No contexto de interação natural e rastreamento de seres humanos, é fundamental a obtenção da configuração das poses humanas. A obtenção desta pose a partir de imagens RGB, estas obtidas através de câmeras, traz a possibilidade de uma extensa gama de aplicações para áreas como segurança (ex.: monitoramento de atividade no local), saúde (ex.: análise postural) e entretenimento (ex.: jogos e concepção de animações). Porém, isso é considerado um desafio, uma vez que dados puramente visuais não nos dão explicitamente informações a respeito da localização das juntas (*keypoints* em pixels) do corpo humano na imagem. Neste trabalho, propõe-se o desenvolvimento de um método de aprendizagem de máquina (especificamente aprendizagem profunda baseada em redes convolucionais) capaz de abordar tal problemática, tendo como motivação o seu uso no contexto de aplicações do mundo real como comentado acima. Para a realização do treinamento da rede neural, também foi proposta uma nova função de erro, baseada na função de erro médio quadrado. Foram obtidos valores de *Precision* e *Recall* de 29,22% e 36,87%, respectivamente, para um valor de limiar de distância de 20 *pixels*.

Palavras-chave: Interação Natural, Poses Humanas, Aprendizagem de Máquina, Redes Neurais Convolucionais

Abstract

In the context of natural interaction and human body tracking, it is fundamental to obtain the human poses configuration. Obtain this pose from RGB images through cameras brings the possibility of a wide set of applications in the areas of security (i.e.: local activity monitoring), healthcare (i.e.: postural analysis) and retail (i.e.: games and animations conception). However, this is considered a challenge, once that pure visual data doesn't explicitly give us information about the human body joints (Keypoints in pixels) localization in the image. In this work, it is proposed the development of a machine learning method (more specifically deep learning based on convolutional networks) capable of tackling this problem, while having as a motivation its usage in the context of real-world applications. For performing neural network training, it was also proposed a new loss function, based on mean squared error function. This method achieved a Precision of 29,22% and a Recall of 36,87%, for a 20 pixels distance threshold.

Keywords: Natural Interaction, Human Poses, Machine Learning, Convolutional Neural Networks

Lista de Figuras

Lista de Tabelas

Sumário

1

Introdução

O rastreamento de poses humanas diz respeito à obtenção de informações espaciais de seres humanos em um determinado ambiente. Existem diversas abordagens, bem como uma vasta quantidade de recursos tecnológicos os quais auxiliam na realização desta atividade. Considerando isso, as diversas aplicações existentes para o rastreamento de poses definem o nível de complexidade e custo do método aplicado.

Na maioria das aplicações, as informações espaciais das poses humanas que se deseja obter resume-se a, para cada ser humano presente na detecção, fornecer um conjunto de posições das juntas (também denominados *keypoints*). Estas juntas são os pontos mais característicos do corpo humano, e geralmente são pontos de articulações, tais como ombros, cotovelos, joelhos e extremidades. A representação do corpo desta maneira é simples e suficiente para compreender a configuração de pose de um corpo humano.

Em grande parte dos métodos existentes, é necessário o uso de informações visuais, com o predomínio de duas categorias:

1. **Imagem RGB:** É o tipo de representação visual a qual busca replicar a informação visual extraída pelo ser humano. Está presente em diversos aparelhos do cotidiano, tais como câmeras fotográficas, câmeras de dispositivos móveis, câmeras de segurança, dentre outros. Definida como uma matriz, cada célula da matriz (também denominada pixel) possui informações a respeito da sua cor a partir de três variáveis:
    (a) **R:** Define a tonalidade de vermelho (*Red*) do pixel;
    (b) **G:** Define a tonalidade de verde (*Green*) do pixel;
    (c) **B:** Define a tonalidade de azul (*Blue*) do pixel;
2. **Imagem infravermelha:** É o tipo de representação visual a qual busca extrair características a respeito das radiações infravermelhas no espectro eletromagnético, e mapeá-las para um espectro visível. Ela está presente em dispositivos como o *Leap Motion* e o *Microsoft Kinect*. Também representável por uma matriz, cada pixel guarda a informação da estimativa da temperatura naquele ponto. É possível utilizar esses dados para estimar um mapa contendo informações a respeito da distância da câmera para as regiões da imagem.

Ao desenvolver um método cuja finalidade seja o rastreamento de poses humanas, deve-se levar em consideração uma série de desafios. Dentre esses, os principais são:

1. **Oclusão:** Em muitos casos, partes do corpo de um indivíduo podem estar ocluídas por algum elemento da cena, causando dificuldades em compreender se partes do corpo pertencem a uma mesma entidade (imagem à esquerda na Figura 1.1);

2. **Presença de múltiplas pessoas:** A presença de múltiplos indivíduos pode ocasionar equívocos no processo de detecção (imagem ao centro na Figura 1.1); partes do corpo de uma pessoa podem ser associadas a regiões de um outro indivíduo;

3. **Iluminação:** A qualidade de iluminação de uma imagem pode acarretar numa má extração de poses. Uma imagem com baixa iluminação, por exemplo, pode dificultar a detecção de juntas a partir de imagens RGB, impossibilitando a extração de poses (imagem à direita na Figura 1.1). Por outro lado, utilizando um sensor infravermelho para a captura da imagem, essa extração de poses pode ser dificultada em ambientes com elevado grau de iluminação natural.

Figura 1.1: Exemplos de obstáculos na obtenção de poses humanas; oclusão de parte do corpo humano (bola sobre o braço, à esquerda), presença de vários indivíduos na imagem (ao centro) e iluminação baixa (à direita).

1.1 Motivação

Nos últimos anos, contextos usando informações de poses humanas estão cada vez mais comuns: aplicações em realidade virtual e aumentada, jogos, animações de personagens, análise de comportamento de indivíduos em determinados ambientes e aplicações para o auxílio em atividades fisioterapêuticas são alguns dos principais exemplos onde necessidade de analisar configurações corporais pode estar presente e auxiliar nessas tarefas (MEHTA et al., 2017).

1.1.1 Jogos

Na área de jogos, existem diversas temáticas onde análise de movimento possibilita a concepção de interfaces que fazem sentido ao usuário. Um exemplo de gênero são jogos

de dança, como *Just Dance* e *Dance Central* (imagens à esquerda e à direita na Figura 1.2, respectivamente), onde os jogadores são instruídos a executar passos de dança, à medida em que a música é tocada. Os jogos avaliam a performance da dança feita pelo usuário a partir da sequência de movimentos executada.

Figura 1.2: Just Dance (esquerda) e Dance Central (direita).

Fontes: *Ubisoft* e *Harmonix*.

Um outro gênero de jogos são os de luta, no qual os movimentos do jogador mapeiam ações em cenários de luta (socos, chutes, esquiva, dentre outros). O jogo *Fighters Uncaged* (imagem à esquerda na Figura 1.3), por exemplo, remonta a uma cena de luta de rua, onde o jogador possui uma visão em terceira pessoa do jogo. Já o jogo *Kung-Fu Live* (imagem à direita na Figura 1.3) situa-se num cenário de plataforma 2D (similar ao jogo *Mortal Kombat*), onde o jogador é capaz de visualizar a si mesmo dentro do mundo do jogo.

Figura 1.3: *Fighters Uncaged* (esquerda) e *Kung-Fu Live* (direita).

Fontes: *Ubisoft* e *Virtual Air Guitar Company*.

Também há o gênero de jogos de esportes. Dentre eles, o mais conhecido é o Wii Sports (Figura 1.4). O Wii Sports consiste numa coletânea de jogos de vários esportes, tais como basebol, boxe, golfe, boliche e tênis. Diferentemente dos jogos citados anteriormente, este analisa o movimento do jogador a partir de um controle remoto, cuja posição e movimentação no espaço é realizada por uma câmera infravermelho. Com isso, é possível realizar a análise de ações que podem ser executadas pelo usuário, tais como tacadas (golfe e basebol), raquetadas (tênis), golpes (boxe) ou arremessos (boliche).

Figura 1.4: Wii Sports e suas modalidades.

Fonte: Nintendo.

1.1.2 Ambientes Inteligentes

Um outro contexto de uso de análise de movimento refere-se à concepção de ambientes (residências, escritórios, dentre outros), nos quais movimentos podem ser mapeados para executar ações. Com o avanço em áreas como sistemas embarcados e comunicação, surgiu o conceito de Internet das Coisas (IoT), o qual extende a capacidade de objetos comuns (eletrodomésticos, móveis, luzes, janelas, dentre outros) de comunicarem-se com sensores via Internet.

Um exemplo de projeto neste contexto é o *CityHome* (Figura 1.5). Desenvolvido pela *MIT Media Lab*, o *CityHome* objetiva concentrar um conjunto de utilidades de uma residência em um espaço reduzido; é possível ter acesso a mobílias, armários, iluminações e até mesmo sistemas de entretenimento através de gestos, utilizando apenas uma câmera *Kinect*.

1.1.3 Interação Humano-Robô

No contexto de robótica, também é possível modelar ações de robôs inteligentes a partir de movimentos realizados por um ser humano. Por exemplo, a partir da captura desses movimentos, é possível desenvolver robôs capazes de replicar o movimento humano, a partir da aprendizagem por imitação; esse tipo de aplicação pode ser utilizado no manuseio de equipamentos remotamente, como por exemplo:

1. Equipamentos para realização de cirurgias em pacientes (imagem à esquerda na Figura 1.6);

2. Equipamentos para construção de automóveis (imagem à direita na Figura 1.6).

1.1.4 Captura de Movimentos para Animações

Na indústria cinematográfica, é bastante comum o uso de equipamentos para realizar a captura de movimentos no desenvolvimento de personagens em desenhos animados. Em

Figura 1.5: Utilização do *CityHome* para a manipulação da luz no ambiente.

Fonte: *CityHome*.

Figura 1.6: Robô para a realização de cirurgias (esquerda) e robô para a montagem de carros em fábricas (direita).

Fonte: *Google*.

geral, são utilizados marcadores, os quais são posicionados em várias partes do corpo humano, bem com várias câmeras, com o intuito de garantir o mais refinado grau de precisão na captura possível. Feito isso, a sequência de movimentos capturados são aplicados ao modelo 3D do personagem, a fim de que tal modelo possa replicar tais ações.

Não só em relação à captura de movimentos, a reconstrução 3D facial também pode ser utilizada nesse contexto; com isso, é possível aplicar ao personagem expressões faciais mais fidedignas (Figura 1.7)). Esse tipo de abordagem possui uma potencial aplicabilidade quando se trata de dublagem de animações em múltiplos idiomas.

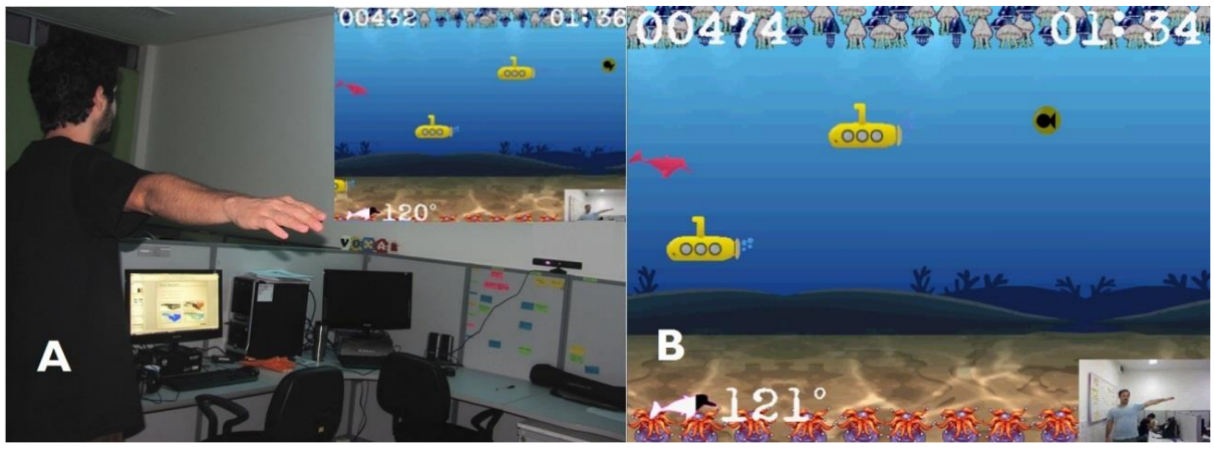
Figura 1.7: Captura de movimentos do personagem Smaug (direita), do filme O Hobbit: A Desolação de Smaug, interpretada pelo ator Benedict Cumberbatch (esquerda).

Fonte: *Google*.

1.1.5 Saúde

No contexto de fisioterapia, muitos trabalhos buscam mapear a captura de movimentos para um sistema de reabilitação motora. É o caso do Ikapp, um sistema de reabilitação motora o qual mapeia movimentos fisioterapêuticos realizados por um paciente para uma interface amigável, além de fazer uso de características de gamificação (Figura 1.8)). a fim de tornar a experiência de fisioterapia mais prazerosa e motivadora.

Figura 1.8: Uma das interfaces do Ikapp. Enquanto o paciente executa os movimentos fisioterapêuticos (esquerda), ações são realizadas no jogo (direita).



Fonte: *Voxar Labs*.

A captura de movimentos também pode ser utilizada no contexto de monitoramento de idosos. Em residências e asilos, o cuidado com pessoas idosas deve ser redobrado; em casos de acidentes, as complicações são em geral mais graves, e devem ser resolvidas com mais urgência. Ao realizar análise de movimento nesses casos, é possível manter esse tipo de situação sob constante vigilância.

