Universidade Federal de Juiz de Fora
Instituto de Ciências Exatas
Bacharelado em Ciência da Computação

Aplicação de modelos de reconhecimento de ações em vídeos para o problema de monitoramento de idosos

Beatriz Aparecida Benedicto Heleno

JUIZ DE FORA
JANEIRO, 2026

Aplicação de modelos de reconhecimento de ações em vídeos para o problema de monitoramento de idosos

Beatriz Aparecida Benedicto Heleno

Universidade Federal de Juiz de Fora

Instituto de Ciências Exatas

Departamento de Ciência da Computação

Bacharelado em Ciência da Computação

Orientador: Luiz Maurílio da Silva Maciel

JUIZ DE FORA

JANEIRO, 2026

Aplicação de modelos de reconhecimento de ações em vídeos para o problema de monitoramento de idosos

Beatriz Aparecida Benedicto Heleno

MONOGRAFIA SUBMETIDA AO CORPO DOCENTE DO INSTITUTO DE CIÊNCIAS EXATAS DA UNIVERSIDADE FEDERAL DE JUIZ DE FORA, COMO PARTE INTEGRANTE DOS REQUISITOS NECESSÁRIOS PARA A OBTENÇÃO DO GRAU DE BACHAREL EM CIÊNCIA DA COMPUTAÇÃO.

Aprovada por:

Luiz Maurílio da Silva Maciel
Doutor em Engenharia de Sistemas de Computação

Saulo Moraes Villela
Doutor em Engenharia de Sistemas de Computação

Marcelo Bernardes Vieira
Doutor em Ciência da Computação

JUIZ DE FORA
9 DE JANEIRO, 2026

Aos meus amigos, familiares e ídolos

Resumo

A garantia da qualidade de vida da população idosa é um tema central diante do envelhecimento global, e os sistemas de Ambiente de Vida Assistida (*Ambient Assisted Living* – AAL) têm se mostrado essenciais para o monitoramento de atividades diárias. Esses sistemas integram diferentes tecnologias para apoiar a autonomia, segurança e bem-estar de idosos em seus ambientes domésticos, permitindo detectar atividades, comportamentos de risco e situações de emergência. Neste trabalho, aplica-se o *framework* PoseConv3D ao conjunto de dados Toyota Smarthome Trimmed, composto por vídeos de atividades de vida diária de idosos em ambientes reais, para o reconhecimento de ações a partir de vídeos. O modelo utiliza a estimativa de pose humana para gerar mapas de calor 2D das articulações ao longo do tempo, que são então processados por uma rede neural convolucional tridimensional, permitindo capturar padrões espaciais e temporais das ações. Nos experimentos, o modelo alcançou 72,2% de acurácia global e 54,5% de acurácia média por classe, superando em acurácia média o desempenho do modelo proposto no trabalho original do conjunto Toyota Smarthome Trimmed. Considerando a alta granularidade do conjunto, que inclui 31 classes de ações com variações posturais e atividades visualmente semelhantes, foi adotada uma estratégia adicional de agrupamento semântico, reduzindo as classes para 19 categorias-base, permitindo avaliar o desempenho do modelo em termos de padrões mais amplos de comportamento de idosos. O modelo sobre o conjunto agrupado apresentou uma acurácia global de 77,7% e uma acurácia média por classe de 67,9%.

Palavras-chave: Reconhecimento de Ações Humanas, Aprendizado Profundo, Redes Neurais Convolucionais, Visão Computacional, Ambiente de Vida Assistida, Monitoramento de Idosos.

Abstract

Ensuring the quality of life for the elderly population is a central theme in the face of global aging, and Ambient Assisted Living (AAL) systems have proven essential for monitoring daily activities. These systems integrate different technologies to support the autonomy, safety, and well-being of older adults in their home environments, allowing the detection of activities, risky behaviors, and emergency situations. In this work, the PoseConv3D framework is applied to the Toyota Smarthome Trimmed dataset, composed of videos of daily living activities of older adults in real environments, for action recognition from videos. The model uses human pose estimation to generate 2D heat maps of joints over time, which are then processed by a three-dimensional convolutional neural network, allowing the capture of spatial and temporal patterns of actions. In the experiments, the model achieved 72.2% overall accuracy and 54.5% average accuracy per class, surpassing the performance on mean class accuracy of model proposed in the original Toyota Smarthome Trimmed study. Considering the high granularity of the dataset, which includes 31 classes of actions with postural variations and visually similar activities, an additional semantic clustering strategy was adopted, reducing the classes to 19 base categories, allowing the model's performance to be evaluated in terms of broader patterns of older adult behavior. The model on the grouped set showed an overall accuracy of 77.7% and an average accuracy per class of 67.9%.

Keywords: Human Action Recognition, Deep Learning, Convolutional Neural Networks, Computer Vision, Ambient Assisted Living, Elderly Monitoring.

Agradecimentos

Aos meus amigos, os próximos e aqueles que nunca encontrei pessoalmente, e ao meu psicólogo pelo encorajamento e confiança no meu potencial.

Ao meu professor orientador Luiz pelo apoio e por ter despertado minha curiosidade, interesse e entusiamo com a área de visão computacional, isso transformou completamente minha vida.

À minha mãe, minha irmã e meus avós que me cuidaram e me propiciaram estar aqui hoje. Em especial a minha avó Maria da Glória Benedicto, minha segunda mãe, foi a inspiração e motivação do meu tema de trabalho e por isso desejo eternizá-la por meio do mesmo. A ela minha eterna e imensa gratidão, espero que meu trabalho um dia possa fazer a diferença na vida de outras famílias.

"Lift your head up high and scream out to the world I know I am someone, and let the truth unfurl" - Michael Jackson.

-

Conteúdo

Lista de Figuras

Lista de Tabelas

Lista de Abreviações

AAL	Ambient Assisted Living
AVDs	Atividades de Vida Diária
CNN	Convolutional Neural Networks
CS	Cross-Subject
DIF	Diferença absoluta de desempenho entre subconjuntos de dados
FLOPs	Floating Point Operations
GCN	Graph Convolutional Network
HOG	Histogram of Oriented Gradients
IBGE	Instituto Brasileiro de Geografia e Estatística
IoT	Internet of Things
MEI	Motion Energy Image
MHI	Motion History Image
OF	Optical Flow
PS	Pictorial Structures
RAH	Reconhecimento de Ações Humanas
RGB	Red Green Blue
RGB-D	Red Green Blue - Depth
RIPSA	Rede Interagencial de Informações para Saúde
RNN	Recurrent Neural Network
SIFT	Scale-Invariant Feature Transform
TST	Toyota Smarthome Trimmed

1 Introdução

De acordo com o relatório de Perspectivas da População Mundial 2024 das Nações Unidas (ONU, 2024), a transição demográfica global caminha para um marco histórico: até meados da década de 2050, as pessoas com 65 anos ou mais representarão cerca de 18% da população mundial. No Brasil, essa realidade é ainda mais imediata; segundo o Censo 2022 do IBGE (IBGE, 2022), a população com 60 anos ou mais já representa 15,8% do total do país, com projeções do Ministério da Saúde indicando que, até 2030, o número de idosos deverá ultrapassar o de jovens na faixa de zero a 14 anos (BRASIL, 2022). O envelhecimento demanda serviços e benefícios que garantam uma vida condigna, diferenciando-se do restante da sociedade devido ao caráter degenerativo de certas condições, conforme destacado pela Rede Interagencial de Informações para Saúde (RIPSA) (RISPA, 2009). Estima-se, ainda, que até 80% da população idosa possua ao menos uma condição crônica, projetando um contingente de 50 milhões de pessoas com necessidades permanentes de saúde até 2050. Diante disso, o sistema exige não apenas investimentos vultosos, mas uma readequação em infraestrutura e capital humano, focando na formação geriátrica integrada e no fortalecimento das redes sociais de suporte.

Um exemplo de ação diante de desafios como os mencionados, realizado pelo governo brasileiro, foi o lançamento em 2018 da Estratégia Brasil Amigo da Pessoa Idosa (BRASIL, 2018), cujas iniciativas incluem a promoção de ambientes seguros, adaptação de residências, lazer e medidas de prevenção de quedas. Mais recentemente, políticas como o Programa Envelhecer nos Territórios e o Viva Mais Cidadania (BRASIL, 2024) têm focado no combate ao idadismo e na garantia de direitos nos locais de residência. Nesse contexto, a inovação tecnológica surge como o pilar estratégico para otimizar recursos e garantir a autonomia e inclusão social da população que envelhece. Assim, a Computação desempenha um papel fundamental seja através do letramento digital ou pelo desenvolvimento de tecnologias assistivas e telessaúde.

A Visão Computacional é uma área da Ciência da Computação dedicada à extração e à interpretação de informações a partir de imagens e vídeos, com o objetivo de

representar e compreender o mundo real por meio de dados visuais (SZELISKI, 2022). Entre os problemas centrais dessa área, destaca-se o reconhecimento de ações humanas, no qual modelos computacionais buscam identificar e classificar atividades a partir de sequências visuais (SHUCHANG, 2022). No contexto do cuidado com a saúde de idosos, o reconhecimento de ações baseado em visão computacional permite o monitoramento automatizado de atividades cotidianas de forma não intrusiva, utilizando apenas informações visuais. Essa abordagem contribui para a preservação da autonomia dos indivíduos, ao dispensar o uso de dispositivos vestíveis ou sensores corporais, possibilitando que os idosos realizem suas atividades diárias de maneira natural. A detecção de eventos relevantes, como quedas, comportamentos atípicos ou situações de emergência, pode, assim, ocorrer sem a necessidade de vigilância constante por parte de familiares ou cuidadores.

Esse tipo de aplicação insere-se no escopo dos Sistemas de Ambiente de Vida Assistida (*Ambient Assisted Living* – AAL), cujo objetivo é oferecer suporte à vida independente e segura de pessoas idosas ou com limitações funcionais em seus próprios ambientes (CICIRELLI et al., 2021). Na prática, muitas soluções AAL são concebidas como ecossistemas complexos, frequentemente associados a infraestruturas baseadas em Internet das Coisas (*Internet of Things* – IoT), que envolvem múltiplos sensores, dispositivos conectados e camadas de comunicação. Embora essas abordagens ampliem as possibilidades de monitoramento e automação, elas também introduzem desafios adicionais de custo, implantação e manutenção em ambientes domésticos reais. É nesse contexto que se estabelece a motivação deste trabalho. Diante da crescente demanda por soluções de monitoramento assistido e da complexidade observada em arquiteturas AAL baseadas em IoT, este estudo propõe investigar uma alternativa mais simples e focada, centrada no uso de modelos de reconhecimento de ações guiados por representações de pose humana, considerando o vídeo como única fonte de entrada. Essa abordagem busca reduzir a dependência de sensores adicionais e de infraestruturas especializadas, isolando a contribuição metodológica do reconhecimento de ações e facilitando a implementação prática.

