Estimativa de Pose 2D com Atenção: Um Estudo Comparativo de Mecanismos de Atenção em ResNet-50

Marlon Nanael Leitão Malheiros
Programa de Pós-Graduação em Engenharia Elétrica
Universidade Federal do Pará (UFPA)
Belém, Brasil
marlon.malheiros1@gmail.com

Adriana Rosa Garcez Castro
Programa de Pós-Graduação em Engenharia Elétrica
Universidade Federal do Pará (UFPA)
Belém, Brasil
adcastro@ufpa.br

***Resumo*—A estimação de pose humana 2D é uma tarefa fundamental em Visão Computacional, onde as Redes Neurais Convolucionais (CNNs) têm alcançado resultados notáveis. Mecanismos de atenção podem aprimorar o desempenho dessas redes ao permitir o foco seletivo em características mais relevantes. Este trabalho investiga o impacto da integração de quatro tipos de mecanismos de atenção – Atenção Coordenada, Self-Attention (Auto-Atenção), Atenção de Contexto Global e o Convolutional Block Attention Module (CBAM), juntamente de uma versão modificada deste – em uma arquitetura ResNet-50 para a estimação de pose humana 2D. Os modelos foram treinados e avaliados no conjunto de dados MS COCO, utilizando a métrica Average Precision (AP). Os resultados demonstram que a incorporação do CBAM modificado alcançou o melhor desempenho dentre os modelos avaliados, com 67.8% AP, representando um ganho de 1.6 pontos percentuais sobre o modelo base (66.2% AP) e com um custo computacional adicional mínimo. A Atenção Coordenada também apresentou melhora significativa (67.7% AP). A análise sugere que mecanismos de atenção com ênfase na captura de informações espaciais são alternativas promissoras para esta tarefa.**

***Index Terms*—Estimação de Pose Humana 2D, Mecanismos de Atenção, Redes Neurais Convolucionais, CBAM, MS COCO.**

I. Introdução

A estimativa de pose humana 2D, um problema fundamental e desafiador da área de Visão Computacional, visa localizar pontos anatômicos-chave do corpo humano, a partir de imagens, para inferência postural de indivíduos. Os avanços no campo de Aprendizado de Máquina, mais especificamente das Redes Neurais Convolucionais (CNN), têm impulsionado o desenvolvimento de modelos de estimativa de pose humana mais eficientes e precisos [1].

Diversas abordagens baseadas em CNNs foram empregadas com sucesso para esta tarefa, como em [2], onde os autores apresentam uma arquitetura baseada na ResNet [3], com resultados promissores em conjuntos de dados públicos, ou em [4], onde é apresentado um método robusto e consolidado que utiliza uma estrutura baseada na arquitetura VGG [5] para estimação multi-pessoa em tempo real. Outras pesquisas vêm explorando o uso de CNNs profundas para fazer simultaneamente estimação de pose e reconhecimento de ações [6] ou aplicação de CNNs em configurações de cascata para refinar progressivamente as predições de pose [7].

Os Mecanismos de Atenção vêm se destacando como uma ferramenta poderosa em estruturas de CNN, demonstrando sucesso em diversos problemas do campo da Visão Computacional, como classificação, detecção, segmentação, entre outros. Esses mecanismos permitem que os modelos foquem seletivamente em partes mais relevantes dos dados de entrada, mitigando limitações de abordagens tradicionais e levando a modelos mais robustos e precisos. Exemplos incluem [8], onde o mecanismo de Self-Attention (Auto-atenção) foi utilizado para melhorar a compreensão de contexto global dos dados extraídos pela CNN; em [9] onde foi empregada atenção espacial e na dimensão dos canais para melhorar a qualidade das representações locais e aumentar a precisão; e em [10], onde uma implementação híbrida combina CNNs com blocos de atenção. O crescente destaque de tais abordagens, bem como de modelos puramente baseados em Transformers [11], ressalta a importância fundamental da atenção para a tarefa.

Nesse contexto, este trabalho tem como objetivo principal avaliar o efeito da integração de diferentes mecanismos de atenção em uma rede CNN baseada na arquitetura ResNet-50 [3] para a tarefa de estimativa de pose humana 2D. Foram explorados quatro mecanismos de atenção distintos: Atenção Coordenada [12], Atenção de Contexto Global [13], Self-Attention [14] e o Convolutional Block Attention Module (CBAM) [15], juntamente com uma variação modificada deste último. Os modelos foram avaliados em experimentos utilizando a base de dados pública MS COCO, analisando a métrica primária Average Precision (AP) e suas variantes. Ao integrar estes diferentes mecanismos em um modelo base simples, procurou-se entender melhor como eles podem aprimorar a precisão no processo de estimativa de pose 2D.

II. Estimativa de Pose 2D

Os métodos de estimativa de pose 2D têm por objetivo a localização de pontos anatômicos específicos do corpo humano (como cotovelos, joelhos, olhos etc.) a partir de imagens ou vídeos [16]. Esta é uma tarefa essencial para análise e

estudo do movimento humano, com uma diversa gama de aplicações, como em reconhecimento de ações, animação, robótica, análise biomecânica para esporte e medicina.

A estimativa de pose pode ser categorizada, de forma geral, em pessoa única ou multipessoa. Métodos de pessoa única assumem que há uma única pessoa na imagem de entrada e tradicionalmente requerem uma etapa anterior de detecção da pessoa na imagem. Os métodos multipessoa têm que lidar com mais de uma pessoa na imagem, sendo que primeiro podem ser detectados todos os pontos anatômicos na imagem, para então agrupá-los por indivíduo, ou, alternativamente, pode-se detectar primeiro cada indivíduo na imagem para posterior estimativa dos pontos individualmente.