No ambiente de trabalho, a análise de poses humanas pode ser utilizada para monitorar o comportamento postural dos empregados. Isso influencia em uma maior qualidade e rendimento do trabalho, bem como na melhoria da qualidade de vida. Similar ao descrito anteriormente, o monitoramento em casos de urgências também é importante quando o trabalho envolve zonas com potenciais riscos de acidentes (máquinas fora de controle, incêndios, dentre outros).

1.1.6 Segurança

A análise de movimento também pode ser utilizada no contexto de segurança pública: na detecção de acidentes envolvendo automóveis e pessoas em vias públicas, bem como na detecção de invasões de propriedade, assaltos e furtos em calçadas públicas.

1.2 Escopo e Objetivos

Considerando a lista de aplicações descrita anteriormente, observa-se que há uma relação entre a qualidade da análise de poses/movimentos humanos e o investimento em recursos tecnológicos para realizar tal tarefa. Enquanto que, na captura de movimentos para animações, o custo de uso de múltiplas câmeras e marcadores reflete uma alta qualidade no rastreamento, técnicas utilizando apenas câmeras RGB como recurso terão um desempenho reduzido, como é o caso da captura de movimentos no contexto de segurança pública.

Apesar de o caso de captura de movimentos para animações ser muito específico, ainda é bastante comum o uso de câmeras RGB e infravermelho em aplicações que façam uso do rastreamento de poses (como é o caso da indústria de jogos, descrita anteriormente), tais como o *Microsoft Kinect* ou o *Open Natural Interaction* (*OpenNI*). Porém, esses tipos de câmeras infravermelho, quando comparadas com câmeras RGB comuns, possuem um custo consideravelmente mais elevado, inviabilizando o seu uso em larga escala e em contextos onde o investimento em recursos é mais escasso. Para se realizar o rastreamento de poses a partir de imagens RGB, primeiramente objetiva-se obter as poses 2D existentes na imagem. Com isso, utiliza-se a informação 2D (e, em alguns casos, informações da imagem original) para obter as poses 3D existentes na imagem.

Nos últimos anos, surgiram trabalhos os quais são capazes de extrair poses 2D humanas, com performance em taxas interativas e com bons resultados quantitativos. Tais trabalhos, em geral, utilizam abordagens em aprendizagem profunda, e serão detalhados adiante.

Apesar de haver trabalhos explorando a obtenção de poses 3D humanas, enxerga-se isso como uma extensão ao projeto, uma vez que o escopo e dificuldade aumentariam consideravelmente.

Considerando tudo isso, o principal objetivo deste trabalho foca em desenvolver uma técnica para a obtenção de poses 2D humanas, a partir de informações provenientes de câmeras RGB. Para tal, foi escolhido o trabalho de CAO et al. (2017) em conjunto com o trabalho de MEHTA et al. (2017) como os alicerces da técnica desenvolvida, uma vez que são considerados

métodos estado da arte (detalhes sobre tais métodos serão explicitados posteriormente). Também enquadra-se como objetivo deste trabalho a realização de melhoria dos resultados, a partir da diminuição da diferença entre as predições de *keypoints* realizadas, quando comparadas com *keypoints* pré-anotados, estes provenientes de bases de imagens externas.

1.3 Estrutura

O trabalho está estruturado da seguinte maneira: no capítulo 2 , serão apresentados alguns conceitos importantes na contextualização das abordagens de rastreamento de poses 2D. Em seguida, o capítulo 3 mostrará os principais trabalhos estado da arte atacando a obtenção de poses. Depois, o capítulo 4 descreverá todo o pipeline do método desenvolvido, detalhadamente. Após isso, o capítulo 5 relatará os principais resultados obtidos do método desenvolvido, bem como trará uma análise dos resultados, vantagens e limitações do método descrito. Por fim, o capítulo 6 apresentará as considerações finais, bem como os potenciais rumos futuros do trabalho.

2

Fundamentação Teórica

Neste capítulo, serão apresentadas algumas áreas de pesquisa e fundamentos a fim de auxiliar no entendimento do método desenvolvido. Dado o estado da arte de extração de poses humanas 2D a partir de imagens RGB, muitos trabalhos focaram no contexto de aprendizagem de máquina, mais especificamente no desenvolvimento de redes neurais convolucionais e transferência de aprendizagem. Serão elucidados conceitos em tais áreas, bem como a fundamentação a respeito das principais ferramentas e recursos que possibilitam o desenvolvimento de tais métodos (inclusive o do trabalho proposto neste documento).

2.1 Aprendizagem de Máquina

A área de aprendizagem de máquina (*Machine Learning*, em inglês) é um ramo de estudo da inteligência artificial na qual criam-se modelos computacionais a partir da premissa que máquinas podem aprender sozinhas, tendo como base a realização de um treinamento a partir de um conjunto de dados, bem como a definição de uma arquitetura de manipulação dos dados a partir de unidades de processamento mais simples. Segundo Tom Mitchell, a aprendizagem de máquina ocorre quando "Um programa de computador é dito para aprender com a experiência E com relação a alguma classe de tarefas T e medida de desempenho P, se o seu desempenho em tarefas em T, medida pelo P, melhora com a experiência E" (MITCHELL et al., 1997).

Tais modelos são especializados em realizar o reconhecimento de padrões, tendo assim um vasto conjunto de aplicações: reconhecimento de sinais, reconhecimento facial, locomoção de robôs, sistemas de recomendação, medicina, dentre outros. Na década de 90, boa parte dessas aplicações não podiam ser resolvidas utilizando técnicas de aprendizagem de máquina, uma vez que recursos computacionais eram mais escassos, bem como informações digitais eram menos acessíveis, impossibilitando o treinamento de modelos mais complexos em larga escala.

Dentre os tipos de aprendizagem de máquina existentes, três se destacam:

1. **Aprendizagem Supervisionada:** Nesse tipo de aprendizagem, deseja-se encontrar uma função tal que descreva o conjunto de dados fornecidos. Nesse caso, tanto o conjunto de entradas quanto as suas respectivas saídas são fornecidas para o treinamento ser realizado. Aqui, há a necessidade da existência de um especialista

no problema, a fim de catalogar tais valores de saída para as entradas dadas. Este modelo de aprendizagem divide-se em dois tipos:

(a) **Classificação:** Em uma classificação, tem-se o objetivo de rotular o valor de entrada de acordo com as distintas categorias pré-definidas no modelo. Neste caso, os valores de saída são discretos (ou categóricos). Um exemplo de problema de classificação seria o desafio de concessão de crédito financeiro por um banco, baseado no perfil de seus clientes; dado um cliente com uma idade específica, histórico de transações e local de residência, decidir se tal cliente merece receber crédito (ou não).

(b) **Regressão:** Em uma regressão, tem-se o objetivo de predizer o valor de saída, em função da entrada fornecida. Neste caso, uma função descreve a relação entre entrada e saída, na qual tais valores de saída podem ser discretos ou contínuos. Um exemplo de problema de regressão seria predizer o preço de residências no mercado imobiliário, em função da sua área, número de quartos e distância ao supermercado mais próximo.

2. **Aprendizagem Não Supervisionada:** No caso da aprendizagem não supervisionada, não há o fornecimento da saída em relação ao conjunto de dados de entrada. Neste caso, não há a existência do especialista para catalogar os dados de entrada. Com isso, é possível realizar o agrupamento (*Clustering*, em inglês) dos dados em função da similaridade entre eles, que se preocupa em minimizar a distância entre dados de um mesmo conjunto. Um sistema de recomendação de filmes é um exemplo de desafio solucionável a partir da aprendizagem não supervisionada; baseado no histórico de filmes assistidos e avaliados positivamente por um cliente, é factível predizer e recomendar filmes com características similares aos previamente assistidos.

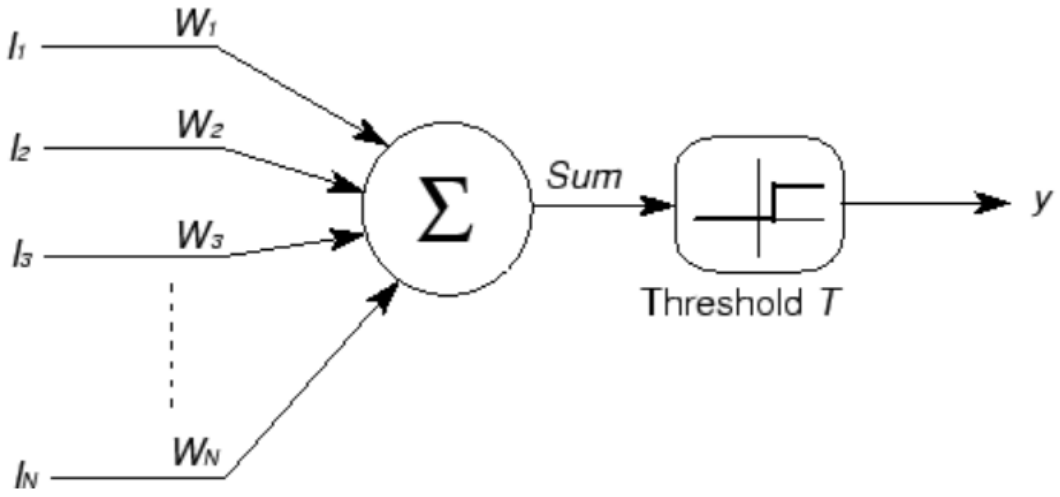
3. **Aprendizagem por Reforço:** Nesse caso, os dados de treinamento são fornecidos durante o processo de uso do modelo; com isso, as predições realizadas vão melhorando conforme penalizações por ações incorretas vão ocorrendo. Um exemplo de utilização desse tipo de abordagem é no contexto de jogos onde, ao jogar uma partida contra um oponente, o modelo é capaz de melhorar suas ações durante a partida.

2.1.1 Unidade de Processamento

No contexto de redes neurais artificiais, é necessário relevar a existência das unidades de processamento nos modelos computacionais. Dentre esses modelos, o primordial foi o neurônio de McCulloch-Pitts (MCCULLOCH; PITTS, 1943). Desenvolvido em 1949 pelo neurocientista Warren McCulloch e pelo lógico Walter Pitts, o neurônio de McCulloch-Pitts (também chamado de MCP) baseou-se num neurônio biológico; é formado por um vetor de entrada, no qual as sinapses são representadas por pesos. Associando cada entrada a seu respectivo peso, é realizada

a soma ponderada dessas entradas, e por fim avalia-se se tal soma está acima ou abaixo de um limiar. Caso sim, a saída do neurônio será 1; caso contrário, será 0. O ajuste dos pesos de entrada do neurônio é realizado à medida que a predição realizada pelo neurônio não corresponde à saída da entrada dada no treinamento. A Figura 2.1 representa uma visualização do modelo deste neurônio.

Figura 2.1: Ilustração do neurônio McCulloch-Pitts. As entradas estão representadas em I_1 a I_N e seus respectivos pesos em W_1 a W_N. É realizada a soma ponderada das entradas e comparada com o limiar T. Por fim, é gerada a saída y.



Com base no neurônio de McCulloch-Pitts, outras unidades de processamento foram surgindo; dentre elas, o Perceptron. Desenvolvido por Frank Rosenblatt em 1958 (ROSENBLATT, 1958), o Perceptron possui estrutura bastante similar ao MCP, porém com uma diferença em relação a como é realizado o treinamento; enquanto que o MCP realiza o treino apenas quando a predição da entrada é incorreta, o Perceptron preocupa-se em ajustar os seus pesos em função do quão errado foi a sua predição. Existem diversas formas de quantizar esse erro; a mais utilizada é o erro médio quadrado (também conhecido como *Mean Squared Error*).

Considerando tais modelos de processamento, é possível arquitetar modelos de redes neurais a partir da conexão entre essas unidades de processamento. Um dos exemplos mais famosos de arquiteturas de redes neurais é o Perceptron multicamadas (também chamado de MLP), no qual há a criação de camadas de neurônios as quais são interligadas entre si. Considerando que, com um Perceptron, é apenas possível computar funções linearmente separáveis, um MLP é capaz de modelar problemas com um grau de complexidade mais elevado, podendo assim computar funções que não são linearmente separáveis.

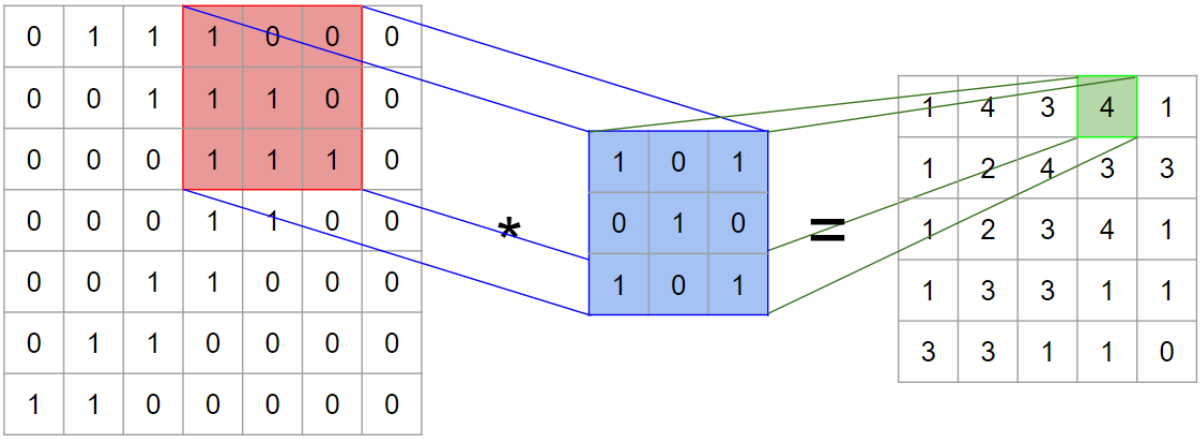
2.2 Redes Neurais Convolucionais

Dentre as principais áreas de pesquisa em aprendizagem de máquina, existe a aprendizagem profunda (*Deep Learning*); esta busca modelar problemas com altos níveis de abstração a partir da inserção de um grande número de camadas de processamento, bem como a realização do treinamento das unidades de processamento presentes nessas camadas. Diversos problemas do mundo real mostraram soluções de alta qualidade quando abordados a partir de conceitos envolvendo aprendizagem profunda, como por exemplo: entendimento do comportamento dos clientes de um determinado serviço, reconhecimento de objetos, reconhecimento de ações, assistentes pessoais, recuperação de informações e até mesmo reconhecimento de voz.