1.1 Definição do problema

A implementação dos modelos de reconhecimento de ações envolve, primariamente, a extração de características das imagens. Essas características, também referidas como *features*, são estruturas capazes de representar uma informação e podem ser obtidas de forma manual ou de forma automatizada com uso de aprendizado de máquina. Existem diferentes formatos de entrada para o treinamento de modelos, como imagens provenientes de câmeras comuns, câmeras de profundidade (RGB-D), câmeras infravermelhas ou térmicas. Há ainda dados obtidos por sensores de movimento e entradas biomédicas como sensores de frequência cardíaca e respiração. Os formatos diferem no grau de informação que fornecem e influenciam na acurácia do modelo, alguns sendo mais invasivos em relação à privacidade do indivíduo monitorado e podem requerer interação direta com o dispositivo receptor da informação.

O problema central abordado por este trabalho é a necessidade de desenvolver sistemas de monitoramento contínuo de idosos que sejam não invasivos e não dependam de sensores físicos, utilizando apenas entradas no formato de imagens e vídeo. Essa abordagem busca garantir o conforto e a privacidade dos idosos, evitando a necessidade de dispositivos vestíveis ou sensores intrusivos. A qualidade e natureza dos dados obtidos refletem diretamente na acurácia e eficiência dos modelos de reconhecimento de ações. Para formatos visuais de entrada, por exemplo, a precisão pode ser afetada por variáveis como iluminação, ângulos de câmera e a presença de obstruções no ambiente. Além das preocupações com a invasividade, o custo computacional e financeiro atrelado ao tipo de informação consumida impacta diretamente a aplicabilidade dos modelos em um cenário real. Sistemas baseados em entradas visuais precisam ser robustos e adaptativos para funcionar de maneira eficaz em diversas condições. Outro desafio é a disponibilidade de conjuntos de dados (*datasets*) adequados para o treinamento e teste dos modelos. Os conjuntos de dados para o problema de reconhecimento de ações de idosos são majoritariamente restritos, muitas vezes produzidos pelos próprios autores dos sistemas e não disponibilizados publicamente. Portanto, é essencial explorar e avaliar modelos avançados de reconhecimento de ações que utilizem apenas entradas visuais, superando as limitações mencionadas para garantir um monitoramento seguro, eficiente e não invasivo dos idosos.

1.2 Objetivos

O objetivo geral deste trabalho é aplicar um modelo de reconhecimento de ações para monitoramento não invasivo que utilize apenas entradas de imagens e vídeo, visando identificar atividades cotidianas e situações de risco de forma precisa e eficiente, sem comprometer o conforto dos usuários. As métricas utilizadas são acurácia geral e acurácia média por classe de ação e custo computacional. As propriedades consideradas são tipo de entrada, método de extração de características e método de classificação. São objetivos específicos:

- Estudar diferentes modelos de reconhecimento de ações baseados em entradas visuais.
- Levantar os principais conjuntos de dados de ações relacionadas ao monitoramento de idosos.
- Aplicar um modelo eficiente de reconhecimento de ações baseado em entradas visuais.
- Avaliar o desempenho do modelo em um conjunto de dados voltado para o monitoramento de idosos.

2 Fundamentação teórica

Este capítulo apresenta os conceitos fundamentais para a compreensão do problema de monitoramento de pessoas, com foco no reconhecimento de ações. A Seção 2.1 aborda o problema de reconhecimento de ações, explorando suas origens, objetivos e relevância. Na Seção 2.2, são discutidos os conceitos de ações e as diversas formas de representá-las computacionalmente, destacando a importância de uma representação eficiente para o sucesso dos modelos. A Seção 2.3 introduz o conceito de redes neurais, detalhando seu funcionamento e a evolução para o aprendizado profundo (*deep learning*), que impulsiona grande parte das inovações no reconhecimento de ações. Por fim, a Seção 2.3.1 foca nas Redes Neurais Convolucionais (CNNs), uma arquitetura amplamente utilizada devido à sua eficiência na extração de características relevantes de imagens e vídeos, aspectos críticos na resolução do problema em questão.

2.1 Reconhecimento de ações humanas

O reconhecimento de ações humanas (RAH) vem sendo um problema tratado especialmente pelos campos de visão computacional e aprendizado de máquina, sendo proveniente do ramo de análise de vídeos. Aggarwal e Ryoo (2011) abordam o reconhecimento da atividade humana como uma tarefa cujo objetivo é analisar automaticamente as atividades em andamento de um vídeo desconhecido. Kong e Fu (2022) especificam que o problema consiste em inferir ações de indivíduos com base em uma ação já realizada, e prever ações com base em execuções incompletas. O estudo desse assunto, fundamentalmente, busca definir o que são ações humanas e formas de representá-las a partir de abstrações computacionais que reproduzem propriedades e atributos providos pelo sentido da visão humana. Esse processo é feito a partir de algoritmos, que, segundo Kong e Fu (2022), devem produzir um rótulo após observar a execução total ou parcial de uma ação humana. As principais aplicações do uso de modelos de reconhecimento de ações no mundo real se encontram em sistemas de vigilância, monitoramento de pacientes, recuperação e

anotação de vídeos.

O conceito de ação é variado entre os diferentes autores desse objeto de pesquisa. Turaga et al. (2008) faz distinção de ação e atividade, a primeira define como movimentos simples executados na ordem de tempo de segundos, e a segunda como "ações coordenadas entre um pequeno número de pessoas". Aggarwal e Ryoo (2011) ainda dividem as atividades humanas em quatro níveis: gestos, ações, interações e atividades em grupo. Os gestos são considerados como as partes atômicas do movimento, como levantar um braço ou uma perna, e as ações como sendo a composição de múltiplos gestos, como caminhar, acenar. As interações e atividades envolvem mais de um indivíduo realizando diversas ações.

2.2 Representação de ações

O reconhecimento de ações é visto como um problema de classificação, no entanto, a tarefa primária é definir como uma ação será representada computacionalmente. Partindo da ideia de compreender ações usando a anatomia humana, as primeiras formas de representação utilizavam de modelos 2D ou 3D para descrever segmentos e juntas correspondentes do corpo humano (WANG; HU; TAM, 2003). A tarefa comum a toda representação é transformar as entradas visuais, dispostas em *pixels*, em vetores de características (TURAGA et al., 2008). Essas abordagens iniciais fazem parte do que é conhecido como *shallow approaches* (abordagens rasas) e se baseiam em características simples extraídas diretamente das imagens ou vídeos, como contornos, e fazendo uso de algoritmos como regressão linear, regressão logística, árvores de decisão, K-Vizinhos mais próximos*(K-nearest Neighbors)*, e máquina de vetores de suporte (SVM) (AGGARWAL; RYOO, 2011). Se tratando de ações, é de interesse capturar informações quanto a movimento nas imagens. Para isso, alguns modelos baseados em movimento foram propostos, como imagem de energia em movimento (*Motion Energy Image* – MEI) e imagem de histórico de movimento (*Motion History Image* – MHI) (BOBICK; DAVIS, 2001), histograma de gradientes orientados (*Histogram of Oriented Gradients* – HOG) (WANG et al., 2011; DALAL; TRIGGS, 2005) e o fluxo óptico (*Optical Flow* – OF) (BEAUCHEMIN; BARRON, 1995).

Essas abordagens também são conhecidas como *hand-crafted features*, traduzido como "características extraídas a mão", e segundo Kong e Fu (2022), o termo implica que modelos baseados nessas técnicas têm seus parâmetros decididos por especialistas, assim exigindo conhecimento aprofundado do domínio. A partir do entendimento dos padrões visuais específicos do contexto pode-se pensar no método de extração de características adequado ao problema. Assim, por consequência, esses modelos também apresentam dificuldade de performar de maneira generalizada. Turaga et al. (2008) menciona a importância da robustez e da invariância ao lidar com informações no formato de vídeo. A robustez diz respeito a um modelo ser eficaz mesmo com algumas variações na entrada quanto a ruídos, angulação e iluminação da cena, e a invariância considera mudanças de posição, rotação e escala. Os métodos baseados nas abordagens anteriores são bastante sensíveis nesses aspectos. Assim, a alternativa que surge e hoje representa o estado da arte para reconhecimento de ações (SHUCHANG, 2022) são as abordagens baseadas em redes neurais profundas ou *deep architectures*, cujos modelos são capazes de automaticamente aprender a identificar características. O fluxo óptico, apesar de estar incluso nas abordagens rasas, contribui significativamente quando usado como entrada em modelos de aprendizado profundo por carregar informações quanto ao aspecto temporal dos vídeos.

2.2.1 Estimativa de pose

No contexto do reconhecimento de ações, a estimativa de pose humana surge como uma forma de representação estrutural do movimento, na qual a dinâmica de uma ação é descrita a partir da configuração espacial e temporal das articulações do corpo. Essa representação abstrai informações de aparência e fundo, concentrando-se na geometria do corpo humano ao longo do tempo. A estimativa de pose consiste, portanto, na identificação das posições das articulações do corpo humano em imagens ou sequências de vídeo. Inicialmente, esse problema foi abordado por meio de modelos baseados em partes deformáveis, como as *Pictorial Structures* (PS) (FISCHLER; ELSCHLAGER, 1973), nos quais o corpo humano é representado como um grafo. Nesses modelos, cada nó corresponde a uma parte do corpo, enquanto as arestas codificam restrições geométricas entre as articulações, sendo utilizadas características manuais, como SIFT ou HoG, para

a descrição visual das partes.

Com o avanço do aprendizado profundo, surgiram modelos com uma formulação denominada holística, como o *DeepPose* (TOSHEV; SZEGEDY, 2014), na qual a estimativa da pose é realizada a partir de uma representação global da imagem. Nessa abordagem, a CNN processa o corpo humano como um todo e prediz simultaneamente as coordenadas (x, y) das articulações, sem impor explicitamente restrições geométricas entre as partes. Essa mudança representou um marco ao substituir *pipelines* baseados em engenharia manual de características por modelos aprendidos.

Apesar de sua relevância histórica, a regressão direta mostrou limitações em termos de precisão espacial. Em resposta, métodos como o de Tompson et al. (2014) passaram a modelar a estimativa de pose como um problema de predição de mapas de calor (*heatmaps*), nos quais cada *pixel* expressa a probabilidade de ocorrência de uma articulação naquela posição. Durante o treinamento, o rótulo de cada articulação é representado por um mapa de calor Gaussiano bidimensional, cuja média coincide com a posição real da junta e cuja variância é mantida pequena para concentrar a distribuição.

Formalmente, para uma articulação localizada na coordenada (x_g, y_g), o valor do *pixel* (i, j) no mapa de calor alvo T é dado por:

$$T(i,j) = \exp\left(-\frac{(i-x_g)^2 + (j-y_g)^2}{2\sigma^2}\right), \tag{2.1}$$

em que σ determina o grau de espalhamento da distribuição de probabilidade ao redor da articulação. O processo de treinamento consiste em minimizar o Erro Quadrático Médio entre os mapas de calor estimados pela rede e os mapas Gaussianos de referência.