Existem duas estratégias principais para a localização de pontos anatômicos: modelos desenvolvidos para gerar como saída as coordenadas dos pontos, enquadrando-se na categoria de modelos de regressão (ilustrado na Fig. 1), ou modelos desenvolvidos para gerar como saída um mapa de calor para cada ponto (ilustrado na Fig. 2), que usando uma distribuição 2D, encapsulam a probabilidade de o ponto estar em cada posição, enquadrando-se na categoria de modelos baseados em mapa de calor [17].

Modelos baseados em regressão têm dificuldades em lidar com situações onde mais precisão é necessária, pois geram um único par de coordenadas como saída, o que pode limitar a sua capacidade de lidar com pequenos erros [17]. Enquanto modelos baseados em mapas de calor podem ser mais tolerantes a mesma situação, por serem mais flexíveis quanto à localização gerada, sendo por este motivo uma abordagem mais popular.

III. Redes Neurais Convolucionais

As Redes Neurais Convolucionais (CNNs – Convolutional Neural Networks) são uma classe fundamental de modelo de aprendizado profundo, projetadas para processar dados estruturados em grids, como imagens, elas utilizam camadas convolucionais para aprender características visuais, como bordas simples ou padrões geométricos mais complexos, e são invariantes a pequenas translações e rotações nas imagens graças à natureza das operações convolucionais. Além disso, as CNNs têm camadas de pooling, usadas para reduzir a dimensão espacial das representações, ajudando a reduzir a complexidade computacional e a melhorar a generalização do modelo. Existem diversas arquiteturas de CNNs propostas e amplamente difundidas que servem como base para diversas tarefas, inclusive estimação de pose, como a ResNet [3], DenseNet [18] e VGG [5]. Embora as CNNs sejam mais comumente associadas à visão computacional, elas também podem ser aplicadas a outros diversos domínios.

IV. Mecanismos de atenção

Mecanismos de atenção englobam um conjunto de técnicas que permitem às redes CNN ponderar dinamicamente a importância das características extraídas nas camadas convolucionais, através do cálculo de pesos de atenção. Estes mecanismos são inspirados em como os humanos conseguem focar seletivamente em características específicas de um estímulo visual enquanto ignoram outras. Existem diversos mecanismos de atenção propostos na literatura, sendo que a seguir serão apresentados, em particular, os avaliados neste trabalho.

A. Atenção Coordenada

O mecanismo de atenção coordenada (Coordinate Attention), ilustrado na Fig. 3, visa resolver uma das principais limitações do mecanismo de atenção Squeeze-and-Excitation (SE) proposto em [19], que reside na geração de atenção que é aplicada apenas na dimensão dos canais. Em [19], o mecanismo de atenção recebe como entrada um mapa de características $X \in R^{C\times H\times W}$, realiza uma operação de pooling global na dimensão dos canais para gerar um vetor de pesos $X_A \in R^{C\times 1\times 1}$, que recalibra a entrada.

Embora eficaz em tarefas como classificação de imagens, classificação de cenas e detecção de objetos [19], essa abordagem pode causar a perda de informações espaciais refinadas, cruciais para tarefas que demandam localização precisa como a estimação de pose.

O mecanismo de atenção coordenada consegue capturar informação posicional em duas etapas:

1) Incorporação de Informação Coordenada: O mapa de características de entrada $X \in R^{C\times H\times W}$ é decomposto através de uma operação de average pooling nas dimensões da altura e da largura, resultando em dois vetores $z^h \in R^{C\times H\times 1}$ e $z^w \in R^{C\times 1\times W}$.

2) Geração de Atenção Coordenada: z^h e z^w são concatenados e submetidos a uma convolução compartilhada F_1, e

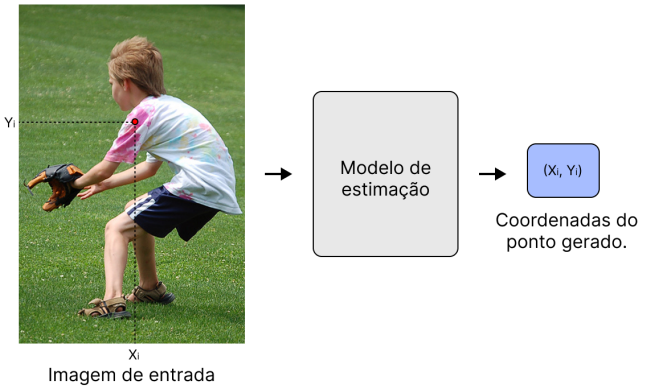


Figura 1. Ilustração do funcionamento de modelo de estimação baseado em regressão.

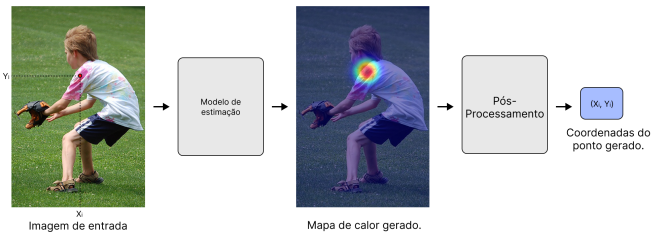


Figura 2. Ilustração do funcionamento de modelo de estimação baseado em mapas de calor.

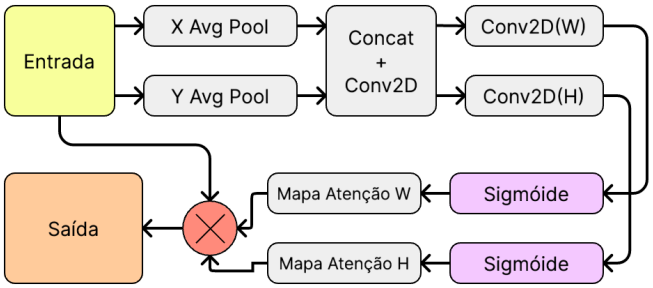


Figura 3. Arquitetura do bloco Atenção Coordenada.