As redes neurais convolucionais (também conhecidas como CNN's ou ConvNets) são uma subclasse de redes neurais profundas, as quais têm mostrado bons resultados no processamento, análise e classificação de imagens. Uma CNN é um tipo de rede MLP, porém com a existência de algumas operações específicas:

1. **Convolução:** Uma operação de convolução é aquela na qual, dada uma matriz de entrada, um filtro (este também chamado de *Kernel*), é perpassado por toda a matriz, com o objetivo de se extrair características da matriz de entrada. O *Kernel* também é uma matriz, e para cada célula *C* da matriz de entrada, esse *Kernel* é aplicado em *C* e nas células ao seu redor, as quais são enquadradas pelo *Kernel*. O valor resultante da aplicação do *Kernel* na janela de células é calculado a partir do produto interno entre as células da janela e do *Kernel*, célula a célula. A Figura 2.2 representa um exemplo de operação de convolução em uma matriz.

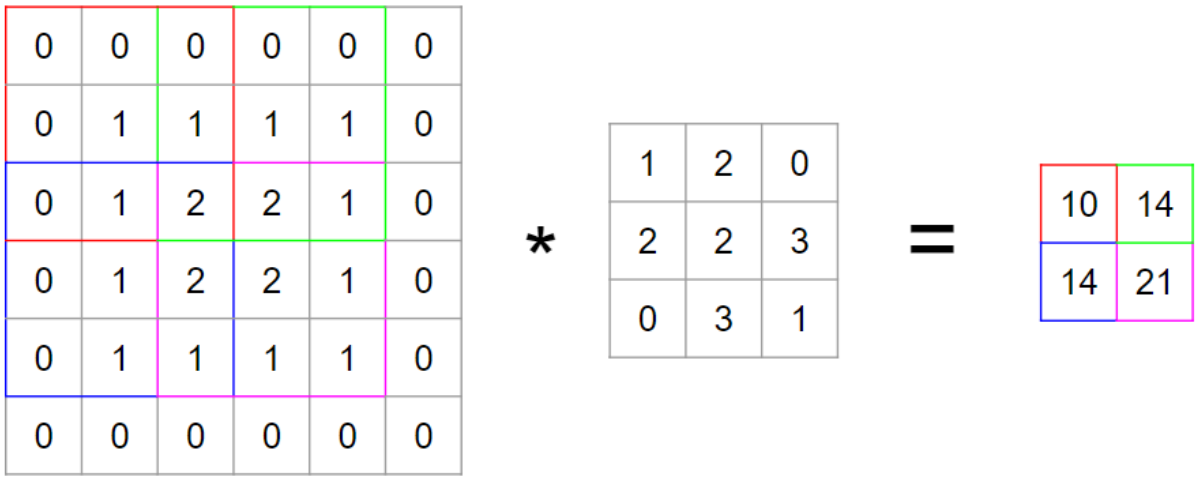
Figura 2.2: Exemplo de convolução em uma matriz. A matriz da esquerda representa a matriz de entrada 7x7, a matriz do meio um Kernel 3x3 e a matriz da direita corresponde ao resultado da convolução. Para cada janela (em vermelho), é aplicado o Kernel (em azul) a partir do produto interno, onde cada janela terá seu resultado em uma célula (em verde) na matriz final.



Também é possível realizar pulos na aplicação do Kernel na matriz de entrada; este pulo é chamado de Stride, o que fará com que algumas janelas de células na matriz de

entrada não tenham seus valores convolucionados. A Figura 2.3 ilustra um exemplo de convolução com a presença de um *Stride*.

Figura 2.3: Exemplo de convolução com valor de *Stride* = 2 em uma matriz. Note que há pulos entre as janelas de células que foram convolucionadas.



2. **Pooling:** Uma operação de Pooling é responsável por reduzir a dimensionalidade de uma matriz a partir da computação de um valor para cada janela de células (Figura 2.4). Um Pooling possui três parâmetros:

   (a) **Tamanho da janela:** Define o tamanho das janelas de células;

   (b) ***Stride*:** Similar à operação de convolução, define os pulos das janelas de células;

   (c) **Função:** Define a função a qual será computada nas janelas de células. Geralmente, são usadas as funções de máximo, média ou soma.

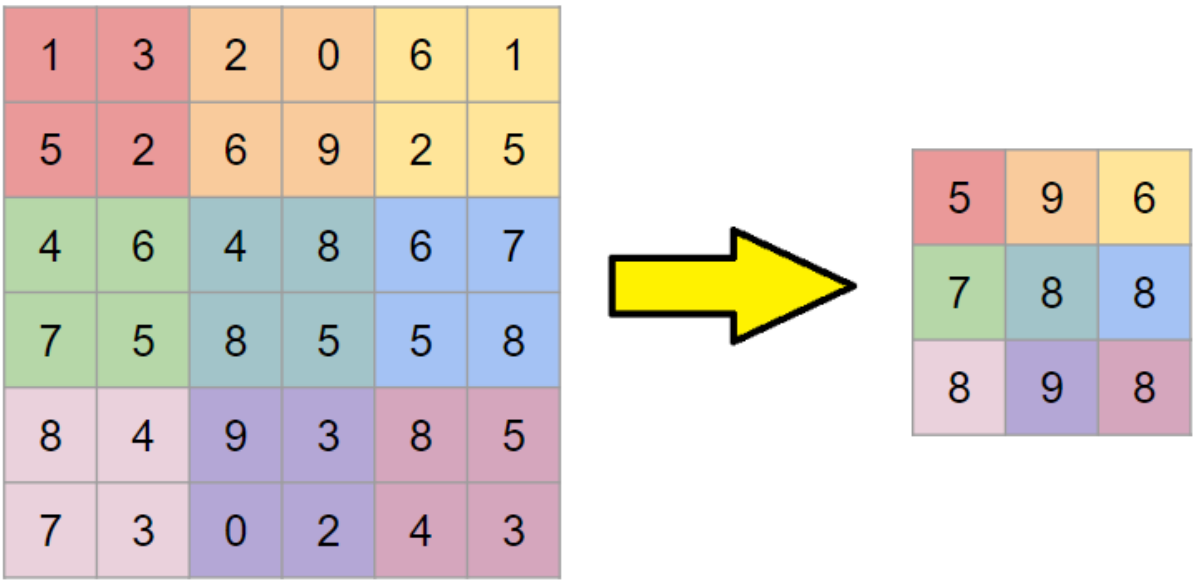
3. ***Rectified Linear Unity* (*ReLU*):** O *ReLU* é uma função de ativação comumente utilizada em redes neurais convolucionais; dada uma matriz, ela é responsável por zerar todas as células com valores negativos da matriz. A Figura 2.5 ilustra um exemplo de operação de ReLU.

2.3 Transferência de Aprendizagem

No contexto de aprendizagem de máquina, existe o conceito de transferência de aprendizagem (*Transfer Learning*); esta é uma técnica na qual um modelo pré-treinado que ataca um determinado problema pode ser reutilizado como ponto de partida para um segundo modelo que ataque um outro problema.

Para problemas em visão computacional e linguagem natural, usar esse tipo de abordagem de aprendizagem de máquina como ponto de partida torna-se interessante, uma vez que, em geral,

Figura 2.4: Exemplo de *Pooling* com tamanho de janela igual a 2*x*2, *Stride* igual a 2, computando a função de máximo valor da janela.



a quantidade de tempo e recursos gastos para realizar o treinamento de redes neurais profundas é elevada. Tendo redes pré-treinadas como partida, o tempo de convergência da rede neural final torna-se mais rápido.

Um dos problemas mais comuns de serem atacados com *Transfer Learning* está relacionado à classificação de imagens e vídeos; para esses tipos de problemas, é interessante utilizar algum modelo de aprendizagem profunda, tal que tenha sido pré-treinado a partir de alguma base de dados para a classificação de imagens. Atualmente, uma das bases de dados mais utilizadas nesse contexto é a *ImageNet* (DENG et al., 2009), a qual possui cerca de 14 milhões de imagens em diferentes contextos, na qual cada uma delas possui informações (também chamado de *Annotations*) a respeito de quais elementos estão presentes (frutas, animais, objetos, etc). Em algumas dessas imagens, provê-se informações acerca da localização dos elementos na imagem, definidos a partir do *bounding-box*.

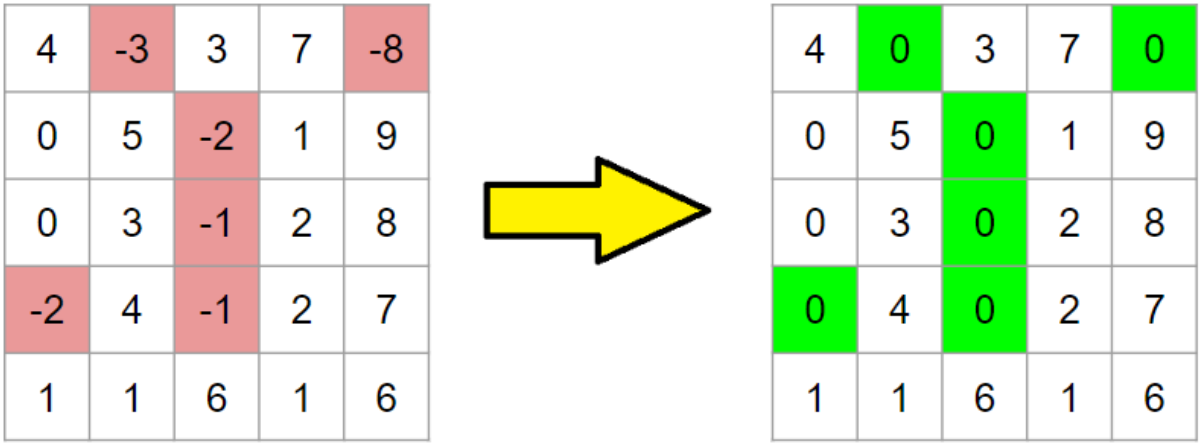
Além disso, a ImageNet dispõe de uma competição de classificação de imagens, na qual grupos de pesquisa de grandes empresas e universidades participam, com o intuito de prover modelos de redes neurais com qualidade o suficiente para serem utilizados por terceiros no contexto de Transfer Learning. Dentre os principais modelos pré-treinados existentes, alguns dos mais utilizados são:

1. ***VGGNet*,** desenvolvido pela Universidade de *Oxford* (SIMONYAN; ZISSERMAN, 2014);

2. ***Inception*,** desenvolvido pela *Google* (SZEGEDY et al., 2015);

3. ***ResNet*,** desenvolvido pela *Microsoft* (HE et al., 2016);

Figura 2.5: Exemplo de *ReLU* em uma matriz. As células da matriz em vermelho tiveram seus valores atualizados para os valores em verde.



4. ***MobileNet*,** deselvolvido pela *Google* (HOWARD et al., 2017).

2.4 *Keras*

No contexto de aprendizagem de máquina, muitas ferramentas tem surgido com o intuito de auxiliar no desenvolvimento e treinamento de modelos de redes neurais. Dentre eles, está o Keras (CHOLLET et al., 2015), uma ferramenta de código aberto de alto nível, escrita em *Python*, que pode utilizar outras ferramentas de aprendizagem de máquina como *backend*, tais como *TensorFlow* (ABADI et al., 2016), *Theano* (BERGSTRA et al., 2010) ou *MXNet* (CHEN et al., 2015). O *Keras* possui uma interface de fácil uso, na qual a modelagem de uma rede neural pode ser realizada com poucas linhas de código. Com uma estrutura de blocos, o *Keras* permite encadear modelos de redes neurais, facilitando o uso de *Transfer Learning* na geração de novos modelos.

Por ser de código aberto, tal ferramenta está sob constantes melhorias, tais como na criação de funções de ativação, tipos de camadas, otimizadores e funções de erro. Além disso, o Keras possui suporte para a exportação da utilização do modelo treinado em dispositivos móveis, tais como iOS e Android.

3

Trabalhos Relacionados

Nessa capítulo, serão apresentados os principais trabalhos focados na obtenção de poses 2D humanas. Muitos dos primeiros trabalhos nessa linha atacam esse desafio utilizando conceitos de visão computacional. Já em trabalhos mais recentes, abordagens utilizando aprendizagem de máquina mostraram resultados mais promissores.