2.3 Aprendizado profundo

O aprendizado profundo ou *deep learning* se refere a modelos que utilizam redes neurais profundas para resolução de problemas de classificação e reconhecimento de padrões. A ideia por trás das redes neurais foi introduzida por Rosenblatt (1958) com a proposta de reproduzir o aparato visual humano que, basicamente, consiste de neurônios interconectados no chamado córtex visual. Essa proposta foi o *perceptron*, considerado o modelo de

um neurônio artificial e foi originalmente proposto como um modelo probabilístico para ilustrar como o cérebro (ou uma máquina) poderia armazenar e organizar informações. A estrutura do *perceptron* é ilustrada na Figura 2.1. O modelo é composto por unidades sensoriais, de associação e de resposta. As unidades sensoriais recebem os valores de entrada, as unidades de associação combinam essas informações por meio de pesos, e a unidade de resposta aplica uma função de decisão para produzir a saída final.

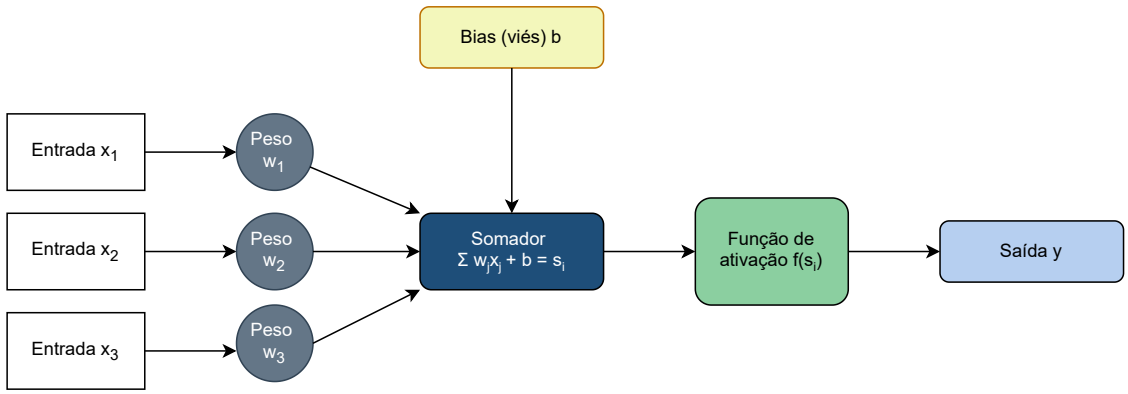


Figura 2.1: Esquema de um *perceptron*, mostrando as operações que ele realiza. Figura elaborada pela autora.

Especificamente, o que ocorre é uma combinação linear formada pelos pesos que multiplicam cada variável de entrada, somados a um viés (*bias*). Esse processo também pode ser interpretado como uma regressão logística, cujo objetivo é criar um limite linear entre duas classes linearmente separáveis. A equação da soma ponderada é dada por:

$$s_i = \sum_{j=1}^{n} w_j x_j + b, \tag{2.2}$$

em que s_i é a saída da soma linear ponderada, w_j são os pesos, x_j os valores de entrada e b o viés.

Em sequência à soma ponderada, aplica-se uma função de ativação não linear, responsável por definir o comportamento de decisão do *perceptron*. Essa função determina a classe de saída com base no valor de s_i. Um exemplo simples é a *função limiar* (ou *função degrau*), que atribui uma saída de 1 ou -1 de acordo com um valor de limiar τ:

$$y = \begin{cases} 1, & \text{se } s_i \geq \tau \\ -1, & \text{se } s_i < \tau, \end{cases} \tag{2.3}$$

onde y representa a classe predita.

De forma geral, a operação do *perceptron* pode ser expressa como:

$$y_i = h(s_i), \tag{2.4}$$

em que $h(\cdot)$ é a função de ativação — neste caso, a função limiar — aplicada à soma ponderada s_i, resultando na saída final y_i.

As redes neurais profundas são implementações dos chamados neurônios artificiais interconectados e dispostos em múltiplas camadas, onde a saída dos neurônios de uma camada alimenta os das próximas e assim por diante (*feedfoward*). O que difere a implementação desses neurônios do próprio modelo *perceptron* é a função de ativação que lida melhor com intervalos contínuos, variando de forma gradual, sem saltos abruptos, em resposta a pequenas mudanças na entrada. Isso contrasta com a função de limiar rígido do *perceptron*, que tem uma transição abrupta entre dois estados, o que pode causar instabilidade em sistemas interconectados. O termo aprendizado profundo vem da profundidade que uma rede alcança pelo seu número de camadas. O que faz com que os modelos tenham a capacidade de "aprender" é a aplicação de derivadas que ajustam os diferentes parâmetros ou pesos da rede a partir do valor obtido na saída, analisando um erro entre o valor esperado e obtido de uma predição. Esse processo é chamado de retropropagação (*backpropagation*) (SZELISKI, 2022) e consiste no treinamento de uma rede.

Um modelo pode ser ainda pré-treinado em uma tarefa (geralmente usando grande quantidade de dados) e reaproveitado para outra tarefa relacionada (GOODFELLOW; BENGIO Y.AND COURVILLE, 2016). A ideia é que os primeiros níveis de uma rede neural capturam características gerais dos dados, e esses conhecimentos podem ser úteis para outras tarefas. Esse procedimento é chamado de transferência de aprendizado (*transfer learning*).

2.3.1 Redes Neurais Convolucionais

As redes neurais convolucionais (*Convolutional Neural Networks* – CNN) são uma implementação de redes neurais multicamadas focadas especialmente na tarefa de processamento de imagens. Essa arquitetura foi amplamente aprimorada e popularizada no trabalho de Krizhevsky, Sutskever e Hinton (2012). Sua principal propriedade é o uso de filtros para capturar padrões relevantes em regiões locais da imagem. Essas regiões específicas são também referidas como campos receptivos. O conjunto de pesos, dispostos na forma do campo receptivo, é chamado de *kernel* ou filtro e pode ter diferentes tamanhos (GONZALEZ; WOODS, 2018).

Os filtros são movidos por toda a imagem realizando a operação de convolução, a qual calcula uma soma de produtos entre os valores dos *pixels* e o conjunto de pesos do *kernel*, os quais serão os parâmetros a serem aprendidos pela rede. Diferentes filtros são aprendidos durante o treinamento para detectar diferentes tipos de padrões na imagem. A saída da operação de convolução gera os chamados *feature maps* ou mapas de características. A operação de convolução é ilustrada na Figura 2.2.

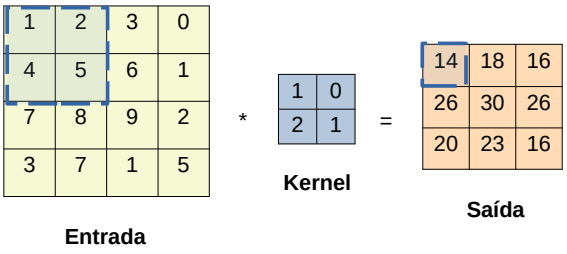


Figura 2.2: Processo de convolução com um *kernel* de tamanho 2×2.

De acordo com Szeliski (2022) a composição de múltiplas camadas nas redes convolucionais busca construir características locais e combiná-las de diferentes maneiras para produzir características mais discriminativas e semanticamente significativas. Assim se cria a ideia de hierarquia de características que variam de baixo nível, como bordas e texturas, a alto nível, que identificam objetos inteiros. Entre as camadas também ocorre o processo de *pooling* (agrupamento) que reduz a dimensão da imagem, ajudando a diminuir a complexidade computacional, mantendo os padrões mais importantes ao eliminar detalhes menos relevantes.

As etapas que compõem a arquitetura das CNNs são:

1. **Convolução e geração dos mapas de características**: aplicação de filtros convolucionais sobre a imagem de entrada para extrair padrões locais.

2. **Aplicação da função de ativação**: aplicação de uma função não linear sobre cada entrada (*pixel*) do mapa de características.

3. **Redução de dimensionamento (*pooling*)**: redução da dimensionalidade espacial dos mapas de características, preservando as informações mais relevantes.

4. **Transformação em vetor unidimensional (*flattening*)**: conversão dos mapas de características em um vetor para entrada nas camadas seguintes.

5. **Camada totalmente conectada (*Fully Connected Layer*)**: responsável pela classificação final, conectando todas as entradas a todas as saídas possíveis. Em problemas de classificação, as saídas representam as probabilidades de a imagem pertencer a cada uma das classes consideradas.

A Figura 2.3 ilustra a arquitetura de uma CNN.

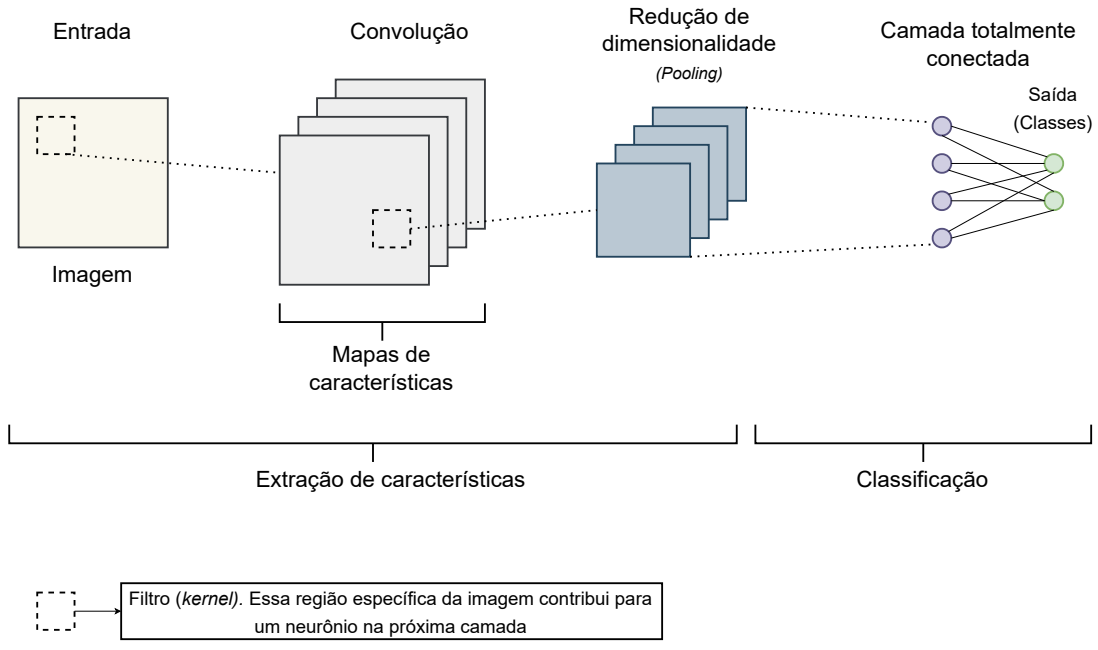


Figura 2.3: Arquitetura simplificada de uma Rede Neural Convolucional (CNN). Adaptado de Phung e Rhee (2019).

3 Trabalhos relacionados

Este capítulo apresenta abordagens de monitoramento de idosos utilizando diferentes tecnologias. As Seções 3.1 a 3.2 dão foco àquelas com aplicação de técnicas de visão computacional, aprendizado profundo sem uso de sensores. A Seção 3.1 apresenta um sistema de detecção de quedas usando redes neurais convolucionais (CNNs) e imagens de fluxo óptico, destacando a robustez da técnica para diferentes cenários e condições de iluminação. Outro trabalho descreve a combinação de modelos baseados em CNNs que utilizam fusão de entradas em RGB e fluxo óptico, além de estimativa de pose para monitorar ações cotidianas e de alerta em idosos. Já a seção 3.2 introduz um sistema de reconhecimento de ações usando câmeras de profundidade estéreo, onde são extraídas características manuais para identificar comportamentos como quedas e transições posturais. A Seção 3.3 traz exemplos de monitoramentos com uso de sensores vestíveis e não vestíveis e apresenta os sistemas AAL inteligentes que fazem uso de IoT.