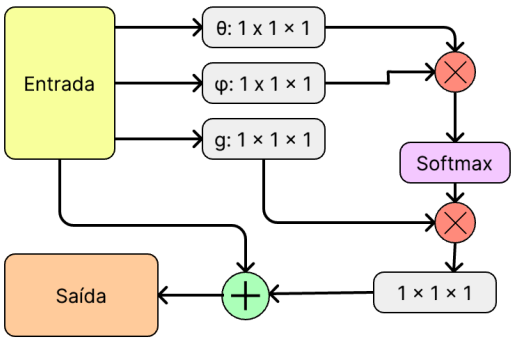


Figura 4. Bloco de Self-Attention.

passados por uma função de ativação não linear δ, resultando em:

$$f = \delta(F_1([z^h, z^w])) \tag{1}$$

O tensor resultante $f \in R^{(C/r)\times(H+W)}$ é um mapa de características intermediário com os canais reduzidos por um fator r definido em F_1. f é então separado em cada componente espacial $f^h \in R^{C/r\times H}$ e $f^w \in R^{C/r\times W}$, processadas por convoluções F^h e F^w para restaurar a dimensionalidade original dos canais em f^h e f^w, e submetidas a função de ativação sigmoide σ, produzindo $g^h = \sigma(F^h(f^h))$ e $g^w = \sigma(F^w(f^w))$.

Que são usados para obter a saída Y do bloco de atenção coordenada.

$$y_c(i,j) = x_c(i,j) \times g_c^h(i) \times g_c^w(j) \tag{2}$$

Esta operação permite que o módulo de Atenção Coordenada realce seletivamente características importantes, considerando tanto as interdependências de canal quanto a informação posicional precisa.

B. Self-Attention

Capturar dependências de longo alcance permite as redes neurais uma melhor capacidade de aprendizado em tarefas complexas. Contudo, abordagens tradicionais que consistem no uso de várias camadas convolucionais ou recorrentes repetidas criam desafios de eficiência computacional e otimização [14].

Para superar essa limitação, as operações não-locais foram propostas por [14], como uma abordagem que permite a captura de dependências de longo alcance de maneira eficiente. Nessa técnica, a resposta em uma posição do mapa de características é influenciada por todas as outras posições. A forma geral do bloco não local é descrita por:

$$\mathbf{y}i = \frac{1}{\mathcal{C}(\mathbf{x})} \sum_{\forall j} f(\mathbf{x}_i, \mathbf{x}_j) g(\mathbf{x}_j) \tag{3}$$

onde f calcula a relação entre as posições i e j, g calcula uma representação da entrada na posição j, e $C(x)$ um fator de normalização.

O mecanismo de Self-Attention, especificamente na forma de Scaled Dot-Product popularizada pela arquitetura de Transformers [20] é uma implementação eficiente dessa técnica. Dado um mapa de características de entrada X, ele é projetado em três representações lineares distintas através de transformações aprendíveis: Consultas (Queries, Q), Chaves (Keys, K), e Valores (Values, V). Em termos simples as Consultas representam o que cada elemento está buscando; as Chaves o que cada elemento de entrada oferece ou qual informação ele contém; e os Valores são as próprias representações das características dos elementos de entrada que serão efetivamente combinadas para formar a saída.

Os pesos de atenção são então calculados medindo a similaridade entre cada Consulta e todas as Chaves. A técnica Scale-Dot-Product faz isso através do produto escalar Q e K, aplica um fator de escalonamento, e em seguida a função de ativação Softmax, resultando na matriz de pesos de atenção A.

A saída final do bloco de Self-attention é obtida através da multiplicação da matriz de atenção A pela matriz de valores V. Isso produz um novo conjunto de representações, onde cada elemento é uma soma ponderada dos Valores, refletindo as interações globais capturadas. Este processo é ilustrado na Fig. 4, frequentemente a saída do bloco de atenção é combinada com a entrada original através de uma conexão residual $Z_i = Saida_i + X_i$, auxiliando o treinamento de redes profundas.

C. Atenção de contexto global

O mecanismo de self-attention, particularmente na forma do bloco não local [14], é eficaz na captura de dependências de longo alcance, beneficiando diversas tarefas de visão computacional. No entanto uma análise mais aprofundada realizada em [13] revelou uma observação importante: os mapas de atenção gerados por blocos não-locais para diferentes posições em um mesmo mapa de características apresentam grandes semelhanças. Isso sugere que o bloco não-local, ao calcular um mapa de atenção distinto para cada posição, poderia estar realizando um esforço computacional redundante, pois a informação de atenção capturada demonstra ser independente da posição.

Com base nisso, [13] propõe o módulo de atenção de Contexto Global (Global Context Attention), como uma abordagem mais eficiente para a modelagem de contexto global. A ideia central é simplificar o processo de geração de atenção, calculando um único vetor global que é compartilhado por todas as posições no mapa de características.

O bloco de Contexto Global ilustrado na Fig. 5, adota uma arquitetura leve e eficaz, que opera em duas etapas principais. Primeiro, uma etapa de captura de contexto utiliza uma camada convolucional 1×1 para transformar o mapa de características de entrada; em seguida o resultado é processado por uma função Softmax, que gera os pesos de atenção representando a importância de cada posição para o contexto global. Diferentemente do bloco não-local, que computa uma matriz de pesos de $N \times N$ (onde N é o número de posições), esta etapa resulta em um vetor de atenção global de dimensão $1 \times N$. Esse vetor de atenção global é usado para ponderar o mapa de características original, resultando em uma representação reduzida do contexto. Uma segunda etapa de transformação de características processa essa representação através de duas convoluções 1×1, com uma camada de normalização e uma de ativação ReLU entre elas. E finalmente, o mapa de características globais agregadas resultante dessa transformação é adicionado de volta ao mapa de características de entrada através de uma conexão residual.