Em REN; BERG; MALIK (2005), extração de poses 2D é vista como um desafio de detecção de componentes retilíneos na imagem (Figura 3.1). Primeiramente, obtém-se um mapa de contornos. Após isso, é aplicado em tal mapa o algoritmo de triangulação de Delaunay, com o objetivo de detectar a existência de paralelismo entre os contornos. Dados o conjunto de paralelismo entre contornos, utiliza-se programação quadrática para definir quais pares de contornos representam partes do corpo humano. Com isso, extrai-se as juntas do corpo humano existentes na imagem. Este trabalho, apesar de apresentar um custo computacional eficiente, não apresenta cobertura para casos onde haja a presença de múltiplas pessoas na imagem.

Figura 3.1: Pipeline do método descrito em (REN; BERG; MALIK, 2005).

Já a obtenção de poses 2D em MORI; MALIK (2006) usa exemplos de imagem com as juntas já rotuladas. Dada uma imagem de entrada, obtém-se um mapa de contornos, e extrai-se uma amostra dos pontos desse mapa. Com isso, são calculadas as correspondências entre o

exemplo de imagem e a imagem de entrada. Enquanto isso, é calculada a deformação entre as imagens e aplicada tal deformação. Após calculado um grau de similaridade alto entre as imagens, a imagem de entrada é então associada a um modelo cinemático de corpo humano, assim obtendo a pose 2D do corpo presente na imagem. Apesar de que a abordagem analítica garante uma corretude no contexto matemático, é necessário informações a respeito da altura da pessoa a qual está sendo rastreada.

PAPANDREOU et al. (2017), por sua vez, utilizou uma abordagem utilizando aprendizagem de máquina; para uma dada imagem, é aplicado um detector de pessoas Faster-RCNN, que prediz o *bounding-box* de cada pessoa detectada. Assim, para cada pessoa detectada, é utilizado uma CNN com o intuito de predizer mapas de calor com as possíveis localizações das juntas do corpo humano detectado. Neste trabalho, foi utilizada a ResNet como rede convolucional para realizar o Transfer Learning da predição das juntas. Apesar de possuir um dos melhores resultados no conjunto de dados do COCO-Dataset 2016, seu método peca na performance, uma vez que é necessário executar a predição da rede neural para cada um dos *bounding-boxes* obtidos a partir do Faster-RCNN.

Similar ao trabalho descrito anteriormente, CAO et al. (2017) também utilizou CNN para realizar a detecção de cada uma das juntas do corpo humano através da predição de mapas de calor (Figura 3.2). Além disso, também é realizada a predição dos mapas de calor dos ossos, os quais são denominados *Part Affinity Fields*. Com essas duas informações, torna-se possível associar conjuntos de juntas a um mesmo indivíduo. Neste trabalho, a CNN modelada utiliza-se da *VGGNet* para realizar o *Transfer Learning*, acelerando assim o treinamento da rede. O resultado desse trabalho traz bons resultados para imagens sob diversas condições, incluindo a presença de múltiplas pessoas e regiões de baixa iluminação. Além disso, este trabalho apresenta bons resultados no contexto de performance, podendo ser utilizado em aplicações de tempo real.

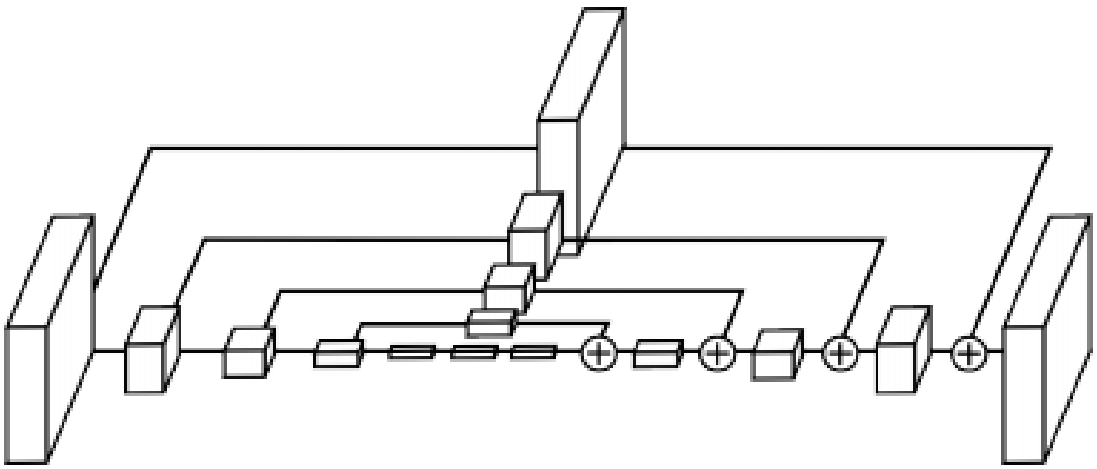
O trabalho de MEHTA et al. (2017) realizou uma extensão, com o objetivo de obter poses 3D humanas, em função das poses 2D obtidas, seguindo o trabalho de CAO et al. (2017). Diferentemente deste, utilizou-se da ResNet (HE et al., 2016) para realizar o *Transfer Learning* das características da imagem. Além disso, este trabalho realiza um *Transfer Leaning* dos resultados das poses 2D obtidas, somadas às características da imagem, a fim de obter poses tridimensionais. Similar a CAO et al. (2017), este trabalho também apresenta bons resultados na extração de poses para imagens sob diversos contextos.

Para realizar a predição das juntas, NEWELL; YANG; DENG (2016) utiliza uma técnica similar a CAO et al. (2017); utilizando CNN, mapas de calor são gerados para cada uma das juntas, porém a pose corresponde ao corpo mais central da imagem (logo, tal método não funciona para múltiplas pessoas na imagem). Uma vez que a realização de convoluções na entrada aumenta o número de filtros e diminui a resolução da imagem, o diferencial desse método diz respeito a reaumentar tal resolução a partir de operações de *pooling* e *Upsampling* (este último funcionando basicamente como um redimensionador de matrizes). Além disso, camadas da rede posteriores são realimentadas por camadas anteriores, auxiliando no treinamento. Este

Figura 3.2: Exemplo de predições realizadas por (CAO et al., 2017). As imagens de cima representam a predição da localização das juntas, enquanto que as imagens de baixo representam a predição das ligações entre juntas.

trabalho é um dos precussores das técnicas de CNN aplicada à extração de poses, esta sendo expandida para diversas outras técnicas. A Figura 3.3 ilustra a arquitetura utilizada para a predição dos *keypoints*.

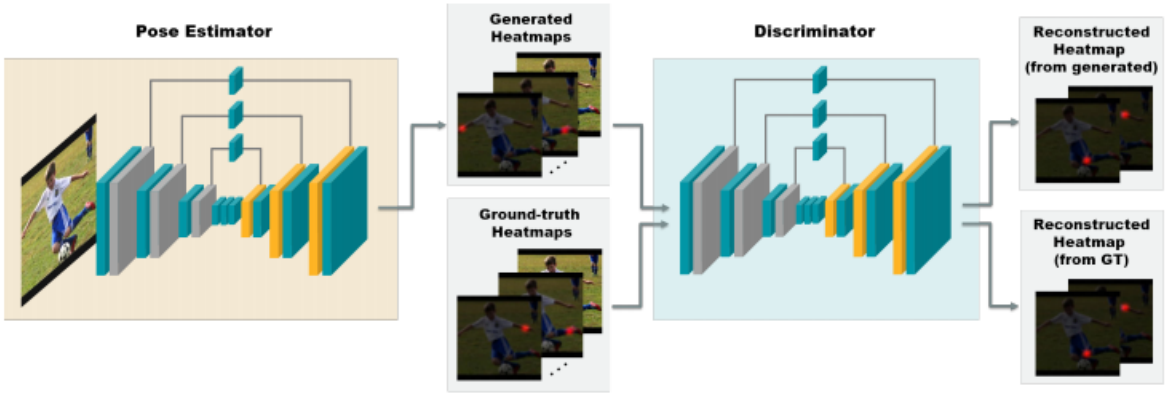
Figura 3.3: Pipeline do modelo descrito por NEWELL; YANG; DENG (2016). Até a metade, o modelo foca em realizar operações de convoluções. Após isso, realiza-se uma sequência de *pooling* e *upsampling*.



O trabalho de CHOU; CHIEN; CHEN (2017) focou no uso de redes adversariais. Neste, são utilizados dois modelos: um gerador, encarregado de gerar novos casos de teste, em função da base de dados recebida, e um discriminador, o qual é responsável por predizer a pose humana na imagem, levando tanto em consideração a corretude dos casos da base de dados original

quanto a incorretude dos casos gerados pelo gerador. Em tal trabalho, tanto o gerador quanto o discriminador utilizam o modelo descrito por NEWELL; YANG; DENG (2016). Similar aos trabalhos envolvendo CNN descritos anteriormente, seu poder de generalização é satisfatório quando trata-se de imagens sob diferentes circunstâncias. A Figura 3.4 ilustra a arquitetura do modelo, bem como a relação entre o modelo gerador e discriminador.

Figura 3.4: Modelo descrito por CHOU; CHIEN; CHEN (2017). Nele, o gerador (*pipeline* à esquerda) é responsável por gerar casos de teste), enquanto o discriminador (*pipeline* à direita) é responsável por distinguir se uma entrada refere-se a um caso de teste original ou sintetizado pelo gerador.



4

Desenvolvimento

Neste capítulo, são apresentados os passos necessários para o desenvolvimento do método de rastreamento de poses 2D humanas a partir de imagens RGB, objetivo desse trabalho. Primeiramente, será discutido o pipeline adotado para a estimação de poses 2D. Após isso, serão esclarecidos detalhes a respeito da arquitetura da rede neural convolucional adotada, bem como sobre a realização do seu treinamento. Com isso, serão explicadas informações a respeito do tratamento dos dados obtidos da rede neural modelada.

4.1 Pipeline

O pipeline do método toma como base o trabalho proposto por CAO et al. (2017). Nele, é adotada a técnica de *Transfer Learning*, na qual uma rede pré-treinada é inicializada, a fim de obter-se características da imagem e utilizá-las como entrada para uma rede neural convolucional, esta capaz de realizar a detecção das juntas dos corpos humanos existentes na imagem (Figura 3.2).

A rede pré-treinada utilizada é um subconjunto contendo as dez primeiras camadas da *VGG-19* (SIMONYAN; ZISSERMAN, 2014), esta treinada na base de imagens da *ImageNet* (KRIZHEVSKY; SUTSKEVER; HINTON, 2012), que é comumente utilizada no contexto de classificação de imagens a partir de técnicas de aprendizagem de máquina. Com isso em mãos, foi adicionada uma camada de convolução capaz de predizer um mapa de calor para cada uma das juntas do corpo humano, bem como dois mapas de calor (denominados *Part Affinity Fields*) para cada uma das ligações existentes entre as juntas, as quais auxiliam na ligação entre as juntas localizadas na formação de esqueletos humanos.

4.2 Arquitetura

A arquitetura da rede neural convolucional está estruturada da seguinte maneira: primeiramente, tem-se como entrada uma imagem RGB, a qual é redimensionada para $224x224x3$. Tal dimensão é utilizada, pois é alta o suficiente para trazer uma boa qualidade na predição, e é baixa o suficiente para não demandar processamento excessivo. Baseado no trabalho de MEHTA et al. (2017), foi realizado o *Transfer Learning* da rede entitulada *ResNet-50*, esta sendo pré-treinada

também a partir da base de imagens da *ImageNet*, e é extraido o resultado da camada $res4f$. A partir desta arquitetura, encadeia-se mais algumas camadas de convolução, escala e *ReLU*, em uma estrutura similar à *ResNet-50*, com o objetivo de aumentar a dimensionalidade da saída e possibilitar a regressão linear multidimensional. a Figura 4.1 ilustra o *pipeline* desenvolvido para a extração dos mapas de calor.

Por fim, foi adicionada uma camada de convolução, na qual o número de filtros estará estritamente relacionado à quantidade de juntas e ligações entre juntas do corpo humano. A Figura 4.2 ilustra o conjunto de juntas e ligações entre juntas cujas localizações deseja-se predizer. O conjunto de juntas do corpo humano cujas localizações na imagem deseja predizer são:

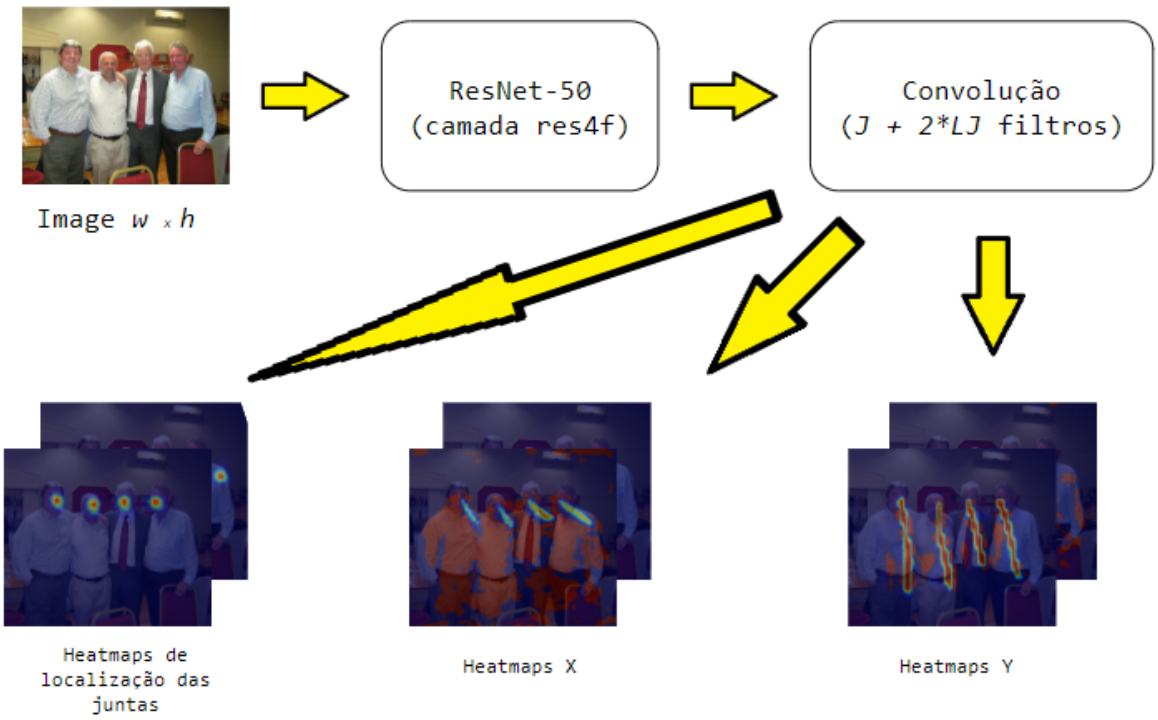
1. Nariz;
2. Ombro esquerdo;
3. Ombro direito;
4. Cotovelo esquerdo;
5. Cotovelo direito;
6. Pulso esquerdo;
7. Pulso direito;
8. Pélvis esquerda;
9. Pélvis direita;
10. Joelho esquerdo;
11. Joelho direito;
12. Tornozelo esquerdo;
13. Tornozelo direito.