3.1 Monitoramento com fluxo óptico e CNNs

Núñez-Marcos, Azkune e Arganda-Carreras (2022) têm como objetivo desenvolver um sistema baseado em visão computacional para detectar quedas de idosos, utilizando CNNs. A motivação, assim como a do presente trabalho, é criar um sistema mais confortável para idosos sem uso de sensores vestíveis. O sistema usa imagens de fluxo óptico para identificar movimentos de queda, tornando-se independente de características visuais do ambiente, como cor ou iluminação. O trabalho destaca como as propriedades desse tipo de representação contribuem para reconhecimentos baseados em entradas visuais. Com a utilização das técnicas citadas também há o objetivo de tornar o sistema generalizado para outros tipos de cenário.

O algoritmo de fluxo óptico foi usado para descrever os vetores de deslocamento entre dois quadros (*frames*) das entradas RGB. No entanto, os autores levantam que imagens de fluxo óptico registram um intervalo muito curto entre os quadros para detectar

uma queda. Para contornar esse detalhe foi aplicada a ideia de empilhar um conjunto dos quadros. Assim, a rede também pode aprender recursos relacionados a tempo mais longo. A representação de fluxo óptico foi usada como entrada de uma rede neural totalmente conectada (*Fully Connected Neural Network* - FCNN), que funciona como o classificador e emite um sinal de "queda"ou "sem queda".

O modelo aplicado implementou uma versão modificada da arquitetura da CNN VGG-16 e foi pré-treinada com o conjunto de dados ImageNet. Em seguida a rede foi re-treinada utilizando conjuntos de dados para reconhecimentos de ações, como o UCF101, para que a rede pudesse aprender a interpretar movimentos humanos. Posteriormente o modelo foi treinado com três conjuntos de dados públicos específicos para a tarefa de detecção de queda, os conjuntos foram o *UR Fall Dataset* (URFD) que contém 30 vídeos de quedas e 40 de atividades diárias; o *Multiple Cameras Fall Dataset* (Multicam) que inclui 24 performances gravadas de várias perspectivas; e o *Fall Detection Dataset* (FDD), que contém quedas simuladas em cenários diversos.

Os resultados foram comparados entre os três conjuntos de dados específicos e apresentados em termos de sensibilidade e especificidade. Sensibilidade mede a capacidade de detectar corretamente quedas (verdadeiros positivos), enquanto especificidade avalia a habilidade de identificar corretamente eventos que não são quedas (verdadeiros negativos). O conjunto URFD, atingiu 100% de sensibilidade e 94,86% de especificidade. O Multicam, teve sensibilidade de 98,07% e especificidade de 96,20%. O FDD, alcançou 93,47% de sensibilidade e 97,23% de especificidade. O sistema também apresentou robustez ao ser testado em condições diferentes de iluminação e em cenários diversos. No trabalho os autores ainda abordam a questão da limitação quanto ao conjunto de dados e à quantidade de amostras para treinamento dos modelos no contexto apresentado e sugerem a técnica de transferência de aprendizado (*transfer learning*) como uma solução adequada.

O trabalho de Buzzelli, Albé e Ciocca (2020) propõe um sistema para monitoramento de idosos baseado em vídeos a partir da análise de dois modelos de reconhecimento de ações implementados com CNNs. O primeiro modelo é o I3D (*Inflated 3D ConvNet*) que utiliza a fusão de entradas nos formatos RGB e fluxo óptico, o outro é o DeepHar focado em estimativa de pose e utiliza apenas formato RGB. A análise foi feita considerando

o desempenho dos modelos para detecção de determinadas categorias de ações. A taxonomia desenvolvida pelos autores divide as ações em 3 categorias principais: Alerta, Diário (vida diária) e *Status*. As ações de alerta são aquelas que podem representar situações de risco e incluem tocar a cabeça, tocar as costas, vomitar, tossir/espirrar, se exercitar, e cair. A categoria Diário diz respeito a ações cotidianas comuns como beber, comer, ler, usar o telefone. A classe *Status* representa as possíveis poses em que um indivíduo pode se encontrar e as ações que a compõem são: sentado, em pé, deitado, no chão e caminhando. A escolha do modelo I3D foi pensada na sua capacidade de discriminar entre diferenças sutis em classes muito semelhantes, como as classes de alerta e vida diária. O DeepHar devido ao seu bom desempenho em classificação perante uma representação explícita de esqueleto humano, serve melhor à classe de *Status* que é baseada em poses.

As classes de cada um desses grupos não são mutualmente exclusivas entre os grupos, assim, por exemplo, uma pessoa pode se encontrar sentada enquanto come ou bebe, e o sistema classificará as ações de forma independente. Esse rastreio paralelo que definiu a arquitetura do modelo final utilizando uma estrutura que combina três modelos em uma rede neural multiobjetivo exclusiva, que realiza um processamento comum inicial e depois se ramifica em três caminhos independentes. Algumas técnicas aplicadas nas etapas inciais de processamento foram uso de pré-treinamento de redes, ajuste fino e uso de detector de objetos. O sistema atingiu 97% de acurácia na inferência de poses básicas, 83% em situações de alerta e 71% em ações da vida diária com o uso do I3D, o modelo que melhor performou. A classe de ações de vida diária teve acurácia prejudicada por confusões entre ações como beber, comer e usar o telefone, que diferem apenas para o objeto segurado pelo indivíduo.

Esse trabalho também busca superar as limitações em relação a conjuntos de dados contribuindo com a construção de um *dataset* agregado chamado ALMOND (*Assisted Living MONitoring Dataset*), que reúne cinco conjuntos existentes usados para o problema de reconhecimento de ações, porém filtrando ações que são comumente realizadas em ambientes fechados e domésticos. A definição das classes de ações citadas previamente é usada na construção desse conjunto de dados. Outra contribuição é uma metodologia geral para estimar a distância máxima permitida entre a câmera e o objeto monitorado.

3.2 Monitoramento com *hand-crafted features*

O artigo de Zin et al. (2021) apresenta um sistema de reconhecimento de ações em tempo real voltado para o monitoramento de idosos em um centro de cuidados, utilizando câmeras de profundidade estéreo. As classes de ação reconhecidas são "Fora do quarto", "Transição", "Sentado em cadeira de rodas", "Em pé", "Sentado na cama", "Deitado na cama", "Recebendo assistência" e "Queda". Essas ações foram pensadas considerando o que pode ser realizado na ausência de enfermeiros nos quartos.

O sistema localiza pessoas extraindo diferentes regiões de interesse de mapas de disparidade UV (coordenadas U e V em mapeamento de texturas) provenientes de quadros de imagens de profundidade. Para extração de características foi utilizada a fusão das representações Aparência de Movimento de Profundidade (*Depth Motion Appearance* – DMA), que captura a forma e aparência volumétrica do movimento, e Histórico de Movimento de Profundidade (*Depth Motion History* – DMH), que rastreia o histórico temporal do movimento. Ambas as representações são descritas usando o histograma de gradientes orientados (HOG). O sistema também incorporou características baseadas em distância, medindo a distância entre o centro de massa da pessoa e o plano do chão para identificar ações como quedas ou assistência.

O reconhecimento de ações foi realizado utilizando o método de arredondamento automático (*automatic rounding method*) (GUO et al., 2010), o qual divide automaticamente sequências de quadros longos em várias sequências curtas. A classificação é então feita por um SVM a partir dos descritores de características. Um detalhe importante levantado sobre a identificação da ação "Recebendo assistência"é a necessidade de considerar a altura dos pacientes e enfermeiros. O enfermeiro que presta assistência é geralmente mais alto que o idoso, e a altura normal de um idoso também parece maior do que aquela quando o mesmo cai no chão. Assim, dois valores de limiar devem ser definidos para a classificação: se a altura da pessoa for maior que o "limiar de assistência", a ação é reconhecida como "Recebendo assistência", se for menor que o "limiar de queda", a ação é reconhecida como "Queda".

Os dados foram coletados em três quartos de um centro de cuidados para idosos no Japão e o número total de dias de gravação para cada sala foi 9, 6 e 10 dias. Apenas

imagens de profundidade foram registradas, as imagens RGB foram omitidas para preservar a privacidade dos residentes. Para cada quarto o número de sequências de ações registradas foi de 14, 10 e 11 respectivamente. As sequências têm duração entre 1 e 13 horas. O resultado geral foi obtido a partir da média de acurácias obtidas na classificação de determinadas sequências. As acurácias variam de acordo com o uso de filtros de mediana, sem o filtro o valor obtido foi de 90.6% e com o filtro a acurácia chegou a 98.3%. Ainda assim, ocorreram classificações incorretas, como as ações "Deitado na cama" que foram detectadas como "Sentado na cama". O número dessas detecções falsas foi reduzido pela aplicação de um filtro mediano. O tempo de reconhecimento alcançado das ações foi de 5 segundos também por causa da aplicação desse filtro. Os autores destacam a importância do sistema para além do monitoramento de saúde e situações de risco. Com o registro de históricos de ações e comportamentos dos pacientes é possível uma análise automatizada das gravações que pode garantir mais segurança aos residentes, prevenindo dificuldades e permitindo diagnóstico e tratamento oportunos de doenças.

Ainda explorando técnicas manuais de visão computacional, Gaikwad et al. (2023) propõem um sistema de monitoramento de idosos treinado a partir de um conjunto de dados feito de anotações de juntas de esqueleto. A motivação para um conjunto de dados não composto por imagens é que, de acordo com os autores, modelos com esse tipo de entrada demandam muito tempo de treino e também alto consumo de memória. O sistema utiliza o *framework* de reconhecimento de pose *BlazePose* baseado em um *k-NN* (algoritmo de vizinho mais próximo). O *framework* extrai pontos-chave (*landmarks*) de esqueleto dos quadros dos vídeos. A partir das *landmarks*, as características extraídas são: o ângulo das articulações chave (*key-joint angles*), calculado a partir das coordenadas de três pontos-chave de esqueleto utilizando uma fórmula trigonométrica; a distância euclidiana entre as articulações chave (*Euclidean distance*), determinada entre dois desses pontos, com base em suas coordenadas tridimensionais; e a inclinação entre as articulações chave (*slope*) obtida através das coordenadas de dois pontos-chave de esqueleto. São essas características que compõem o conjunto de dados numérico que possui 780.000 valores de *features* calculados de 20.000 imagens.

Esse sistema também é projetado para ambientes fechados e as ações reconheci-

das são “Sentado”, “Em pé”, “Deitado”, “Andando”e “Caindo”. A coleta de dados no estudo foi realizada com quatro participantes idosos (homens), com idades entre 60 e 65 anos. Os participantes tinham diferentes características físicas, como altura e peso, para garantir diversidade no conjunto de dados. Foram feitas gravações dos participantes realizando atividades normais como caminhar, sentar, deitar e incluiu quedas. Posteriormente os vídeos criados foram convertidos em imagens estáticas. Para cada atividade, foram gravadas 1.000 imagens por participante.