D. CBAM e CBAM Modificado

O Convolutional Block Attention Module (CBAM) [15] é um mecanismo de atenção leve e eficaz, projetado para aprimorar a capacidade de representação de Redes Neurais Convolucionais (CNNs). O CBAM infere mapas de atenção sequencialmente ao longo de duas dimensões distintas: canais e espaço. A intuição é que o módulo aprende o quê focar (atenção de canal) e onde focar (atenção espacial) no mapa de características de entrada. Esses dois submódulos complementares ilustrados na Fig. 6, permitem que a rede aprenda a enfatizar seletivamente características informativas e a suprimir as menos úteis, tanto em termos de quais canais são importantes quanto de quais regiões espaciais contêm a informação mais relevante.

O primeiro componente, o módulo de atenção de canal, foca em explorar as interdependências entre os canais. Dado um mapa de características de entrada $X \in R^{C \times H \times W}$, ele primeiro agrega informação espacial utilizando operações de average-pooling e max-pooling em paralelo, gerando dois vetores descritores distintos que capturam diferentes aspectos dos canais. Ambos os descritores são então processados por uma rede perceptron multicamadas (MLP) compartilhada, que inclui uma camada oculta com redução de dimensionalidade para eficiência. Os vetores de características resultantes da MLP são somados elemento a elemento e, em seguida, passam por uma função de ativação sigmoide para produzir o mapa de atenção de canal final $M_c \in R^{C \times 1 \times 1}$. Este mapa M_c é então multiplicado pelo mapa de características de entrada X para produzir um mapa de características refinado pelos canais.

Posteriormente, o módulo de atenção espacial opera sobre o mapa de características já refinado pelo módulo de canal, com o objetivo de identificar as regiões espaciais mais significativas. Para gerar o mapa de atenção espacial, aplica-se inicialmente operações de average-pooling e max-pooling ao longo da dimensão dos canais, resultando em dois mapas de características 2D. Estes dois mapas são concatenados e, em seguida, processados por uma camada convolucional padrão seguida por uma função sigmoide. Isso produz o mapa de atenção espacial 2D final $M_s \in \mathbb{R}^{1 \times H \times W}$. Este mapa M_s é então multiplicado elemento a elemento pelo mapa de características que entrou neste módulo, resultando no mapa de características final do bloco CBAM, que foi refinado tanto nos canais quanto espacialmente.

Adicionalmente às investigações com o CBAM em sua formulação original, este trabalho também avaliou uma versão modificada, referida como CBAM Modificado. A alteração principal nesta variante reside no módulo de atenção de canal: em vez de utilizar as operações de average-pooling e

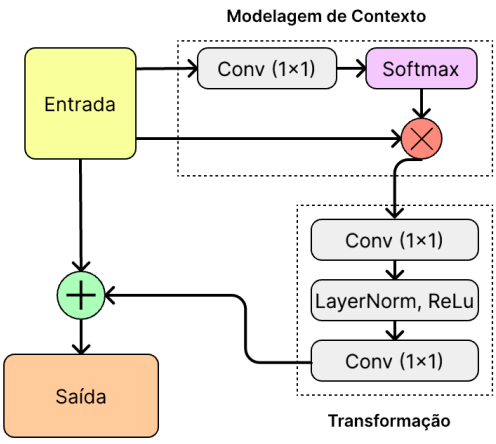


Figura 5. Bloco de Contexto Global.

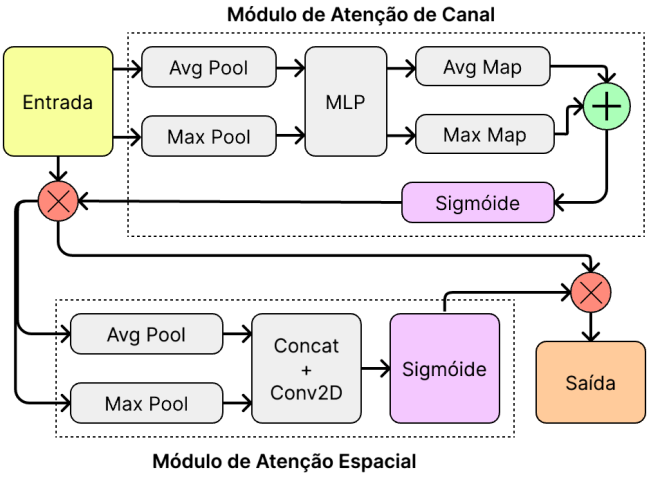


Figura 6. Arquitetura do módulo de atenção CBAM.

max-pooling em paralelo para gerar os descritores de canal (ilustrado na Fig. 6), empregou-se apenas uma operação de global average-pooling. Esta modificação simplifica o módulo de atenção de canal, tornando seu mecanismo de agregação de informação espacial para os canais similar ao do bloco Squeeze-and-Excitation [19]. O módulo de atenção espacial permaneceu inalterado em relação à arquitetura CBAM original.

V. Metodologia

A. Banco de dados

O banco de dados público Microsoft COCO: Common Objects in Context (MS COCO) foi utilizado em todos os experimentos para desenvolvimento e avaliação dos modelos propostos. O banco de dados é composto por imagens comuns do dia a dia, em seu ambiente natural, sendo um conjunto de dados amplamente utilizado como referência em tarefas como detecção, segmentação, geração de legendas e outras [21]. A versão de 2017 foi escolhida para os experimentos, sendo que contém 118.000 imagens para treinamento e 5.000 imagens para validação. Cada instância no conjunto de dados COCO apresenta no máximo 17 pontos anotados, como ilustrado na Fig. 7. O número de pontos varia devido a sua visibilidade (pontos oclusos não são anotados).

B. Modelos para estimativa de pontos anatômicos

1) Modelo sem mecanismos de atenção: O modelo de CNN base sem mecanismo de atenção escolhido, conforme apresentado na Fig. 8, baseia-se na estrutura proposta em [22]. Esta é uma das arquiteturas mais simples e, ainda assim eficaz para realizar a estimativa de pontos 2D, sendo composta por uma rede backbone ResNet-50 [3] utilizada como extratora de características e um bloco de estimação com 3 camadas de deconvolução e uma camada convolucional final para estimação, sendo que o modelo gera na saída um mapa de calor 2D para cada ponto anatômico-chave a ser estimado. O processo de estimação dos pontos a partir de cada mapa de calor gerado na saída será explicado em detalhes em uma seção posterior.