Já a lista de ligações entre juntas cujos valores deseja-se predizer são:

1. Nariz e ombro esquerdo;
2. Nariz e ombro direito;
3. Ombro esquerdo e cotovelo esquerdo;
4. Ombro direito e cotovelo direito;
5. Cotovelo esquerdo e pulso esquerdo;

6. Cotovelo direito e pulso direito;
7. Nariz e pélvis esquerda;
8. Nariz e pélvis direita;
9. Pélvis esquerda e joelho esquerdo;
10. Pélvis direita e joelho direito;
11. Joelho esquerdo e tornozelo esquerdo;
12. Joelho direito e tornozelo direito;

Figura 4.1: Pipeline adotado. O número de filtros da convolução de saída foi definido em função da quantidade de juntas (*J*) e ligações entre juntas (*LJ* para os heatmaps dos valores X dos vetores unitários das ligações e *LJ* para os heatmaps dos valores Y dos vetores unitários).



Em relação à última camada de convolução referenciada anteriormente, o objetivo é que, para cada uma das juntas, a saída da rede neural gere um mapa de calor o qual seja capaz de predizer, em cada pixel, o grau de confiabilidade de haver a respectiva junta naquela posição da imagem. Os valores provenientes dos mapas de calor variam entre 0 (extremamente improvável de haver uma junta naquela posição) e 1 (extremamente provável de haver uma junta naquela posição). Para a arquitetura implementada, cada um dos mapas de calor de saída possui 25% das dimensões da entrada da rede neural (o que equivale a possuir dimensões de 56x56). A

Figura 4.2: Representação visual das informações as quais desejam-se predizer. Os círculos vermelhos representam as juntas, e as linhas verdes representam as ligações entre as juntas.

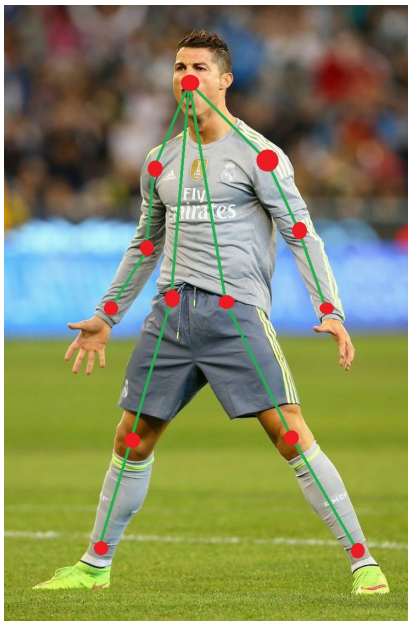


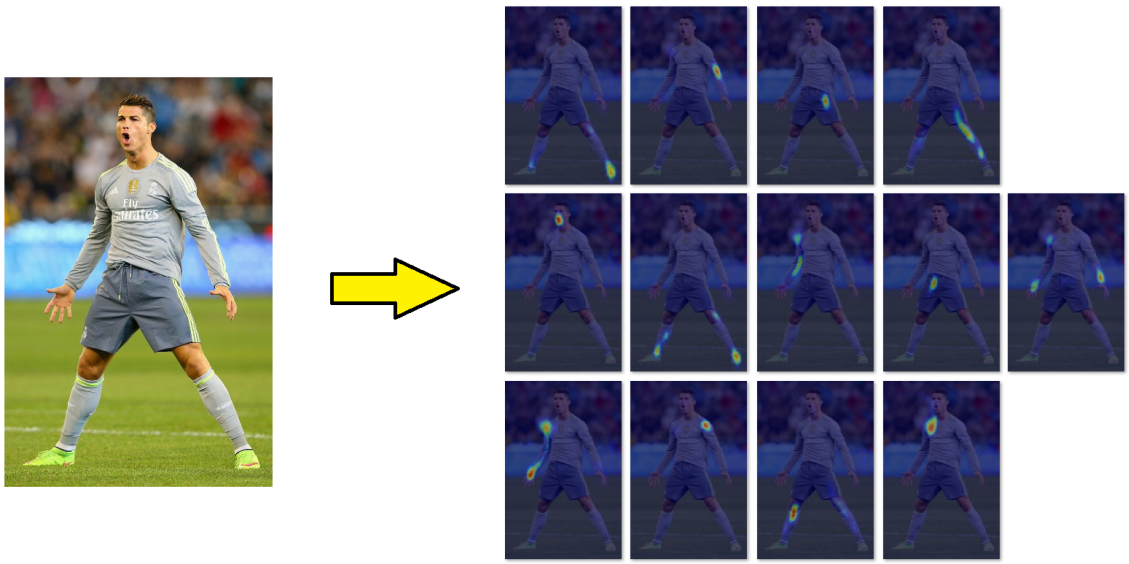
Figura 4.3 representa uma predição dos mapas de calor para cada uma das juntas, dada uma imagem.

Porém, a obtenção dos mapas de calor das juntas não é o suficiente para extração das poses humanas finais. Uma vez que os mapas de calor são independentes e que a presença de mais de um corpo humano na imagem é possível, é necessário calcular os mapas de calor relativos às ligações das juntas do corpo humano, conforme listadas anteriormente. Para cada uma das ligações entre as juntas, são gerados dois mapas de calor:

1. **Mapa de calor da dimensão *x*:** Para cada um dos pixels do mapa de calor, a rede prediz o direcionamento *x* do vetor unitário da ligação das juntas, caso seja provável a existência dessa ligação na região. Diferentemente dos mapas de calor das juntas, os valores variam entre -1 (direcionamento da ligação das juntas para a esquerda) e 1 (direcionamento da ligação das juntas para a direita). Valores no mapa de calor próximos de 0 representam a inexistência da ligação das juntas ou a existência de ligação de juntas verticais (estas melhor explicitadas no mapa de calor da dimensão *y*, estas descritas em seguida). A Figura 4.4 representa uma predição dos mapas de calor da dimensão *x*, para cada uma das ligações de juntas, dada uma imagem;

2. **Mapa de calor da dimensão *y*:** Similar ao mapa de calor da dimensão *x*, nesse caso a rede prediz o direcionamento *y* do vetor unitário da ligação das juntas, caso seja provável a existência dessa ligação na região. Diferentemente do mapa de calor da dimensão *x*, este será capaz de verificar a existência de ligação de juntas na direção vertical, predizendo resultados entre -1 (direcionamento da ligação das juntas para cima) e 1 (direcionamento da ligação das juntas para baixo) para cada *pixel*. Análogo

Figura 4.3: Mapas de calor gerados para cada uma das juntas do corpo humano. As regiões em vermelho representam uma alta confiabilidade de haver a junta no local.



ao mapa de calor da dimensão x, valores no mapa de calor próximos de 0 representam a inexistência de ligação de juntas ou a existência delas, porém evidenciadas na posição horizontal (sendo assim melhor visualizadas no mapa de calor da dimensão x). A Figura 4.5 representa uma predição dos mapas de calor da dimensão y, para cada uma das ligações de juntas, dada uma imagem.

Tendo os mapas de calor das ligações das juntas para as dimensões x e y, é possível realizar o cálculo de correlação entre juntas de tipos diferentes, a fim de se descobrir se tais juntas pertencem ou não a um mesmo corpo humano. Mais detalhes desse cálculo serão explicitados na seção 4.3.

4.2.1 Treinamento

Para realizar o treinamento da rede, foi necessária a obtenção de uma base de dados contendo um extenso conjunto de imagens, bem como as anotações das posições das juntas dos corpos humanos nas imagens. Para tal, foi utilizado o *COCO-Dataset* (LIN et al., 2014), que é uma base de imagens de larga escala, bastante utilizada na realização de detecção e segmentação de objetos. Tal base de imagens também é amplamente utilizada no contexto de obtenção de poses humanas (CAO et al., 2017; PAPANDREOU et al., 2017; MEHTA et al., 2017). Nesse trabalho, foi utilizado o *COCO-Dataset 2017*, o qual provê uma base de 118.287 imagens para a realização do treinamento, além de 5.000 imagens para a validação dos resultados de treino. Em ambos conjuntos de imagens, a presença de corpos humanos é arbitrária; no caso em que há a presença de humanos na imagem, também é fornecido um conjunto de anotações, o qual provê a localização (em *pixels*) de cada uma das juntas para cada humano, caso a junta esteja presente na

Figura 4.4: Mapas de calor da dimensão x gerados para cada uma das juntas do corpo humano. As regiões em vermelho representam uma alta confiabilidade de haver a junta no local.

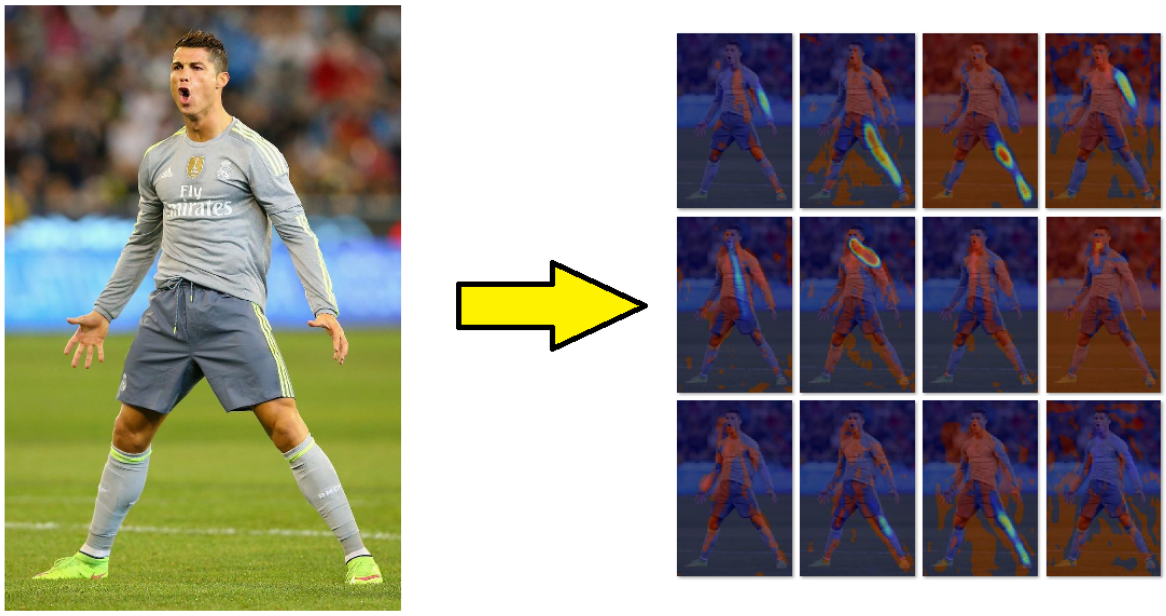


imagem. Com isso, é possível gerar os casos de treino para a rede neural descrita anteriormente, uma vez que também temos a informação de correspondências de juntas de tipos diferentes a um mesmo humano.

Uma vez que, para cada junta, é fornecido apenas o *pixel* central da sua localização, é preferível que, para os casos de imagens de treinamento, o seu respectivo mapa de calor considere os arredores do pixel como possível candidato para a localização real da junta. Logo, os casos de treinamento computam uma função gaussiana, com o objetivo de aumentar o poder de generalização da predição da localização da junta. a Figura 4.6 representa um exemplo de imagem (usada como entrada para o treinamento), bem como o seu mapa de calor de junta (no caso a junta do tipo nariz), esta usada como saída do treinamento.