Para a classificação são usados três algoritmos: árvore de decisão (*Decision Tree* - DT), floresta aleatória (*Random Rorest* - RF), SVM e o método *ensemble*. O método *ensemble* consiste na combinação de previsões dos demais modelos visando melhorar a precisão e a robustez. O sistema foi testado realizando 200 testes para cada método. O método de *ensemble* se destacou, atingindo uma precisão de 99%. O *Random Forest* também apresentou resultados notáveis, com uma precisão de 98%, enquanto o SVM e o *Decision Tree* tiveram precisões em torno de 95% a 96%. O trabalho também destaca a importância da aplicação do sistema em cenários reais, sendo feita uma análise subjetiva de custo, potência e compatibilidade dos idosos com o sistema implementado. Concluiu-se que o sistema funciona com custo reduzido de energia, porém com tempo de execução operacional estendido, a instalação é fácil e usa componentes sem fio, assim garantindo conforto aos usuários. Há compatibilidade com configurações de casas inteligentes.

3.3 Monitoramento com uso de sensores

Em (OUDAH; AL-NAJI; CHAHL, 2020) é proposto um sistema de reconhecimento de gestos manuais para cuidados de saúde de idosos, em especial indivíduos surdos e mudos. O modelo é baseado em CNNs e junto da entrada em formato RGB utiliza imagens de profundidade (*depth*) provenientes dos sensores *Microsoft Kinetic*. A escolha foi feita por esse tipo de sensor ser considerado acessível e também confiável para monitoramento a longo prazo. Além disso não exige contato direto com o a pessoa monitorada, um objetivo que se alinha com a proposta do presente trabalho. As 5 classes de gestos reconhecidos são “Água”, “Refeição”, “Banheiro”, “Ajuda”e “Remédio”. Assim que os gestos são identificados, os cuidadores dos idosos são notificados por mensagem de texto

via dispositivo móvel.

Uma CNN foi utilizada para a tarefa de extração de características, implementando a arquitetura ResNet-50 (DENG et al., 2009) pré-treinada. Para classificação dos gestos foi utilizado o SVM. Além disso, um *hardware* foi desenvolvido para fazer a comunicação entre o sistema e os dispositivos móveis. O *hardware* inclui um sensor Kinect V2, um microcontrolador Arduino Nano e um módulo GSM Sim800l, que envia mensagens para cuidadores em tempo real. O sistema funciona em um ambiente fechado e enfrenta limitações, como a precisão da captura a distâncias de até 4,5 metros.

O sistema foi testado com três participantes idosos e um adulto em ambientes domésticos, obtendo uma taxa de reconhecimento de gestos de 96,62%. Apesar do bom valor na acurácia, os autores levantam desafios a serem superados, como a distância de captura, e a ocorrência de sobreposição de gestos com o corpo, o que dificulta sua identificação.

Hussain et al. (2015) apresentam uma plataforma para cuidados de saúde e emergências em cidades inteligentes. A plataforma combina sensores IoT com sistemas de alerta para monitoramento contínuo. O objetivo é usar as capacidades da IoT para criar um sistema inteligente que permita monitoramento e interação em tempo real, voltado para a saúde personalizada de idosos e pessoas com deficiência em suas casas. O sistema consiste em uma parte remota, que permite armazenar e distribuir os dados para provedores de serviços, e uma parte local que lida com a coleta de informações dos sensores conectados a um paciente. O sistema inclui sensores portáteis e dispositivos inteligentes conectados à IoT para medir parâmetros vitais, como batimentos cardíacos, temperatura, oxigenação, entre outros. Os sensores biomédicos monitoram vários parâmetros fisiológicos como ECG, temperatura corporal, frequência cardíaca, e postura do corpo (como quedas). Esses sensores estão conectados a dispositivos móveis via redes sem fio, como *bluetooth* e enviam dados continuamente para o sistema. O sistema detecta ações como quedas, anomalias fisiológicas (ex.: arritmia cardíaca, febre alta), inatividade prolongada e eventos manuais de emergência acionados pelo próprio usuário. Sensores monitoram continuamente parâmetros vitais e de movimento, acionando alarmes automáticos em caso de desvios críticos ou eventos de emergência. Essas detecções geram alertas que

são enviados a cuidadores ou serviços médicos para intervenção imediata. Um ponto levantado pelos autores é o desafio dos sistemas de saúde centrados no paciente para integrar informações recentes e históricas dos pacientes em sistemas de saúde pessoais, transformando esses dados em suporte para a tomada de decisões. Além disso descrevem as necessidades desse tipo do sistema como sendo a coleta de dados de fontes variadas, armazenamento de forma uniforme em uma plataforma de compartilhamento, e implementação de mecanismos para análise e recuperação dos dados.

O trabalho de Alemán et al. (2016) se trata de um modelo de fusão de dados aplicado a um sistema com foco no monitoramento de idosos em ambientes externos. O objetivo é utilizar sensores de *smartphones* e outros dispositivos para rastrear e detectar possíveis situações de risco, como quedas ou desvios de rotas, notificando cuidadores em casos de emergência. O sistema utiliza o modelo de fusão de dados JDL (*Joint Directors of Laboratories*) para integrar informações de sensores, como acelerômetros, GPS e sensores de temperatura. O sistema foi implementado através de dois componentes principais: a aplicação Android "CareofMe"e o sistema web "SafeRoute". Além disso, o sistema inclui uma rede de segurança formada por trabalhadores locais que atuam como sensores e atuadores para ajudar idosos em caso de emergência. A metodologia do trabalho envolve a coleta de dados de sensores embutidos em *smartphones* para detectar ações como quedas ou desvios de rotas. A fusão de dados é usada para processar as informações e gerar diagnósticos sobre o estado do idoso. Experimentos foram realizados simulando quedas e desvios em rotas predefinidas, com três idosos testando o sistema. Os resultados mostram que o sistema baseado em *smartphones* oferece maior precisão na localização e monitoramento dos usuários em comparação com sensores alternativos, como sensores do Arduino. Além disso, o sistema reduziu o tempo de resposta em situações de emergência, ao incluir os trabalhadores locais como parte da rede de suporte. Os autores destacam a necessidade de melhorar a calibração dos sensores para aumentar ainda mais a precisão. Além disso, sugerem o desenvolvimento de novas funcionalidades inteligentes, como respostas automáticas em caso de emergência.

Abdelgawad, Yelamarthi e Khattab (2017) abordam o desenvolvimento de um sistema de monitoramento de saúde baseado em IoT voltado para oferecer assistência

ativa e suporte a idosos e pessoas com limitações físicas. O objetivo principal do sistema é melhorar a qualidade de vida desses usuários, permitindo que seus sinais vitais e condições ambientais sejam monitorados em tempo real, com os dados sendo processados na nuvem. Esse sistema possibilita a tomada de ações preventivas e imediatas, como resposta a quedas ou a detecção de problemas cardíacos, garantindo assim a segurança e o bem-estar dos usuários. São utilizados sensores leves e vestíveis, como oximetria de pulso, ECG, sensores de fluxo nasal/oral, temperatura, além de sensores de luz e detecção de quedas. Esses sensores monitoram sinais vitais, como o nível de oxigênio no sangue, a frequência cardíaca e a temperatura corporal, além de detectar quedas bruscas e condições de iluminação inadequadas. A metodologia envolveu experimentos em cenários controlados, simulando situações como quedas e mudanças nas condições de iluminação, além de rastreamento da localização interna dos usuários. Os resultados mostraram que o sistema foi eficaz na detecção de quedas, na precisão do rastreamento de localização e no monitoramento de variações de luz, demonstrando sua viabilidade para uso em assistência à saúde. Os autores ressaltam que, embora o sistema tenha se mostrado eficiente e de baixo custo, ainda há a necessidade de aprimorar a segurança no acesso aos dados.

3.4 Considerações

Os trabalhos relacionados apresentados neste capítulo de forma geral exemplificam a aplicação de modelos de reconhecimento de ações para monitoramento de idosos em diferentes cenários e a partir de diferentes métodos. Um ponto comum em todos os trabalhos é a preocupação com o conforto e independência dos idosos como pilares importantes na garantia de qualidade de vida do grupo a partir do uso desses sistemas. As aplicações com modelos baseados em vídeo e tecnologias de visão computacional destacam maior facilidade no acesso e construção quanto a custo e desenvolvimento, apesar de ainda apresentarem limitações. As implementações baseadas em redes neurais convolucionais evidenciam o potencial dos modelos de aprendizado profundo para o reconhecimento de ações, especialmente quando combinadas com representações de movimento, como o fluxo óptico. Os trabalhos com uso de sensores e tecnologias inteligentes apresentam sistemas mais robustos e completos em níveis de informação para monitoramento de idosos e com

métodos alternativos de rastreamento de ações e atividades. No entanto demonstram maior complexidade e custo na implementação.

A seguir foi feita uma análise comparativa dos trabalhos que utilizam de técnicas de visão computacional a fim de padronizar as métricas a serem comparadas e também para que a análise esteja alinhada com os objetivos deste trabalho. A análise foi dividida em duas tabelas apenas por questões de formatação e melhor visualização dos dados. A Tabela 3.1 traz comparações acerca dos objetivos e ações reconhecidas. A Tabela 3.2 traz informações acerca dos métodos de extração de características, métodos de classificação, métricas de avaliação e técnicas utilizadas nos treinamentos dos modelos.

Tabela 3.1: Comparação de trabalhos por objetivo e ações reconhecidas entre sistemas de monitoramento de ações para idosos.

Trabalho	**Objetivo**	**Ações Reconhecidas**
(NúñEZ-MARCOS; AZKUNE; ARGANDA-CARRERAS, 2022)	Detecção de quedas de idosos em tempo real usando visão computacional	Quedas
(ZIN et al., 2021)	Monitoramento em tempo real de idosos em centros de cuidados	Fora do quarto, Transição, Sentado em cadeira de rodas, Em pé, Sentado na cama, Deitado na cama, Recebendo assistência e Queda
(OUDAH; AL-NAJI; CHAHL, 2020)	Reconhecimento de gestos de idosos para comunicação de necessidades básicas	Gestos manuais (Água, Refeição, Banheiro, Ajuda, Remédio)
(GAIKWAD et al., 2023)	Reconhecimento de atividades e monitoramento residencial de idosos	Sentar, Andar, Ficar de Pé, Deitar, Cair
(BUZZELLI; ALBé; CIOCCA, 2020)	Monitoramento de idosos para independência e emergências (queda)	**Status**: Sentado, em pé, caminhando, deitado, no chão, **Alerta**: tocando a cabeça, tocando as costas, tocando o tronco, tocando o pescoço, vomitando, tossindo/espirrando, acenando com as mãos, fazendo exercícios, caindo, rejeitando, **Vida diária**: bebendo, comendo, lendo, usando o telefone, vestindo/despindo, usando o laptop, rejeitando

Tabela 3.2: Comparação de extração de características, classificação, métricas e técnicas de treinamento.