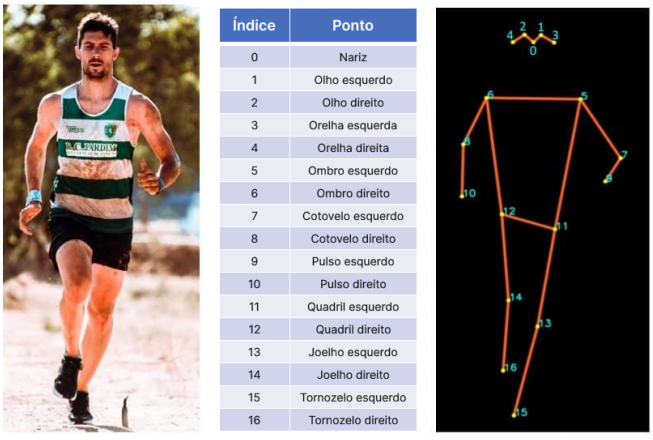
2) Modelos com mecanismos de atenção: Nos modelos com mecanismo de atenção, os blocos de atenção foram inseridos sempre na mesma posição da rede CNN base, para cada um dos experimentos, visando analisar o impacto na qualidade da estimativa de pontos realizada.

Especificamente, para cada modelo com mecanismo de atenção, o bloco de atenção foi inserido na estrutura da Fig. 8, após a rede backbone ResNet-50, de modo que o mapa de características extraído por esta estrutura fosse utilizado como entrada para o mecanismo de atenção, conforme ilustrado na Fig. 9. A função principal do bloco de atenção é refinar esse mapa de características, destacando as informações mais relevantes ou suprimindo as menos necessárias, antes de passá-las ao bloco de estimação dos pontos.

C. Extração de pontos a partir de mapas de calor

O modelo de estimativa de pose gera um conjunto de 17 mapas de calor (um para cada ponto a ser estimado) com a resolução de (64x48) pixels. Para extrair as coordenadas 2D a partir destes mapas de calor durante a inferência, é utilizado um procedimento de múltiplos passos inspirado em [9], [22], [23], sendo eles:

1) Aplicação de Test-Time Augmentation para refinar a saída do modelo: Seguindo a metodologia apresentada em [22], a técnica Test-Time Augmentation (TTA) é usada para refinar a previsão do modelo, sendo que esta consiste em primeiro computar a saída para a imagem original e outras versões submetidas a algum processo de aumento de dados (data augment) como espelhamento ou rotação, para em seguida agregar cada saída em um único objeto, onde neste trabalho utilizou-se uma média simples como ilustrado na Fig. 10



Índice	Ponto
0	Nariz
1	Olho esquerdo
2	Olho direito
3	Orelha esquerda
4	Orelha direita
5	Ombro esquerdo
6	Ombro direito
7	Cotovelo esquerdo
8	Cotovelo direito
9	Pulso esquerdo
10	Pulso direito
11	Quadril esquerdo
12	Quadril direito
13	Joelho esquerdo
14	Joelho direito
15	Tornozelo esquerdo
16	Tornozelo direito

Figura 7. Banco de dados MS COCO.

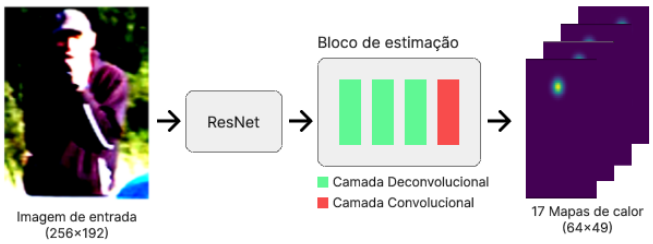


Figura 8. Arquitetura do modelo sem mecanismos de atenção.

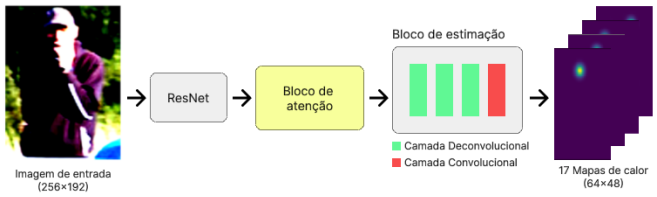


Figura 9. Arquitetura dos modelos com mecanismo de atenção.

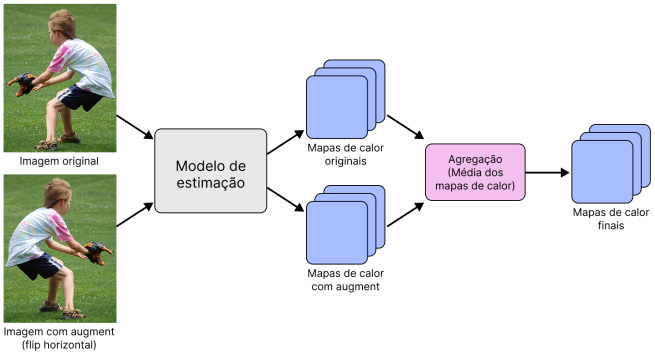


Figura 10. TTA para refinar a saída do modelo.

2) Aplicação de Desfoque Gaussiano: Um desfoque Gaussiano (Gaussian blur) com $\sigma = 2$ e kernel 3x3, é aplicado ao conjunto de mapas de calor para suavizar a distribuição.

3) Detecção de Pico: A localização (x_i, y_i) primária da ativação máxima é encontrada para cada mapa de calor, conforme ilustrado na Fig. 11.

4) Refinamento Sub-pixel: Para aumentar a precisão, as coordenadas (x_i, y_i) de cada mapa de calor são ajustadas através de um deslocamento de um quarto de pixel, na direção do maior para o segundo maior pico encontrado no mapa de calor, ilustrado na Fig. 12.