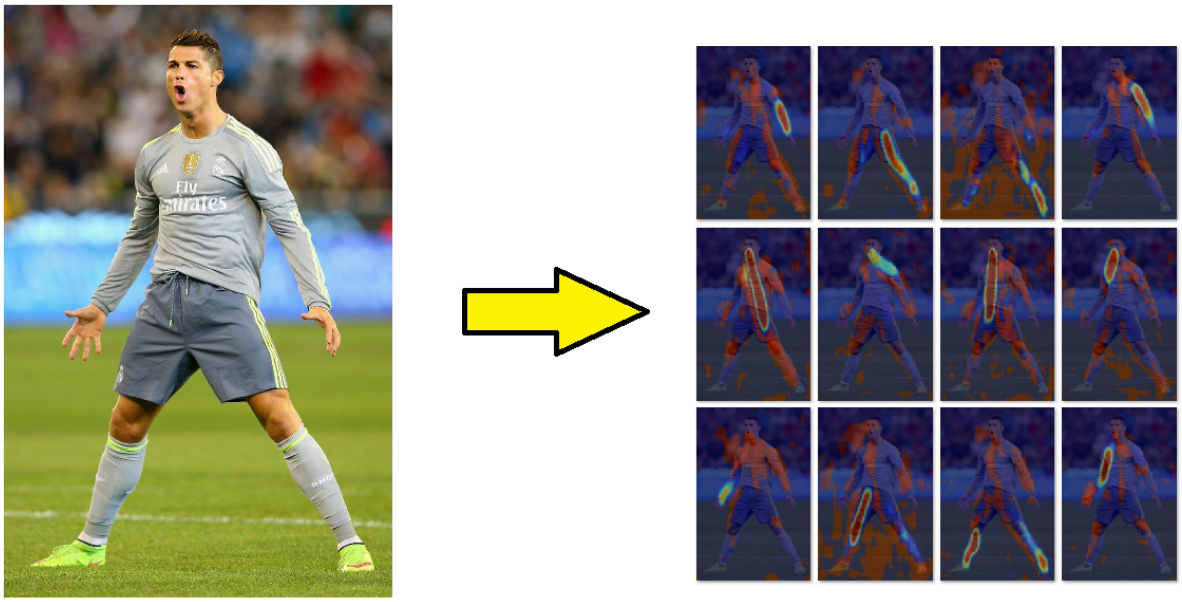
Baseado em CAO et al. (2017), a construção de um mapa de calor para uma junta a fim de realizar-se o treinamento funciona da seguinte maneira:

1. Considere K juntas do tipo J na imagem. Para cada uma das anotações da junta X_j, calcula-se o valor da função gaussiana S, para cada pixel P do mapa de calor para a anotação $X_{j,k}$ e considerando um valor para a variância σ (a qual define a probabilidade de distribuição do pixel P), onde:

$$S(P, X_{j,k}) = exp(-\frac{\|P - X_{j,k}\|^2}{\sigma^2}) \tag{4.1}$$

2. Após obtidos todos os valores da função gaussiana para cada pixel P para cada anotação $X_{j,k}$, extrai-se o valor máximo para cada pixel, com o objetivo de "somar"

Figura 4.5: Mapas de calor gerados da dimensão *y* para cada uma das juntas do corpo humano. As regiões em vermelho representam uma alta confiabilidade de haver a junta no local.



todas as juntas em um único mapa de calor, tal que:

$$S(P,X_j) = \max_k(S(P,X_{j,k})) \tag{4.2}$$

Figura 4.6: Exemplo de um mapa de calor de narizes de uma imagem, sintetizado a partir das anotações de suas localizações. Quanto mais próximo das regiões em vermelho, maior o grau de confiança de existência de um nariz naquela localização.

Já em relação à construção dos mapas de calor das dimensões *x* e *y* das ligações entre as juntas, o processo também é baseado no trabalho de CAO et al. (2017), e funciona da seguinte maneira:

1. Considere K ligações entre juntas J_1 e J_2 na imagem. Para cada uma das anotações da ligação das juntas X_{j_1} e X_{j_2}, calcula-se o vetor unitário o qual liga as duas juntas,

no qual:

$$V(X_{j_1,k}, X_{j_2,k}) = (X_{j_2,k} - X_{j_1,k}) / \|X_{j_2,k} - X_{j_1,k}\|^2 \tag{4.3}$$

2. Calculados todos os K vetores unitários $V(X_{j_1,k}, X_{j_2,k})$, tem-se agora o objetivo de preencher os mapas de calor das ligações das juntas L_x e L_y com o valor do vetor unitário. Cada um dos valores x e y do vetor unitário são preenchidos nos mapas de calor das ligações das juntas das dimensões x e y respectivamente. O preenchimento para cada *pixel* P é feito na forma tal que:

$$L(P, V(X_{j_1,k}, X_{j_2,k})) = \begin{cases} V(X_{j_1,k}, X_{j_2,k}), & \text{se } P \text{ estiver na reta que conecta } X_{j_1,k} \text{ a } X_{j_2,k} \\ 0, & \text{caso contrário} \end{cases} \tag{4.4}$$

3. Uma vez que pode ocorrer o cruzamento de N ligações entre o mesmo tipo de pares de juntas C em cada pixel P na imagem, isso é resolvido através do cálculo da média dos $N(P)$ valores de anotações de cada *pixel* P, tal que:

$$L_c(P,k) = \frac{1}{N(P)} \sum_{k=1}^{K} L(P, V(X_{j_1,k}, X_{j_2,k})) \tag{4.5}$$

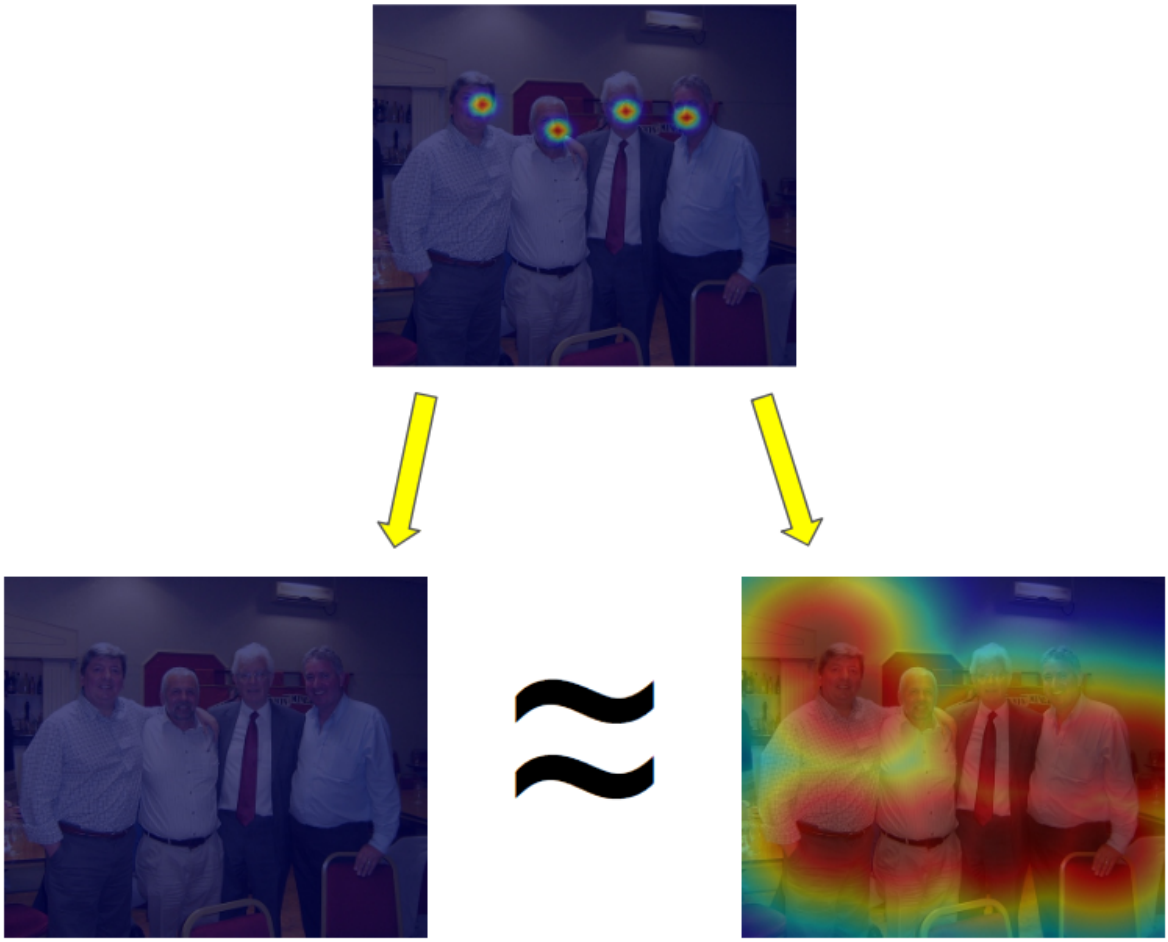
4.2.2 Função de Erro

Considerando o que foi detalhado na seção "Arquitetura" anteriormente, há o desafio de modelar a função de erro, a fim de se realizar o treinamento da rede neural. Em CAO et al. (2017), é utilizada uma variação do erro médio quadrado (*Mean Square Error*), na qual o cálculo do erro considera a distância quadrada entre a predição e o *ground truth*, porém o conjunto de *pixels* P é tal que engloba apenas os *pixels* que possuem anotação de junta (e também desconsiderando a computação da função gaussiana do treinamento). Para a predição dos mapas de calor das juntas, essa abordagem apresenta-se crítica, uma vez que predizer mapas de calor com valores de confiança máximos traria resultados incorretos, uma vez que desconsidera o erro causado pela existência de falsos positivos, os quais não seriam expostos por tal função de erro.

Pelo outro lado, ao utilizar *Mean Square Error* como função de erro da rede enviesaria o resultado das predições dos mapas de calor para o outro extremo; predições de mapas de calor com valores de confiança mínimos trariam resultados incorretos em muitos casos, uma vez que minimizaria o erro causado pela existência de falsos negativos, os quais não seriam evidenciados por esta função de erro.

Dadas as duas situações, propõe-se uma função de erro, tal que considere quantitativamente similar o erro existente em mapas de calor com valores de confiança maximizados e minimizados, a fim de evitar a convergência do treinamento da rede neural para algum desses dois pólos. O mesmo processo é proposto para o cálculo de erro dos mapas de calor das ligações entre juntas. A Figura 4.7 ilustra a ideia a qual moldou esta função de erro.

Figura 4.7: Ilustração da consequência da escolha da nova função de erro. O objetivo é que, para uma dada imagem de entrada, mapas de calor com predições minimizadas (imagem inferior à esquerda) tenham um erro similar a mapas de calor com predições maximizadas (imagem inferior à direita).



Para modelar tal função de erro, levou-se em consideração dois casos: a predição incorreta de confiança, em *pixels*, os quais possuem anotações positivas no treinamento (o que configura-se como falsos negativos), bem como a predição incorreta em locais os quais possuem anotações negativas no treinamento (o que configura-se como falsos positivos). Uma vez que, no contexto de extração de juntas em imagens, a quantidade de anotações positivas é geralmente muito mais reduzida em relação à quantidade de anotações negativas, procura-se proporcionar a função de erro tal que ela seja mais penalizadora em casos de falsos negativos quando comparados com casos de falsos positivos.

Considera-se, para a predição das juntas, J mapas de calor de predições X_j de dimensões H_w por H_h, bem como os seus respectivos J mapas de calor *ground truths* Y_j, também com dimensões H_w por H_h. Com isso, a função de erro E_j para a predição das juntas é definida por:

$$E_j = \sum_{j}^{J} \sum_{w}^{H_w} \sum_{h}^{H_h} \left(\frac{H_w + H_h}{2}\right)^{Y_{j,w,h}} * (X_{j,w,h} - Y_{j,w,h})^2 \tag{4.6}$$

Já para a predição das ligações das juntas para as dimensões x e y, considera-se LJ mapas de calor de predições $X_{lj,x}$ e $X_{lj,y}$ de dimensões H_w por H_h, bem como os seus respectivos LJ mapas de calor *ground truths* $Y_{lj,x}$ e $Y_{lj,y}$, também com dimensões H_w por H_h. Com isso, as funções de erro $E_{lj,x}$ e $E_{lj,y}$ para a predição das ligações das juntas são definidas por:

$$E_{lj,x} = \sum_{lj}^{LJ} \sum_{w}^{H_w} \sum_{h}^{H_h} (\frac{H_w + H_h}{2})^{Y_{lj,x,w,h}} * (X_{lj,x,w,h} - Y_{lj,x,w,h})^2 \tag{4.7}$$

$$E_{lj,y} = \sum_{lj}^{LJ} \sum_{w}^{H_w} \sum_{h}^{H_h} (\frac{H_w + H_h}{2})^{Y_{lj,y,w,h}} * (X_{lj,y,w,h} - Y_{lj,y,w,h})^2 \tag{4.8}$$

Dadas as três funções de erro parciais, a função de erro final é definida por:

$$E = E_j + E_{lj,x} + E_{lj,y} \tag{4.9}$$

4.3 Pós-Processamento

Após obtidos os mapas de calor das juntas e das ligações entre as juntas, é necessário realizar uma série de processamentos, tal que resulte na obtenção do conjunto de poses humanas existentes na imagem. Isto equivale a extrair uma matriz de dimensões NxJ, na qual N equivale ao número de indivíduos na imagem e J equivale ao número de juntas existentes. Vale ressaltar que, em praticamente todos os casos de imagens, haverão juntas de indivíduos cujas detecções não foram realizadas. A extração do conjunto de poses é dividida em três partes:

1. Extração das juntas;
2. Obtenção das ligações entre juntas;
3. Agrupamento das ligações entre juntas.

4.3.1 Extração das Juntas

Nesta etapa, o objetivo é extrair um conjunto de pares de juntas a partir dos J mapas de calor das juntas provenientes da rede neural, onde J equivale ao número de juntas. Podemos observar esse problema como um desafio de detecção de máximos locais em uma matriz bidimensional. Para evitar a detecção de máximos locais com valores de confiança baixos, primeiramente zera-se valores abaixo de um limiar t. Com isso, são obtidos os *pixels* os quais são máximos locais em relação a uma janela de pixels de dimensões $3x3$.