Trabalho	**Ex. de Características**	**Classificação**	**Métricas**	**Técnicas de Treinamento**
(NúñEZ-MARCOS; AZKUNE; ARGANDA-CARRERAS, 2022)	Imagens de fluxo óptico (TVL-1)	CNN (VGG-16) + FC-NN	Sens.: 100% (URFD), 98.07% (Multicam) — Espec.: 94.86% (URFD), 96.20% (Multicam)	*Transfer learning* (ImageNet, UCF101) + *Fine-tuning*
(ZIN et al., 2021)	DMA + DMH	SVM	Acurácia: 90% – 98%	Treinamento direto
(OUDAH; AL-NAJI; CHAHL, 2020)	CNN (ResNet-50)	SVM	Acurácia: 96.62%	CNN pré-treinada (ResNet-50)
(GAIKWAD et al., 2023)	BlazePose + Ângulos, Distâncias e Inclinações	*Ensemble* (SVM, DT, RF)	Sensibilidade: 99%, Especificidade: 97%	Ensemble (SVM, DT, RF)
(BUZZELLI; ALBé; CIOCCA, 2020)	Faster R-CNN (detecção) + I3D e DeepHAR (ação)	I3D + DeepHAR	Acurácia: 97% (básico), 83% (alertas), 71% (vida diária)	*Transfer learning* (Kinetics-400 e NTU) + *Fine-tuning* (ALMOND dataset)

4 Conjunto de dados

Este capítulo apresenta o levantamento dos principais conjuntos de dados relacionados ao reconhecimento de atividades humanas em ambientes domésticos, com foco no monitoramento de idosos, bem como a justificativa para a escolha do conjunto de dados utilizado neste trabalho. Inicialmente, é realizada uma análise comparativa entre diferentes bases disponíveis na literatura. Em seguida, o conjunto selecionado é descrito em detalhes, destacando suas características, desafios e adequação aos objetivos da pesquisa.

4.1 Levantamento dos conjuntos de dados

Diversos conjuntos de dados têm sido propostos na literatura para o problema de reconhecimento de atividades humanas e monitoramento de idosos, conforme resumido na Tabela 4.1. No entanto, a maioria apresenta limitações que comprometem sua adequação a cenários domésticos reais ou aos requisitos demográficos deste trabalho.

O conjunto IXMAS (WEINLAND; RONFARD; BOYER, 2006), por exemplo, restringe-se a gestos simples capturados em baixa resolução, o que limita a complexidade das ações analisáveis. Os conjuntos UWA3D II (RAHMANI et al., 2013) e N-UCLA (KOPPULA; GUPTA; SAXENA, 2024) oferecem maior diversidade de ações e múltiplas visões, porém não incluem participantes idosos, falhando em atender ao principal critério demográfico da pesquisa. De forma semelhante, o NTU RGB+D (SHAHROUDY et al., 2016), apesar de sua abrangência e ampla adoção, é composto majoritariamente por adultos jovens, não representando adequadamente os padrões de movimento da população idosa.

O MSR Daily Activity 3D (WANG et al., 2012) apresenta ações relacionadas a atividades cotidianas, porém é capturado em um ambiente altamente controlado, com ângulo de câmera fixo, o que reduz significativamente sua capacidade de generalização para residências reais. O Fall Dataset (PLANINC; KAMPEL, 2012), embora relevante para aplicações de segurança geriátrica, concentra-se exclusivamente em quedas e posturas

estáticas, além de ser composto apenas por imagens individuais, e não por sequências de vídeo, tornando-o inadequado para o monitoramento abrangente de atividades de vida diária (AVDs).

Diante dessas limitações, destaca-se o conjunto Toyota Smarthome Trimmed (TST) (DAS et al., 2019), que se diferencia por contemplar idosos realizando atividades espontâneas de vida diária em ambientes domésticos realistas, atendendo simultaneamente aos critérios demográficos, ambientais e de formato de dados exigidos por esta pesquisa. Além do TST, o projeto Toyota Smarthome disponibiliza o conjunto Toyota Smarthome Untrimmed (TSU) (DAI et al., 2022). No entanto, o TSU é composto por vídeos longos e não segmentados, o que impõe elevado custo computacional e maior complexidade no pré-processamento, especialmente em abordagens que requerem segmentação temporal precisa das atividades. Por esse motivo, o TSU não foi adotado neste trabalho.

Tabela 4.1: Comparativo dos conjuntos de dados levantados para reconhecimento de ações de idosos. "N/I" significa "Não informado".

Conjunto	**Nº de Ações**	**Nº de Amostras**	**Nº de Sujeitos**	**Resolução**	**Nº de Câmeras**	**Ambientes**	**Faixa etária dos sujeitos**
IXMAS	11	1148	11	60×48	5	Único	N/I
UWA3D II	30	1075	10	640×480	4	Único	N/I
N-UCLA	10	1494	10	N/I	3	Variados	N/I
NTU RGB+D	60	56880	N/I	1920×1080	3	Variados	N/I
MSR Daily Activity 3D	16	320	N/I	N/I	1 (fixa)	Único	N/I
Fall Dataset	5	21499	5	320×240	1 (fixa)	Único	19 a 50 anos
Toyota Smarthome Trimmed	31	16115	18	640×480	7	3	60 a 80 anos
Toyota Smarthome Untrimmed	51	536	18	640×480	7	3	60 a 80 anos

4.2 Toyota Smarthome Trimmed

O conjunto de dados Toyota Smarthome Trimmed (TST) (DAS et al., 2019) foi selecionado para o presente trabalho por abordar uma lacuna crítica na área de reconhecimento de atividades humanas: a escassez de bases que representem de forma realista as atividades de vida diária realizadas por pessoas idosas em ambientes domésticos.

A composição demográfica dos conjuntos de dados tradicionalmente utilizados

constitui uma limitação relevante, especialmente em abordagens baseadas exclusivamente em informações de pose. A presença — ou ausência — de participantes idosos influencia diretamente a validação, a capacidade de generalização e a aplicabilidade de sistemas de reconhecimento de atividades humanas voltados a esse público. A literatura aponta que grande parte das pesquisas concentra-se no reconhecimento genérico de atividades, resultando em uma lacuna no que se refere a dados especificamente direcionados a idosos (DAI et al., 2022). Essa necessidade é reforçada pelo fato de que características de movimento, como amplitude, ritmo e variabilidade, diferem significativamente entre idosos e populações mais jovens, impactando diretamente as coordenadas de pose utilizadas como entrada dos modelos (ZHAI et al., 2023). Consequentemente, a ausência de validação em dados representativos desse público é frequentemente apontada como uma limitação metodológica (HAYAT et al., 2022; DAI et al., 2022).

O TST é composto por 16.115 vídeos curtos de pessoas idosas realizando 31 atividades espontâneas em ambientes domésticos controlados. As gravações ocorrem em três ambientes distintos — sala de estar, cozinha e sala de jantar — com 7 câmeras em diferentes pontos do ambiente. Os vídeos possuem resolução de 640×480 *pixels*, taxa de amostragem temporal de 20 FPS (*frames per second*) e estão disponíveis em três modalidades: RGB, profundidade e juntas de esqueleto (com coordenadas em 3 dimensões).

A duração dos vídeos varia entre 1 segundo e 2 minutos e 54 segundos, com duração média de aproximadamente 12 segundos. No total, o conjunto de dados acumula 55 horas, 29 minutos e 33 segundos de gravações. Exemplos de amostras do conjunto podem ser observados na Figura 4.1. A Tabela 4.2 apresenta as 31 classes de ações presentes no conjunto, enquanto a Figura 4.2 ilustra a distribuição de amostras por classe.

As características do TST introduzem desafios relevantes para o reconhecimento de atividades, incluindo alta variação intraclasse, forte desbalanceamento entre as classes, coexistência de atividades simples e compostas, presença de ações com padrões de movimento semelhantes e variação significativa na duração das atividades. Esses fatores tornam o conjunto particularmente adequado para a avaliação da robustez e da capacidade de generalização do método proposto neste trabalho.

Embora apresente limitações e maior complexidade, o conjunto TST ainda pode se

Tabela 4.2: As 31 classes de ações do conjunto Toyota Smarthome Trimmed.

ID	Ação (Inglês)	Tradução (Português)
1	*Cook.Cleandishes*	Cozinhar.Lavar louça
2	*Cook.Cleanup*	Cozinhar.Limpar
3	*Cook.Cut*	Cozinhar.Cortar
4	*Cook.Stir*	Cozinhar.Mexer
5	*Cook.Usestove*	Cozinhar.Usar fogão
6	*Cutbread*	Cortar pão
7	*Drink.Frombottle*	Beber.De uma garrafa
8	*Drink.Fromcan*	Beber.De uma lata
9	*Drink.Fromcup*	Beber.De uma xícara
10	*Drink.Fromglass*	Beber.De um copo
11	*Eat.Attable*	Comer.À mesa
12	*Eat.Snack*	Comer.Lanche
13	*Enter*	Entrar
14	*Getup*	Levantar-se
15	*Laydown*	Deitar-se
16	*Leave*	Sair
17	*Makecoffee.Pourgrains*	Fazer café.Colocar o pó
18	*Makecoffee.Pourwater*	Fazer café.Adicionar água
19	*Maketea.Boilwater*	Fazer chá.Ferver água
20	*Maketea.Insertteabag*	Fazer chá.Colocar o saquinho
21	*Pour.Frombottle*	Despejar.De uma garrafa
22	*Pour.Fromcan*	Despejar.De uma lata
23	*Pour.Fromkettle*	Despejar.De uma chaleira
24	*Readbook*	Ler um livro
25	*Sitdown*	Sentar-se
26	*Takepills*	Tomar remédios
27	*Uselaptop*	Usar laptop
28	*Usetablet*	Usar tablet
29	*Usetelephone*	Usar telefone
30	*Walk*	Caminhar
31	*WatchTV*	Assistir TV

beneficiar de modelos previamente treinados em bases amplas e consolidadas no problema de reconhecimento de ações, como o NTU RGB+D (SHAHROUDY et al., 2016). Assim, o uso de transferência de aprendizado torna-se uma estratégia adequada para este trabalho, pois permite aproveitar representações já aprendidas em um domínio mais geral e adaptá-las ao contexto específico de monitoramento de idosos, reduzindo o custo de treinamento e potencialmente melhorando o desempenho sobre o TST.

4.2.1 Protocolo do conjunto Toyota Smarthome

O conjunto de dados Toyota Smarthome fornece protocolos padronizados de divisão dos dados para fins de avaliação experimental. Neste trabalho, foi adotado o protocolo *Cross-Subject*, no qual os sujeitos utilizados para treinamento, validação e teste pertencem a grupos distintos, garantindo que o modelo seja avaliado em indivíduos não vistos durante

Figura 4.1: Amostras de quadros de vídeos do conjunto Toyota Smarthome Trimmed.