5) Escalonamento das coordenadas finais: As coordenadas refinadas são então escalonadas linearmente de volta para o tamanho da imagem de entrada, para avaliação e visualização.

VI. Configuração Experimental

A biblioteca PyTorch foi utilizada para implementar todos os modelos de CNN utilizados nos experimentos e para a realização de parte das tarefas de pré-processamento. Alguns parâmetros para treinamento foram baseados em [22], visando melhorar o desempenho dos modelos, como: 1) tamanho da imagem de entrada de (256 x 192) pixels; 2) tamanho dos mapas de calor de saída de (64 x 48) pixels; 3) operações de aumento de dados durante treinamento: escalonamento aleatório (na faixa de 30%), rotação aleatória (40%), e espelhamento horizontal (com probabilidade de 50%).

Para o treinamento dos modelos, foi utilizado o otimizador Adam (Adaptive Momentum) [24] com weight decay de 1×10^{-4}. Para todos os modelos foi utilizada a taxa de aprendizado no valor de 1×10^{-3}, em conjunto com o escalonador de taxa de aprendizado. Este escalonador reduziu a taxa de aprendizado por um fator de 0.5 após 8 épocas sem redução observada no loss para o subconjunto de validação. O treinamento foi finalizado com parada antecipada (early stopping) após 20 épocas sem melhoria no desempenho.

Todos os modelos utilizaram lotes (batches) de 64 imagens, com a função de custo a ser minimizada sendo a função Erro Quadrático Médio (MSE), comumente utilizada para esta tarefa. Todos os experimentos foram executados em uma única GPU NVIDIA RTX 4070.

A. Métricas de avaliação

1) Average Precision (AP): A métrica padrão para avaliar modelos de estimação de pose humana, especialmente aqueles treinados no conjunto de dados MS COCO [21], é a Average Precision (AP). O cálculo da AP para esta tarefa baseia-se na Similaridade de Pontos-Chave do Objeto (Object Keypoint Similarity - OKS) de cada instância de pessoa detectada.

A OKS é uma medida que quantifica a proximidade entre os pontos preditos pelo modelo e os pontos de referência. Ela considera fatores como a distância euclidiana entre os pontos correspondentes, a escala do objeto (normalmente baseada na área do segmento do objeto) e constantes específicas por tipo de ponto chave, que ajustam a sensibilidade à distância (de modo que erros em pontos mais difíceis de localizar ou com maior variabilidade natural são menos penalizados do que erros em pontos mais estáveis). Uma predição de pose é considerada um verdadeiro positivo (TP) para um determinado limiar de OKS se o valor de OKS calculado for maior ou igual a esse limiar; caso contrário é um falso positivo (FP).

A AP é então calculada a partir da curva de precision-recall. Precision indica a proporção de previsões corretas entre todas as previsões feitas, enquanto Recall indica a proporção de instâncias de referências que foram preditas corretamente. O valor de AP sumariza o formato dessa curva, fornecendo um único valor que representa o desempenho do modelo em diferentes níveis de confiança de predição. Uma AP mais alta indica melhor desempenho.

Neste trabalho foram usadas as seguintes métricas de AP, seguindo a convenção de avaliação para o dataset MS COCO:

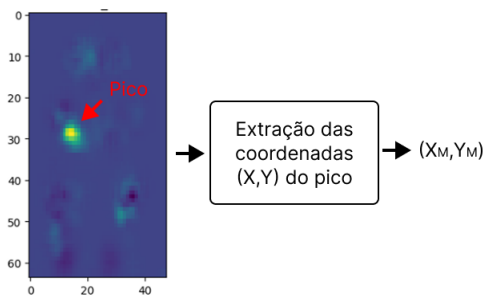


Figura 11. Detecção do pico no mapa de calor.

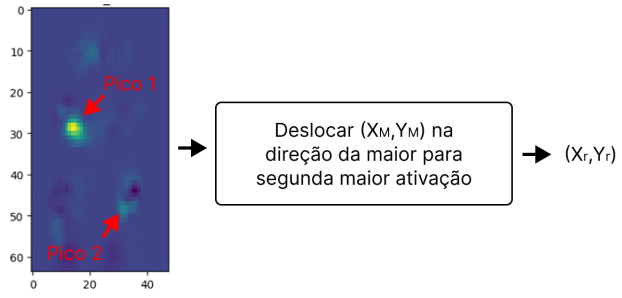


Figura 12. Refinamento sub-pixel da localização.

- AP: É a métrica primária. Representa a média de AP calculada sobre dez diferentes limiares de OKS, variando de 0.5 a 0.95 com incrementos de 0.05. Esta métrica fornece uma avaliação robusta do desempenho em um espectro de exigências de precisão.

- $AP^{OKS=0.5}$: AP calculada utilizando um limiar de OKS de 0.5. Este limiar avalia a capacidade do modelo em localizar pontos de forma mais geral.

- $AP^{OKS=0.75}$: AP calculada utilizando um limiar de OKS 0.75. Este limiar exige uma localização mais precisa dos pontos, e portanto é mais rigoroso.

VII. Resultados e discussões

A Tabela 1 apresenta os resultados dos experimentos, comparando o desempenho do modelo base de estimação (sem atenção) com suas variantes integradas com diferentes mecanismos de atenção. Foram testadas duas variantes do CBAM: a implementação padrão que utiliza Average Pooling e Max Pooling em seu módulo de atenção de canal, e uma versão modificada, cujo módulo de atenção de canal emprega apenas Global Average Pooling, emulando o mecanimo Squeeze-and-Excitation [19].