4.3.2 Obtenção das Ligações Entre Juntas

Obtido o conjunto de juntas, é necessário descobrir a relação entre as juntas que se conectam. Para exemplificar tal problema, considere que a extração das juntas detecte 8 cotovelos

esquerdos e 6 punhos esquerdos. Apenas com a informação dos pixels das juntas, uma abordagem ingênua seria conectar antebraços esquerdos a partir da distância euclidiana mínima entre os pixels. Visto que corpos humanos em imagens possuem características de profundidade, esta abordagem erraria em muitos casos.

É nesta etapa que a utilização dos mapas de calor das ligações entre as juntas ocorre. Visto que os mapas de calor das dimensões x e y computam a direção do vetor unitário de ligação das juntas, pode-se calcular a correlação entre pares de juntas a partir da leitura dos valores no mapa de calor os quais são interpolações entre esses pares.

Para ilustrar tal ideia, considere a ilustração da Figura 4.8. Dados dois narizes N_1 e N_2 e dois ombros esquerdos O_1 e O_2, o objetivo é ligar N_1 e N_2 ao ombro esquerdo cujo conjunto de pontos (interpolações entre nariz e ombro) esteja bem correlacionado com os valores computados nos mapas de calor x e y. Na imagem, enquanto O_1 mostra-se mais bem correlacionado com N_1, O_2 poderá estar correlacionado com outro nariz na imagem.

Dada uma ligação entre as juntas de tipos a e b, o cálculo de correlação é realizado para todas as combinações entre as i juntas do tipo a e as j juntas do tipo b. Baseado no trabalho de CAO et al. (2017), o cálculo de correlação entre a_i e b_j é tal que:

$$C_{a_i,b_j} = \sum_{u=0}^{10} H_{P_{u,a_i,b_j}} \cdot \left(\frac{b_j - a_i}{\|b_j - a_i\|^2}\right)$$
$$\text{onde } P_{u,a_i,b_j} = \left(1 - \frac{u}{10}\right) * a_i + \frac{u}{10} * b_j$$

Feito o cálculo de todas as combinações de juntas, obtém-se uma matriz M com i linhas e j colunas, com cada valor de correlação $M_{i,j}$ computado.

A partir daqui, tem-se o objetivo de selecionar o conjunto de correlações cuja soma seja a máxima. Para resolver tal desafio, utiliza-se o algoritmo húngaro, o qual traz uma solução ótima a este problema (KUHN, 1955). Aplicando esta etapa a todas as LJ ligações entre juntas, o resultado obtido resume-se a um conjunto de arestas, tal que cada aresta representa uma ligação entre uma junta a e outra junta b.

4.3.3 Agrupamento das Ligações Entre Juntas

Após obtidas todas as ligações entre juntas, tem-se um conjunto de arestas. A partir daqui, o objetivo resume-se a descobrir a quais corpos humanos cada aresta pertence. Encarando o conjunto de arestas como um grafo, este problema resume-se a encontrar os componentes conexos existentes, onde cada componente conexo representa um corpo humano. Para tal, utiliza-se o algoritmo de busca em profundidade. Feito isto, por fim obtém-se o conjunto de seres humanos e suas respectivas juntas.

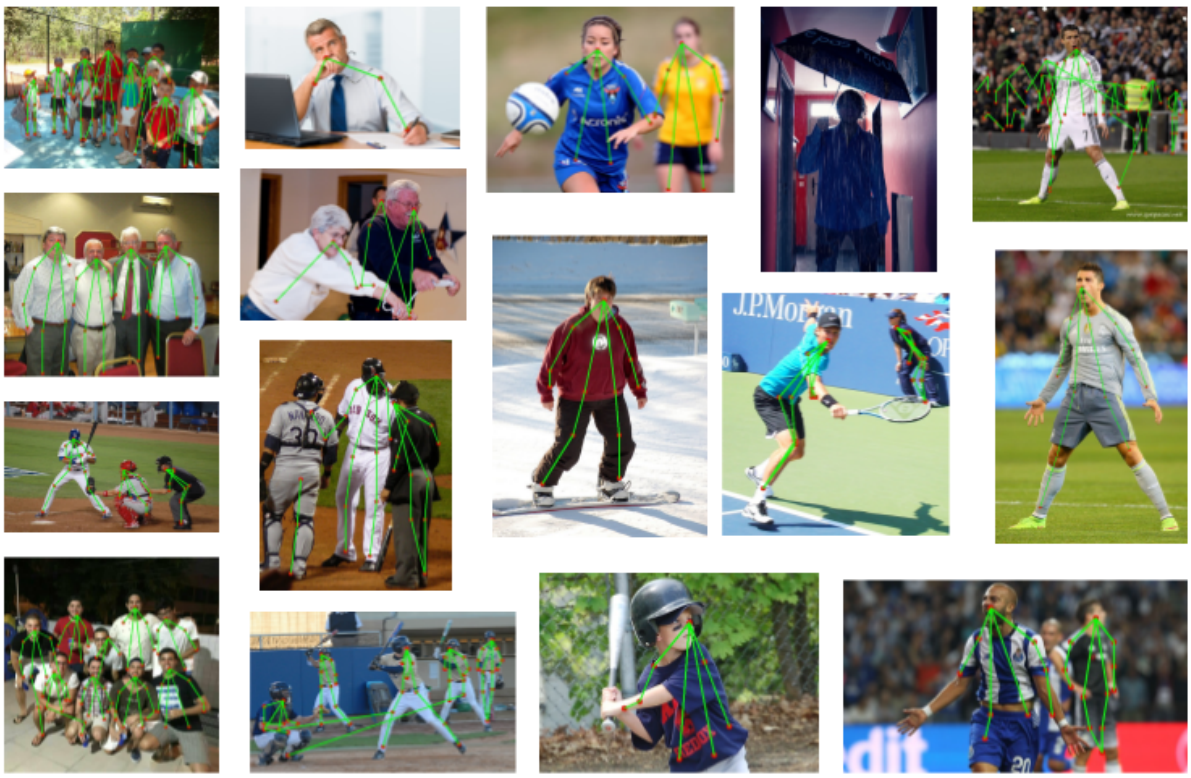
Figura 4.8: Exemplo de desafio de ligação de juntas, ilustrando a ligação entre nariz e ombro esquerdo. Acompanhando as linhas dos mapas de calor das dimensões x e y (imagens superior à direita), é possível visualizar com clareza com qual ombro esquerdo cada nariz detectado está atrelado (imagens superior à esquerda).

5

Resultados e Análise

Com o objetivo de entender a qualidade do método desenvolvido, foi reailzada uma avaliação a partir da base de imagens proveniente do conjunto de validação do *COCO-Dataset 2017*, a qual dispõe de 5.000 imagens diversas, bem como o conjunto de anotações das localizações (em *pixels*) das juntas dos corpos humanos existentes. Também foram utilizadas outras imagens avulsas, a fim de obter retornos visuais do método. É possível observar alguns resultados visuais na Figura 5.1, envolvendo casos onde há a presença de múltiplos indivíduos, oclusão parcial de partes do corpo, poses adversas e iluminação.

Figura 5.1: Exemplo de conjunto de predições em diversas imagens.



5.1 Avaliação Quantitativa

Foi realizada uma avaliação no conjunto de imagens de validação do COCO-Dataset 2017. Nela, extrai-se a informação das juntas dos corpos humanos presentes na imagem e as compara com as anotações de localizações disponibilizadas pela base.

Para entender o cálculo de resultado do método, considere a Figura 5.2 como um mapa de calor de uma junta qualquer. Nela, busca-se calcular a precisão (Precision) e revogação (Recall) do método, a fim de se obter o F-score. Para tal, é necessário contabilizar a quantidade de:

1. **Verdadeiros positivos:** Os *True Positives* (TP) são os casos de pares de juntas (uma sendo a predição, enquanto a outra sendo o *ground truth*), no qual a distância euclidiana, em *pixels*, entre elas é menor que um limiar t. Na Figura 5.2, é representado pelos casos A e B;

2. **Falsos positivos:** Os *False Positives* (FP) são os casos onde uma junta de predição cuja distância euclidiana, em *pixels*, a qualquer outra junta *ground truth* está acima de um limiar t. Na Figura 5.2, é representado pelo caso C;

3. **Falsos negativos:** Os *False Negatives* (FN) são os casos onde uma junta *ground truth* cuja distância euclidiana, em *pixels*, a qualquer outra junta de predição está acima de um limiar t. Na Figura 5.2, é representado pelos casos D e E.

O processo de contagem de TP, FP, e FN é realizado em todas as imagens do conjunto de validação. Por fim, calcula-se o *Precision* (P), *Recall* (R) e *F-score* (FS), onde:

$$P = \frac{TP}{TP + FP} \tag{5.1}$$

$$R = \frac{TP}{TP + FN} \tag{5.2}$$

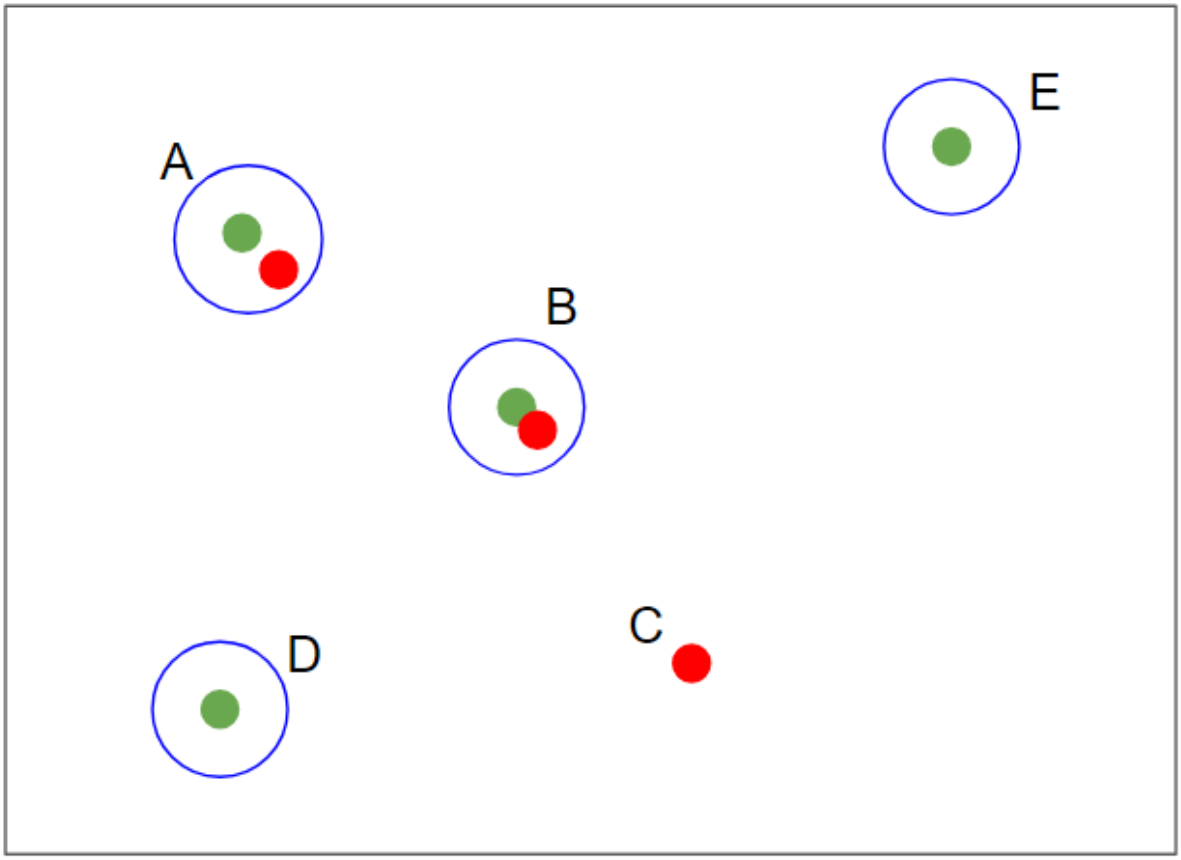
$$FS = 2 * \frac{P * R}{P + R} \tag{5.3}$$

É possível calcular os valores de *Precision*, *Recall* e *F-score* para quaisquer valores de limiar t. A Figura 5.3 ilustra os resultados obtidos para valores de t entre 1 e 50. Já a tabela 5.1 mostra os resultados numericamente para valores de limiares t intermediários.

5.2 Análise

Observando a Figura 5.3, a superioridade dos valores de *Recall* em relação aos valores de *Precision* sob diferentes valores de limiar explicita uma cautela do método desenvolvido, no que diz respeito à extração de juntas dos mapas de calor. Por outro lado, observa-se que resultados melhores podem ser atingidos; uma considerável parcela dos casos de FP e FN deve-se à diferença de quantidade entre juntas preditas e juntas provenientes do *ground truth*. Analisando

Figura 5.2: Exemplo de caso de predição de localização de um tipo de junta. Os pontos em vermelho representam as predições, enquanto que os pontos em verde representam o *ground truth*.