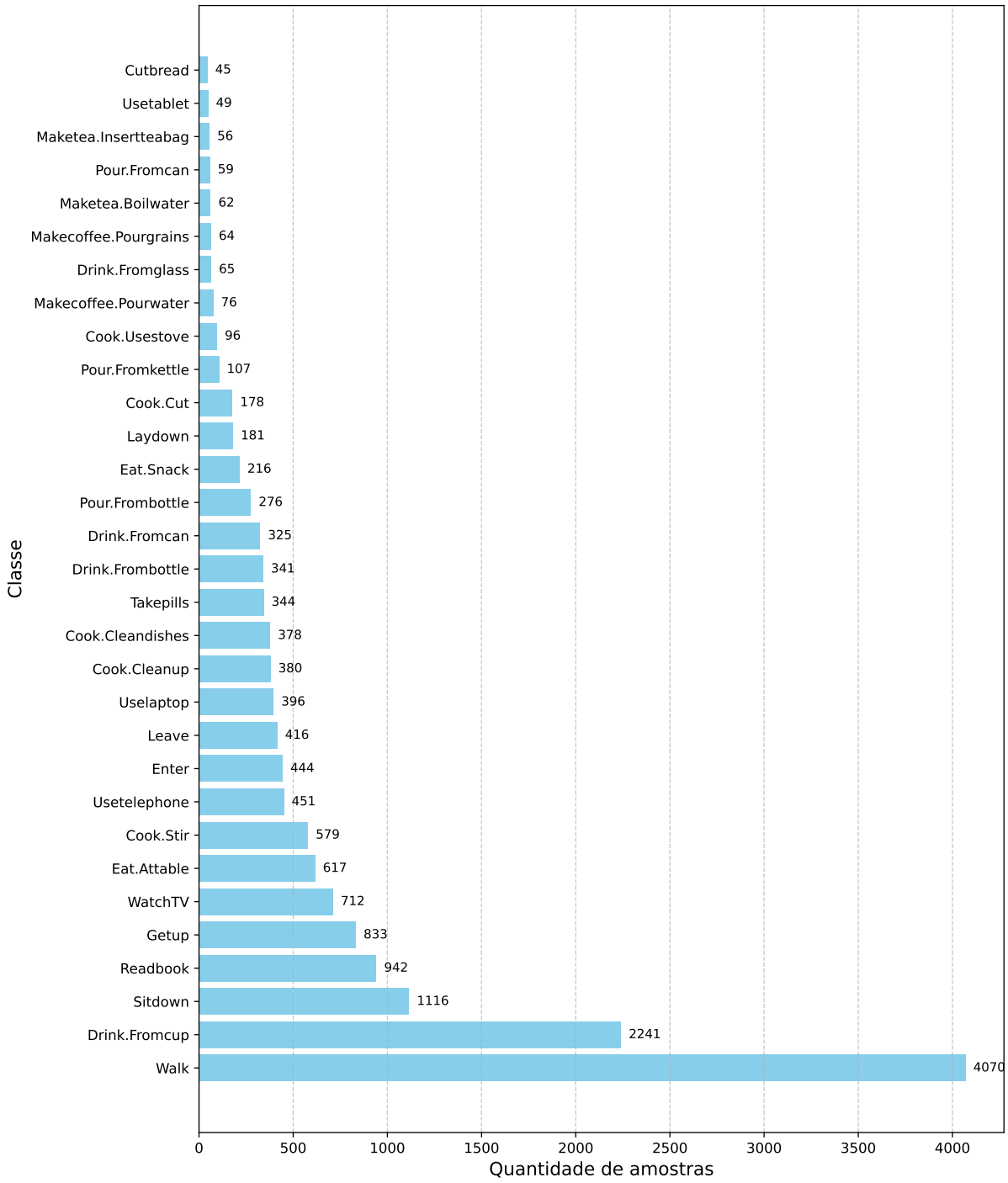


Figura 4.2: Distribuição de vídeos por classe no conjunto Toyota Smarthome Trimmed.

o treinamento. Neste protocolo os 18 indivíduos do conjunto de dados são divididos em grupos de treinamento, (indivíduos 3, 4, 6, 7, 9, 12, 13, 15 e 17), validação (indivíduos 19 e 25). Os 7 indivíduos restantes (2, 5, 8, 10, 11, 14 e 18) são reservados para o conjunto de teste.

As listas de divisão de dados foram utilizadas conforme disponibilizadas pelos

autores do conjunto no repositório do projeto [1], sem modificações, assegurando a reprodutibilidade dos experimentos e a comparabilidade com trabalhos relacionados. A Tabela 4.3 apresenta a distribuição das amostras entre os subconjuntos de treinamento, validação e teste.

Tabela 4.3: Divisão dos *splits* do conjunto Toyota Smarthome Trimmed no protocolo *Cross-Subject*.

Classe	**Treino**	**Validação**	**Teste**	**Total**
1	225	20	133	378
2	254	19	107	380
3	93	17	68	178
4	300	80	199	579
5	78	0	18	96
6	23	2	20	45
7	209	0	132	341
8	171	35	119	325
9	1115	379	747	2241
10	40	19	6	65
11	333	31	253	617
12	140	24	52	216
13	282	29	133	444
14	438	78	317	833
15	79	37	65	181
16	289	20	107	416
17	35	8	21	64
18	41	8	27	76
19	37	9	16	62
20	30	6	20	56
21	112	60	104	276
22	34	2	23	59
23	69	10	28	107
24	475	133	334	942
25	560	117	439	1116
26	177	29	138	344
27	184	34	178	396
28	34	0	15	49
29	251	53	147	451
30	2312	521	1237	4070
31	409	73	230	712
Total	**8829**	**1853**	**5433**	**16115**

[1] https://github.com/srijandas07/i3d_smarthome

5 Metodologia

Este capítulo apresenta a metodologia adotada para a realização dos experimentos de reconhecimento de ações humanas desenvolvidos neste trabalho. A contribuição do presente trabalho baseia-se na aplicação do *framework* PoseConv3D (DUAN et al., 2022) ao conjunto de dados Toyota Smarthome Trimmed bem como as análises de desempenho do modelo e conjunto de dados propostos.

Inicialmente, é apresentada uma visão geral do *framework* PoseConv3D e de seus principais componentes. Em seguida, são descritos o protocolo experimental adotado, os procedimentos de preparação do conjunto de dados e as configurações utilizadas para o treinamento, validação e avaliação do modelo.

5.1 *Framework* PoseConv3D

Dentro da categoria de redes neurais convolucionais, destacam-se aquelas que utilizam convoluções tridimensionais (3D-CNNs). Esse tipo de arquitetura é particularmente adequado para processar dados que possuem, além da dimensão espacial (2D), a dimensão temporal, como ocorre em sequências de vídeo. Assim, as 3D-CNNs configuram-se como uma escolha apropriada para o problema abordado neste trabalho.

O PoseConv3D (DUAN et al., 2022) é um *framework* que utiliza representações de juntas do corpo humano (*skeleton joints*) como entrada, combinadas a uma 3D-CNN. Para isso, as articulações do corpo são extraídas de cada quadro do vídeo e, posteriormente, convertidas em mapas de calor bidimensionais, que representam de forma densa a posição das juntas no espaço da imagem. Esses mapas de calor podem ser obtidos a partir de diferentes estimadores de pose. Os mapas são organizados ao longo do tempo, formando tensores tridimensionais que preservam simultaneamente a estrutura espacial e a sequência temporal das ações. Essa representação compacta e expressiva possibilita que o modelo capture padrões relevantes para a classificação das ações. Após a construção dos tensores tridimensionais de mapas de calor, essa representação é utilizada como entrada

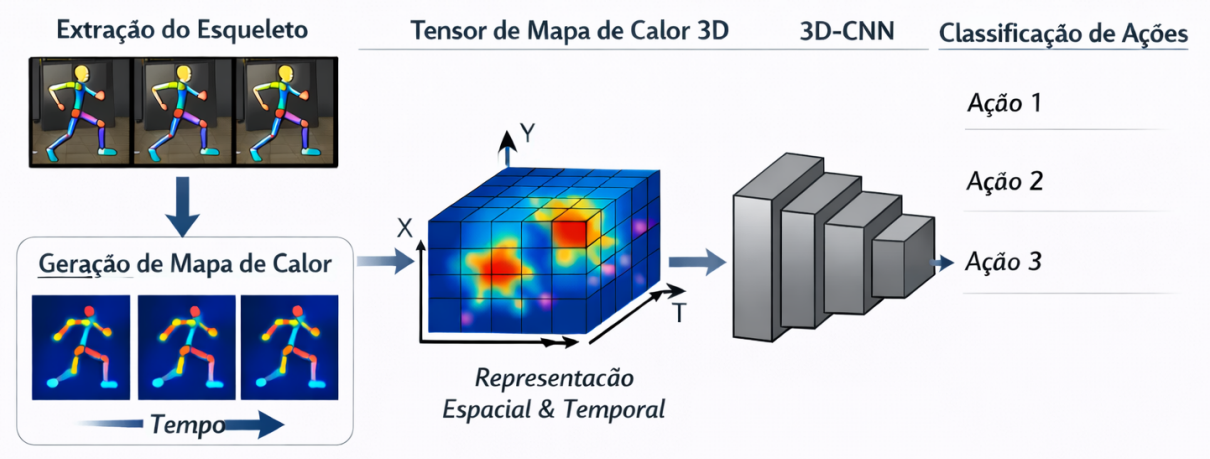


Figura 5.1: Visão geral do *framework* PoseConv3D para classificação de ações a partir de mapas de calor de articulações.

para a 3D-CNN. As convoluções 3D permitem a extração conjunta de padrões espaciais e temporais, explorando simultaneamente a configuração das articulações em cada quadro e sua evolução ao longo do tempo. Como resultado, a rede é capaz de aprender características discriminativas associadas às ações humanas, produzindo, ao final do processamento, uma predição de classe que indica a atividade realizada. A Figura 5.1 apresenta uma visão geral do *framework* PoseConv3D utilizado neste trabalho.

5.1.1 Extração de mapas de calor

A extração dos mapas de calor utilizados como entrada no PoseConv3D é realizada a partir de um estimador de pose baseado na arquitetura *High-Resolution Network* (HRNet) (WANG et al., 2020) e previamente treinado no conjunto de dados COCO (LIN et al., 2014a), amplamente utilizado para tarefas de detecção e estimativa de pose humana. A HRNet destaca-se por manter representações de alta resolução ao longo de toda a rede, característica essencial para localizar articulações de forma precisa em cenários com oclusões, múltiplas pessoas ou movimentos rápidos. Em vez de seguir o paradigma tradicional das redes convolucionais, que reduzem progressivamente a resolução espacial para ampliar o contexto semântico, a HRNet opera com múltiplos fluxos convolucionais em diferentes resoluções mantidos em paralelo. Esses fluxos comunicam-se continuamente por meio de módulos de troca de informações, de modo que as representações de alta resolução são enriquecidas com informações semânticas provenientes das resoluções mais

baixas. Como resultado, os mapas de calor produzidos pela HRNet são densos, espacialmente precisos e apresentam boa separabilidade entre as articulações, mesmo em situações complexas.