Os resultados indicam que a incorporação de determinados mecanismos de atenção pode aprimorar o desempenho do modelo base, com destaque para aqueles que integram explicitamente componentes de atenção espacial. Analisando cada um dos modelos em particular:

A. CBAM

Ambas as variantes do CBAM demonstraram ser as mais eficazes neste estudo. O CBAM Modificado, alcançou o melhor desempenho geral com 67.8% de AP. Isso representa um ganho de 1.6 pontos percentuais (p.p.) em relação ao modelo base. A versão CBAM original, obteve um desempenho muito próximo, com 67.6% de AP (ganho de 1.4 p.p.). Notavelmente, ambas as variantes do CBAM atingiram o melhor resultado conjunto na métrica AP@0.75 (74.7%), que avalia o desempenho sob um limiar de OKS mais rigoroso (0.75).

Este resultado sugere que a arquitetura geral do CBAM, que sequencialmente infere atenção ao longo da dimensão dos canais e depois ao longo da dimensão espacial, se destaca particularmente na precisão da localização dos pontos-chave. Essa abordagem, que seleciona características de canais importantes e depois destaca regiões espaciais de interesse, parece ser intrinsecamente benéfica para a estimação de pose, uma tarefa que demanda tanto a seleção de características relevantes quanto a focalização espacial precisa.

A ligeira vantagem observada para a variante CBAM Modificado sobre a versão original, nesta tarefa específica, é um resultado interessante. Uma hipótese para essa diferença marginal é que, para a arquitetura ResNet-50 e o dataset MS COCO neste contexto de estimação de pose, a informação estatística global capturada pelo Global Average Pooling para cada canal já é suficientemente rica e estável para o módulo de atenção de canal. A adição do Max Pooling, embora teoricamente capaz de capturar características mais salientes, poderia, em alguns casos, introduzir uma variabilidade ou ruído que não se traduz em um benefício adicional ou pode ser marginalmente menos ideal para a subsequente inferência de atenção espacial quando comparado à informação mais suavizada do Global Average Pooling. No entanto, é importante notar que a diferença de desempenho entre as duas variantes é pequena, e ambas demonstram a eficácia superior em relação aos outros módulos de atenção testados.

Tabela I
AP calculado para o dataset de validação MS COCO Val 2017.

Modelo	AP	$AP^{0.5}$	$AP^{0.75}$
Sem atenção	66.2%	88%	72.5%
Atenção Coordenada	67.7%	89.2%	73.7%
Atenção de Contexto Global	66.8%	88.2%	73.5%
Self-Attention	66.6%	88.1%	72.6%
CBAM	67.6%	88.2%	74.7%
CBAM Modificado	67.8%	89.1%	74.7%

B. Atenção Coordenada

O modelo com Atenção Coordenada atingiu 67.7% de AP (ganho de 1.5 p.p. sobre o modelo base). Este mecanismo se destacou na métrica AP@0.5 com 89.2%. A eficácia da Atenção Coordenada reforça a importância de capturar dependências direcionais de longo alcance e preservar informações posicionais precisas ao longo dos eixos coordenados. Para uma tarefa inerentemente espacial como a estimação de pose, essa capacidade de entender relações espaciais orientadas é claramente benéfica.

C. Atenção de Contexto Global

O modelo com Atenção de Contexto Global apresentou um ganho mais modesto sobre o modelo base, de 0.6 p.p. na métrica AP (66.8%). Embora positivo e com um bom desempenho em AP@0.75 (73.5%), seu impacto foi menor que o do CBAM e Atenção Coordenada. Isso pode indicar que, embora a modelagem do contexto global seja útil, a informação espacial mais explícita e localizada fornecida pelo CBAM e pelo módulo de Atenção Coordenada pode ser mais crucial para melhorar a precisão nesta tarefa com a arquitetura base utilizada.

D. Self-Attention

A variante com Self-Attention apresentou o menor ganho de desempenho para modelo base, considerando todos os outros modelos testados, sendo o ganho de apenas 0.4 p.p. (66.6% AP), e de 0.1 p.p. nas métricas secundárias. Como descrito em [13], o mecanismo de Self-Attention, quando implementado na forma do bloco não-local, gera mapas de atenção muito semelhantes para todas a posições espaciais do vetor de características de entrada, indicando uma generalização excessiva das relações espaciais, tornando essa abordagem em particular menos adequada ou otimizada para estimação de pose humana.

Tabela II
NÚMERO DE PARÂMETROS E FLOPS PARA CADA MODELO ANALISADO.

Modelo	N. Parâmetros	FLOPs
Sem atenção	34.00 M	19.40 G
Atenção Coordenada	34.53 M	19.42 G
Atenção de Contexto Global	34.53 M	19.41 G
Self-Attention	39.25 M	19.91 G
CBAM	34.53 M	19.41 G

E. Análise de Desempenho e Eficiência

Os resultados demonstram consistentemente que mecanismos com forte componente de modelagem espacial, como CBAM e Atenção Coordenada, proporcionam os maiores ganhos de desempenho. Conforme a Tabela II, esses módulos se destacam também pela eficiência, adicionando apenas 1.5% de parâmetros e um custo de FLOPS negligenciável (¡0.1%). Em contraste, o Self-Attention impôs um custo computacional significativamente maior (15.4% mais parâmetros) para um ganho de desempenho inferior, reforçando a superioridade das abordagens mais leves para esta aplicação.

F. Contextualização

A eficácia do CBAM e da Atenção Coordenada é consistente com os ganhos relativos reportados na literatura para outras tarefas de visão, como detecção de objetos [12], [15], validando a generalização desses módulos para a localização precisa exigida pela estimação de pose. O desempenho mais modesto das abordagens com ênfase em atenção global [13], [14] neste estudo sugere que, para esta tarefa, o refinamento espacial explícito é mais benéfico do que a captura de contexto global. Estes achados indicam que a escolha de mecanismos de atenção eficientes e com viés espacial explícito é uma estratégia promissora para tarefas de alta precisão como a estimação de pose, uma tendência corroborada por arquiteturas de ponta na área que empregam atenção estruturada e localmente focada [9], [25].