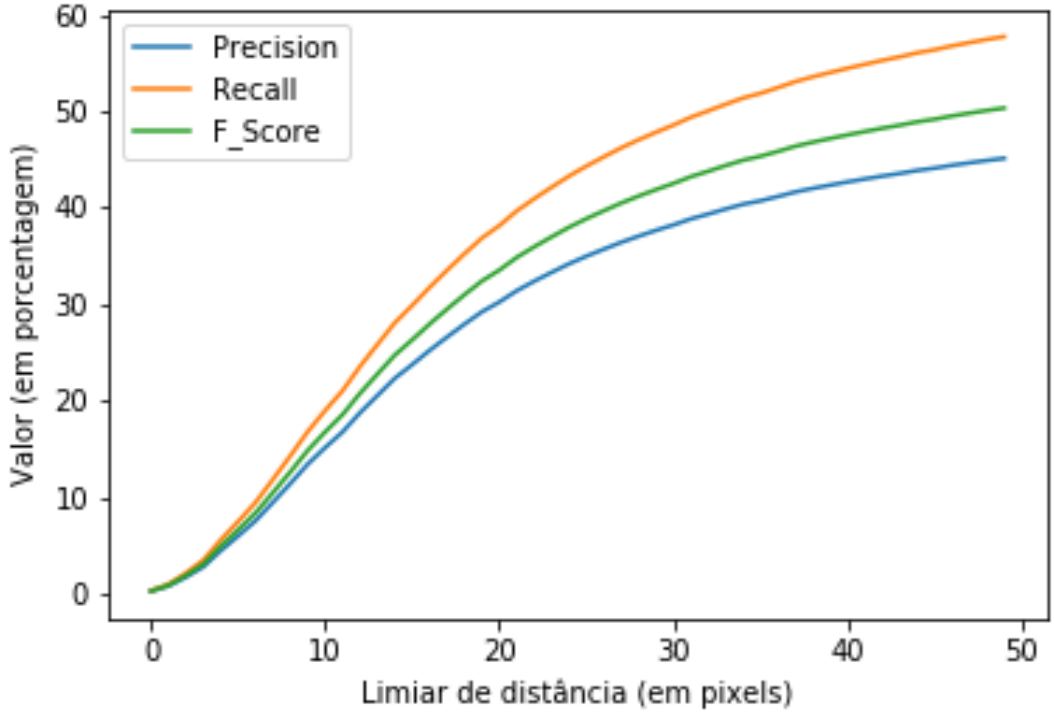
visualmente alguns resultados (Figura 5.1), é possível observar que o método comporta-se como desejado em casos onde seres humanos apresentam-se sob baixa oclusão. Em casos de sobreposição de humanos, houveram muitos casos onde o método confunde-se ao combinar juntas de humanos distintos à mesma pose.

Notou-se uma maior qualidade na extração de posição de juntas de regiões centrais do corpo humano; narizes, ombros e pélvis obtiveram melhores resultados. Isso deve-se ao fato de que, ao utilizar o *COCO-Dataset* para o treinamento da rede neural, esses tipos de juntas são mais presentes nas imagens, quando comparadas com juntas extremas. Com isso, o treinamento das juntas extremas aprende com menos casos de treino, defasando assim a sua qualidade. Não

Tabela 5.1: Resultados de *Precision*, *Recall* e *F-score* para valores de t intermediários.

Limiar (em *pixels*)	*Precision*	*Recall*	*F-score*
10	13,43%	16,80%	14,85%
20	29,22%	36,87%	32,42%
30	37,67%	47,84%	41,89%
40	42,36%	54,03%	47,19%
50	45,14%	57,75%	50,34%

Figura 5.3: Resultados de *Precision*, *Recall* e *F-Score* para diversos valores de limiar t.



só isso, observou-se que, ao predizer pulsos e tornozelos, existiu uma ambiguidade ao realizar a predição dos lados esquerdo e direito; em alguns casos, isso ocasionou em erros na agregação das juntas em poses. Para resolver esse problema, é necessário que haja um aumento no número de imagens cobrindo os casos de possíveis ambiguidades entre juntas de lados diferentes.

Em casos de imagens com baixa iluminação, o uso do método falhou na obtenção de poses; isso se deu pelo fato de que ativação das localizações das juntas não ocorreu com firmeza, quando comparados com imagens de maior iluminação. Por outro lado, imagens bem iluminadas tiveram as suas predições de mapa de calor positivamente intensas (Figura 5.4). Existem potenciais melhorias nesse contexto, a partir do aumento do número de imagens as quais apresentam corpos humanos em ambientes com iluminação reduzida.

Casos de insucesso na predição de poses deu-se na existência de grandes quantidades de pessoas na imagem; esses tipos de casos resultam na predição de mapas de calor "poluídos", tanto para a predição de juntas quanto de ligação de juntas (Figura 5.5). Para esses casos, é necessário que evite-se realizar o treinamento de imagens onde haja a presença excessiva de juntas do corpo humano.

Figura 5.4: Exemplo de predição em uma imagem com baixa iluminação.

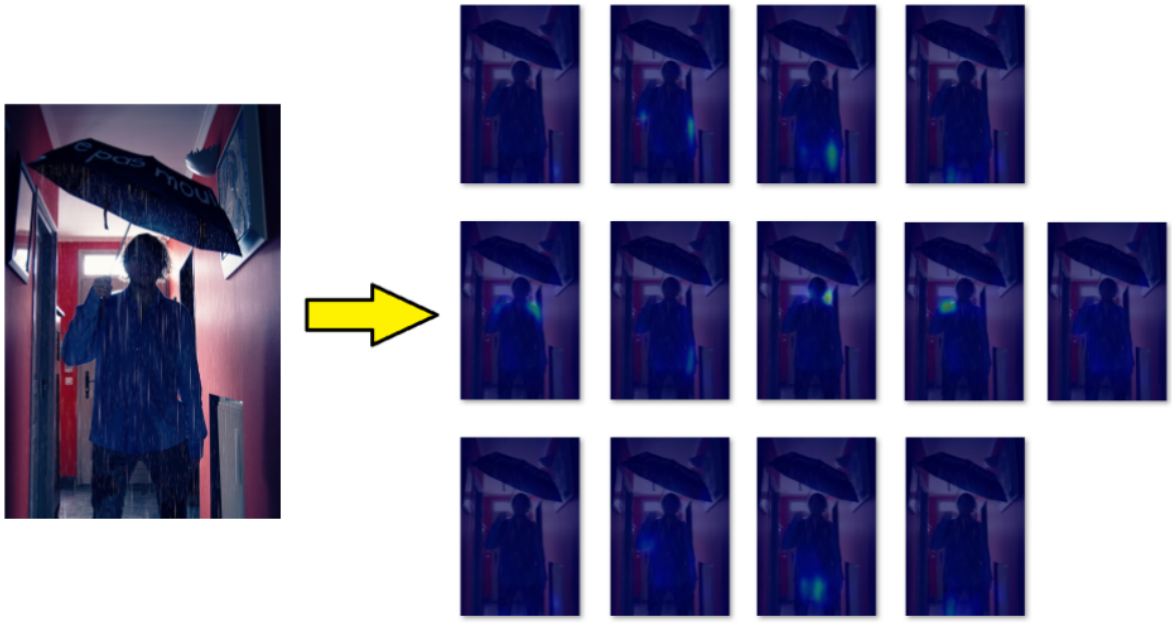


Figura 5.5: Exemplo de predição em uma imagem com múltiplas pessoas ao fundo. Como pode-se ver, a presença de pessoas ao fundo ativa os mapas de calor ao fundo, confundindo a agregação de juntas em poses.

6

Conclusão

O rastreamento de poses humanas mostra-se bastante útil em diversos contextos de aplicações, revelando assim a sua importância. Com o passar do tempo, o avanço tecnológico no contexto de *hardware* tornará o uso de técnicas utilizando CNN para o rastreamento de poses mais acessível, tais como no contexto de aplicações em tempo real e aplicações para dispositivos móveis.

Neste trabalho, notou-se uma dificuldade em seguir a metodologia de outros projetos do estado da arte, uma vez que o poderio computacional utilizado para a concepção dos modelos de redes neurais convolucionais é elevado. O uso do erro médio quadrado como função de erro do treinamento, utilizado nos trabalhos do estado da arte, não foi aplicado com sucesso, assim trazendo resultados negativos na predição dos mapas de calor.

Por outro lado, a limitação em recursos para a realização do treinamento dos modelos permitiu a concepção de ideias paliativas para a obtenção de resultados concretos: a função de erro elaborada neste trabalho, quando aplicada num treinamento reduzido, conseguiu evidenciar predições de juntas e ligações de juntas com maior intensidade, enquanto que, diferentemente de WEI et al. (2016) e MEHTA et al. (2017), o uso da função de erro médio quadrado não apresentou resultados minimamente bons.

Existem pontos de melhoria a serem realizados; dado que o uso de redes neurais pré-treinadas para a realização do Transfer Learning aumenta o custo computacional, há o desejo em atingir resultados satisfatórios com a utilização de redes rasas, como a *MobileNet*. No contexto qualitativo, técnicas de aumento dos dados de treino possuem potencial para serem aplicadas; rotação, alteração de iluminação e cores e aplicação de *Blur* nas imagens são alguns dos exemplos de manipulações que podem enriquecer o treinamento, tornando o modelo mais apto à diversidade das imagens.

ABADI, M. et al. Tensorflow: a system for large-scale machine learning. In: OSDI. **Anais...** [S.l.: s.n.], 2016. v.16, p.265–283.

BERGSTRA, J. et al. Theano: a cpu and gpu math compiler in python. In: PYTHON IN SCIENCE CONF, 9. **Proceedings...** [S.l.: s.n.], 2010. v.1.

CAO, Z. et al. Realtime multi-person 2d pose estimation using part affinity fields. In: CVPR. **Anais...** [S.l.: s.n.], 2017. v.1, n.2, p.7.

CHEN, T. et al. Mxnet: a flexible and efficient machine learning library for heterogeneous distributed systems. **arXiv preprint arXiv:1512.01274**, [S.l.], 2015.

CHOLLET, F. et al. **Keras**. 2015.

CHOU, C.-J.; CHIEN, J.-T.; CHEN, H.-T. Self adversarial training for human pose estimation. **arXiv preprint arXiv:1707.02439**, [S.l.], 2017.

DENG, J. et al. Imagenet: a large-scale hierarchical image database. In: COMPUTER VISION AND PATTERN RECOGNITION, 2009. CVPR 2009. IEEE CONFERENCE ON. **Anais...** [S.l.: s.n.], 2009. p.248–255.

HE, K. et al. Deep residual learning for image recognition. In: IEEE CONFERENCE ON COMPUTER VISION AND PATTERN RECOGNITION. **Proceedings...** [S.l.: s.n.], 2016. p.770–778.

HOWARD, A. G. et al. Mobilenets: efficient convolutional neural networks for mobile vision applications. **arXiv preprint arXiv:1704.04861**, [S.l.], 2017.

KRIZHEVSKY, A.; SUTSKEVER, I.; HINTON, G. E. Imagenet classification with deep convolutional neural networks. In: ADVANCES IN NEURAL INFORMATION PROCESSING SYSTEMS. **Anais...** [S.l.: s.n.], 2012. p.1097–1105.

KUHN, H. W. The Hungarian method for the assignment problem. **Naval research logistics quarterly**, [S.l.], v.2, n.1-2, p.83–97, 1955.

LIN, T.-Y. et al. Microsoft coco: common objects in context. In: EUROPEAN CONFERENCE ON COMPUTER VISION. **Anais...** [S.l.: s.n.], 2014. p.740–755.

MCCULLOCH, W. S.; PITTS, W. A logical calculus of the ideas immanent in nervous activity. **The bulletin of mathematical biophysics**, [S.l.], v.5, n.4, p.115–133, 1943.

MEHTA, D. et al. Vnect: real-time 3d human pose estimation with a single rgb camera. **ACM Transactions on Graphics (TOG)**, [S.l.], v.36, n.4, p.44, 2017.

MEHTA, D. et al. Single-Shot Multi-Person 3D Body Pose Estimation From Monocular RGB Input. **arXiv preprint arXiv:1712.03453**, [S.l.], 2017.

MITCHELL, T. M. et al. **Machine learning. WCB**. [S.l.]: McGraw-Hill Boston, MA:, 1997.

MORI, G.; MALIK, J. Recovering 3d human body configurations using shape contexts. **IEEE Transactions on Pattern Analysis and Machine Intelligence**, [S.l.], v.28, n.7, p.1052–1062, 2006.

NEWELL, A.; YANG, K.; DENG, J. Stacked hourglass networks for human pose estimation. In: EUROPEAN CONFERENCE ON COMPUTER VISION. **Anais...** [S.l.: s.n.], 2016. p.483–499.

PAPANDREOU, G. et al. Towards accurate multiperson pose estimation in the wild. **arXiv preprint arXiv:1701.01779**, [S.l.], v.8, 2017.

REN, X.; BERG, A. C.; MALIK, J. Recovering human body configurations using pairwise constraints between parts. In: COMPUTER VISION, 2005. ICCV 2005. TENTH IEEE INTERNATIONAL CONFERENCE ON. **Anais...** [S.l.: s.n.], 2005. v.1, p.824–831.

ROSENBLATT, F. The perceptron: a probabilistic model for information storage and organization in the brain. **Psychological review**, [S.l.], v.65, n.6, p.386, 1958.

SIMONYAN, K.; ZISSERMAN, A. Very deep convolutional networks for large-scale image recognition. **arXiv preprint arXiv:1409.1556**, [S.l.], 2014.

SZEGEDY, C. et al. Going deeper with convolutions. In: **Anais...** [S.l.: s.n.], 2015.

WEI, S.-E. et al. Convolutional pose machines. In: IEEE CONFERENCE ON COMPUTER VISION AND PATTERN RECOGNITION. **Proceedings...** [S.l.: s.n.], 2016. p.4724–4732.