VIII. CONCLUSÃO

Este trabalho investigou o impacto de diferentes mecanismos de atenção (Atenção Coordenada, Self-Attention, Atenção de Contexto Global e CBAM e CBAM Modificado) em uma arquitetura ResNet-50 para estimação de pose humana 2D no dataset MS COCO. Conforme os resultados obtidos, a incorporação de módulos de atenção demonstrou potencial para aprimorar o desempenho do modelo base. Notavelmente, mecanismos com forte componente de atenção espacial, como o CBAM modificado (que alcançou 67.8% AP, ganho de 1.6 p.p. sobre o base de 66.2% AP) e a Atenção Coordenada (67.7% AP), superaram consistentemente abordagens como a Self-Attention genérica, e a Atenção de Contexto Global nesta configuração experimental, com mínimo custo computacional adicional. Estes resultados mostram que, para a arquitetura simples avaliada, o refinamento explícito de características espaciais é mais benéfico do que mecanismos de atenção mais generalistas. Isso abre perspectivas para trabalhos futuros que explorem estes módulos de atenção espacial em arquiteturas mais complexas e com maior capacidade, visando alcançar patamares de desempenho ainda mais elevados na estimação de pose.

REFERÊNCIAS

[1] T. L. Munea, Y. Z. Jembre, H. T. Weldegebriel, L. Chen, C. Huang, and C. Yang, "The Progress of Human Pose Estimation: A Survey and Taxonomy of Models Applied in 2D Human Pose Estimation," *IEEE Access*, vol. 8, pp. 133 330–133 348, 2020.

[2] E. Insafutdinov, L. Pishchulin, B. Andres, M. Andriluka, and B. Schiele, "DeeperCut: A Deeper, Stronger, and Faster Multi-Person Pose Estimation Model," Nov. 2016.

[3] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," Dec. 2015.

[4] Z. Cao, G. Hidalgo, T. Simon, S.-E. Wei, and Y. Sheikh, "OpenPose: Realtime Multi-Person 2D Pose Estimation using Part Affinity Fields," May 2019.

[5] K. Simonyan and A. Zisserman, "Very Deep Convolutional Networks for Large-Scale Image Recognition," Apr. 2015.

[6] R. Sun, Q. Zhang, C. Luo, J. Guo, and H. Chai, "Human action recognition using a convolutional neural network based on skeleton heatmaps from two-stage pose estimation," *Biomimetic Intelligence and Robotics*, vol. 2, no. 3, p. 100062, Sep. 2022.

[7] A. Bulat and G. Tzimiropoulos, "Human pose estimation via Convolutional Part Heatmap Regression," 2016, pp. 717–732.

[8] H. Xia and T. Zhang, "Self-Attention Network for Human Pose Estimation," *Applied Sciences*, vol. 11, no. 4, p. 1826, Jan. 2021.

[9] Y. Cai, Z. Wang, Z. Luo, B. Yin, A. Du, H. Wang, X. Zhang, X. Zhou, E. Zhou, and J. Sun, "Learning Delicate Local Representations for Multi-Person Pose Estimation," Jul. 2020.

[10] Y. Yuan, R. Fu, L. Huang, W. Lin, C. Zhang, X. Chen, and J. Wang, "HRFormer: High-Resolution Transformer for Dense Prediction," Nov. 2021.

[11] Y. Xu, J. Wang, Z. Wang, W. Chen, and Y.-C. Wang, "ViTPose: Simple vision transformer baselines for pose estimation," in *Advances in Neural Information Processing Systems*, vol. 35, 2022, pp. 33 324–33 337.

[12] Q. Hou, D. Zhou, and J. Feng, "Coordinate Attention for Efficient Mobile Network Design," Mar. 2021.

[13] Y. Cao, J. Xu, S. Lin, F. Wei, and H. Hu, "GCNet: Non-local Networks Meet Squeeze-Excitation Networks and Beyond," Apr. 2019.

[14] X. Wang, R. Girshick, A. Gupta, and K. He, "Non-local Neural Networks," Apr. 2018.

[15] S. Woo, J. Park, J.-Y. Lee, and I. S. Kweon, "CBAM: Convolutional Block Attention Module," Jul. 2018.

[16] C. Zheng, W. Wu, C. Chen, T. Yang, S. Zhu, J. Shen, N. Kehtarnavaz, and M. Shah, "Deep Learning-Based Human Pose Estimation: A Survey," Jul. 2023.

[17] L. Zhou, X. Meng, Z. Liu, M. Wu, Z. Gao, and P. Wang, "Human Pose-based Estimation, Tracking and Action Recognition with Deep Learning: A Survey," Oct. 2023.

[18] G. Huang, Z. Liu, L. van der Maaten, and K. Q. Weinberger, "Densely Connected Convolutional Networks," Jan. 2018.

[19] J. Hu, L. Shen, S. Albanie, G. Sun, and E. Wu, "Squeeze-and-Excitation Networks," May 2019.

[20] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention Is All You Need," Aug. 2023.

[21] T.-Y. Lin, M. Maire, S. Belongie, L. Bourdev, R. Girshick, J. Hays, P. Perona, D. Ramanan, C. L. Zitnick, and P. Dollár, "Microsoft COCO: Common Objects in Context," Feb. 2015.

[22] B. Xiao, H. Wu, and Y. Wei, "Simple Baselines for Human Pose Estimation and Tracking," Aug. 2018.

[23] Y. Chen, Z. Wang, Y. Peng, Z. Zhang, G. Yu, and J. Sun. Cascaded Pyramid Network for Multi-Person Pose Estimation. arXiv.org. [Online]. Available: https://arxiv.org/abs/1711.07319v2

[24] I. Loshchilov and F. Hutter, "Decoupled Weight Decay Regularization," Jan. 2019.

[25] Y. Li, S. Wang, T. Wang, Z. Zhang, and X. Chen, "Tokenpose: Learning keypoint tokens for human pose estimation," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 11 324–11 333.