A HRNet também segue um paradigma *top-down*, no qual um detector de pessoas, baseado na rede Faster R-CNN (GIRSHICK, 2015), primeiro localiza o indivíduo na cena e fornece uma caixa delimitadora (*bounding box*). A rede extrai os mapas de calor dos pontos-chave apenas dentro desse recorte, de modo que, quando o sujeito está parcialmente fora do quadro, a caixa delimitadora também é truncada, reduzindo a quantidade de informação disponível para a estimativa de pose. Ainda assim, a HRNet apresenta certa robustez nesses cenários, pois mantém representações de alta resolução ao longo de toda a arquitetura e é treinada com estratégias de aumento de dados, como a *half-body augmentation*, que simula situações em que partes do corpo estão ocultas ou fora da imagem. Assim, mesmo com visibilidade parcial do sujeito, a rede consegue inferir posições de pontos-chave com razoável consistência, embora casos de truncamento extremo possam comprometer a qualidade dos mapas gerados.

Dentro do *framework*, para cada vídeo processado é gerado um arquivo no formato *pickle* contendo todas as informações necessárias para representar a sequência do esqueleto ao longo do tempo. Esse arquivo segue uma estrutura padronizada composta por metadados do vídeo e pelas anotações dos pontos-chave detectados que representam as juntas. Cada entrada contém o identificador do vídeo, o número total de quadros utilizados e as dimensões originais do vídeo registradas. Esses dados auxiliam tanto na depuração quanto na visualização e normalização das poses. As informações principais concentram-se no tensor *keypoint*, estruturado no formato

$$M \times T \times V \times C,$$

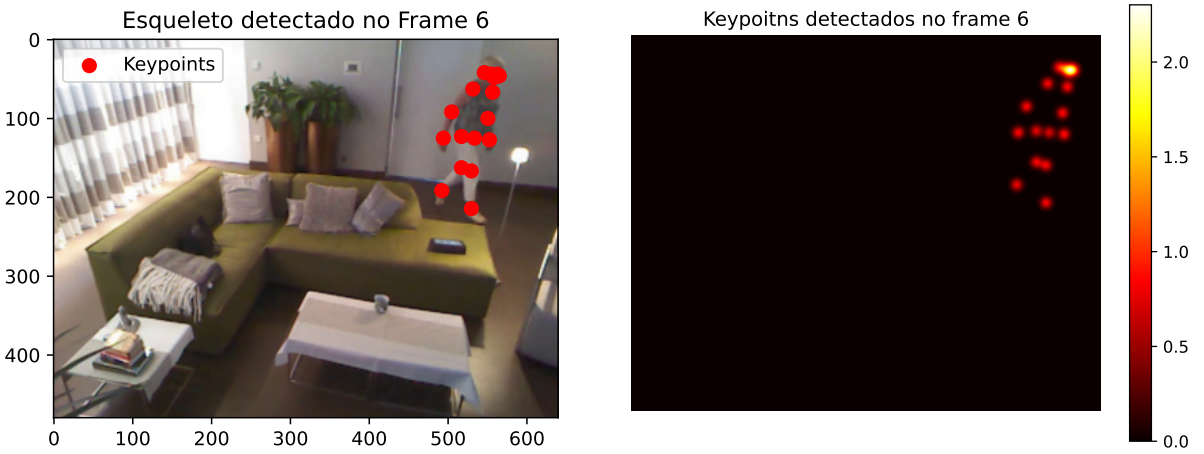
em que M representa o número de indivíduos na cena, T o número de quadros, V a quantidade de pontos-chave e C o número de coordenadas por junta ($C = 2$ para poses 2D ou $C = 3$ para poses 3D). Essa organização é compatível com diferentes convenções de esqueleto, como COCO (17 juntas), ou NTU RGB+D (25 juntas). Além das coordenadas,

o arquivo registra as pontuações de confiança no tensor *keypoint_score*, estruturado como

$$M \times T \times V,$$

indicando o grau de confiabilidade de cada detecção. A combinação entre coordenadas e escores fornece uma representação consistente da pose ao longo do tempo, servindo de base para a geração dos mapas de calor utilizados durante o processamento pelo PoseConv3D.

A Figura 5.2 mostra o resultado gerado neste estudo a partir do processamento de um quadro da ação *Walk*, no qual se observam os pontoa-chave identificados pela HRNet (a) e o mapa de calor associado às mesmas articulações (b).



(a) Detecção dos pontos-chave em um quadro de vídeo da ação *Walk* (Caminhar).

(b) Mapa de calor gerado a partir dos mesmos pontos-chave.

Figura 5.2: Comparação entre um quadro sobreposto com os *keypoints* detectados pela HRNet e o respectivo mapa de calor combinado utilizado como entrada para o processo de modelagem. No mapa de calor combinado, os valores variam de 0 (ausência de resposta ao redor do pixel) até aproximadamente 2, resultantes da soma das respostas gaussianas de múltiplas juntas.

A entrada no formato de mapas de calor confere ao PoseConv3D vantagens em relação a outros modelos baseados em estruturas 3D, como *Graph Convolutional Networks* (GCNs). Conforme apontado pelos autores, nas GCNs, as coordenadas de cada junta são armazenadas individualmente para cada pessoa, multiplicando-se pela resolução espacial, quantidade de quadros e número de juntas, o que faz com que o custo computacional cresça linearmente com o número de indivíduos no vídeo. Já no PoseConv3D, todas as pessoas são representadas em um mesmo volume de mapas de calor: cada canal corresponde a uma junta específica, e múltiplos picos dentro do canal representam a ocorrência dessa

junta em diferentes indivíduos. Por exemplo, em uma cena contendo duas pessoas, o canal correspondente à junta "cabeça" apresentará dois picos de ativação, cada um indicando a posição da cabeça de um indivíduo distinto. Esse padrão se repete nos demais canais de juntas, permitindo que o modelo represente múltiplas pessoas simultaneamente em um único volume de mapas de calor.

Essa abordagem reduz de forma significativa a quantidade de parâmetros da rede, sem comprometer a expressividade necessária ao reconhecimento de ações.

5.1.2 O modelo X3D

O PoseConv3D permite a aplicação de diferentes *backbones* de 3D-CNNs, proporcionando flexibilidade na escolha da arquitetura subjacente. Entre as opções sugeridas no trabalho original, a rede escolhida nesta metodologia é a X3D (FEICHTENHOFER, 2020), cuja estrutura segue o *design* de uma ResNet tridimensional, uma rede composta por blocos residuais 3D organizados em estágios progressivos de convolução. A X3D é uma extensão da X2D que introduz escalonamento progressivo nas dimensões espacial, temporal e de capacidade do modelo. Esse escalonamento é realizado por meio de fatores de escala explícitos: o fator de largura (γ_w), o fator de gargalo (γ_b), o fator de profundidade (γ_d), o fator temporal (γ_t) e o *stride* temporal (γ_τ), e o fator espacial (γ_s). Os fatores γ_t e γ_τ influenciam a etapa de entrada e a camada inicial (*stem*); o fator γ_s define a resolução espacial das ativações; enquanto os fatores γ_w, γ_b e γ_d ajustam, respectivamente, o número de canais, a expansão interna dos blocos residuais e o número de repetições em cada estágio. As diferentes configurações de fatores produzem variantes da X3D, sem alteração da organização estrutural da rede, como a X3D-XS, X3D-S, X3D-M e X3D-L.

A estrutura base comum da X3D se apresenta da seguinte forma: a rede é iniciada por um estágio *stem* baseado em convoluções tridimensionais, responsável pela extração inicial de padrões espaço-temporais de baixo nível a partir das sequências de entrada. Especificamente nessa etapa inicial, a X3D emprega uma convolução espaço-temporal separável, na qual a modelagem espacial e temporal é realizada de forma desacoplada. Após o *stem*, as representações são progressivamente refinadas ao longo de quatro estágios residuais hierárquicos, identificados como `res_2` a `res_5`. Cada estágio é formado por

blocos residuais do tipo *bottleneck*, que utilizam convoluções tridimensionais compactas para expandir a capacidade representacional do modelo. Ao final do processo de extração espaço-temporal, as ativações são agregadas por meio da operação de *Global Average Pooling*, que consolida as informações ao longo das dimensões espacial e temporal. Essa representação compacta é então utilizada pela cabeça de classificação para realizar a predição da ação observada.

No trabalho do PoseConv3D, a variação X3D-S oferecia alto desempenho com o menor número de parâmetros e operações de ponto flutuante (FLOPs), portanto foi a variação escolhida para os experimentos do presente trabalho. Os autores destacam adaptações necessárias para que as 3D-CNNs processassem a entrada no novo formato proposto, o de pose. A ideia é remover as reduções de resolução (*down-sampling*) nas primeiras camadas que ocorrem no estágio *stem*, pois os volumes de mapas de calor já possuem dimensão adequada e menor quando comparados a quadros de vídeo (RGB). Além disso, utilizar uma arquitetura mais leve, com menos camadas e canais, já seria suficiente para capturar a dinâmica espaciotemporal das ações. Dessa forma, os autores implementaram uma versão reduzida da X3D-S, denominada Pose-X3D-S, onde a quantidade de camadas convolucionais foi ajustada a partir da atribuição do valor 1 ao fator de profundidade γ_d. Também houve a remoção do primeiro estágio da rede e do último estágio residual.

A Tabela 5.1 mostra as diferenças entre a arquitetura original e a reduzida. O *stem* original do modelo é configurado com um *stride* temporal $\gamma_\tau = 6$, correspondente à amostragem de um quadro a cada seis, e otimizado para entradas RGB com resolução 160×160. Esse *stem* é substituído por uma versão adaptada, utilizando $\gamma_\tau = 1$, de modo a preservar integralmente a resolução temporal das sequências de pose com dimensão 56×56. Além disso, o bloco `res_5` é removido em sua totalidade, assim como a camada `conv_5`. Dessa forma, a rede passa a encaminhar as ativações diretamente do bloco `res_4` para uma convolução 1×1 com 216 canais, seguida da operação de *Global Average Pooling*.

Tabela 5.1: Comparação das arquiteturas das redes X3D-S Original e Pose-X3D-S adaptada.

Estágio (Residual)	**Descrição do Bloco/Camada**	**X3D-S Original** ($\gamma_d \approx 2.2$)	**Pose-X3D-S Adaptada** ($\gamma_d = 1$)
stem	Data Layer e Conv_1: Amostragem Temporal e espacial.	*Stride* $\gamma_\tau = 6$	*Stride* $\gamma_\tau = 1$
res_2	Bloco residual com filtros $[1 \times 1^2, 24]$	$\times$3 repetições	$\times$2 repetições
res_3	Bloco residual com filtros $[1 \times 1^2, 48]$	$\times$5 repetições	$\times$5 repetições
res_4	Bloco residual com filtros $[1 \times 1^2, 96]$	$\times$11 repetições	$\times$3 repetições
res_5	Bloco residual com filtros $[1 \times 1^2, 192]$	$\times$7 repetições	**Removido**

5.2 Protocolo experimental

O presente trabalho envolve a aplicação de um modelo previamente proposto (PoseConv3D) a um conjunto de dados específico (Toyota Smarthome Trimmed). Nesse contexto, é fundamental distinguir os protocolos definidos pelos autores do conjunto de dados, aqueles inerentes ao *framework* PoseConv3D e as decisões metodológicas assumidas neste estudo.

5.2.1 Protocolo do *framework* PoseConv3D

O PoseConv3D define um protocolo próprio de treinamento relacionado a arquitetura da rede, o *pipeline* de processamento baseado em mapas de calor de articulações e as configurações de otimização utilizadas durante o aprendizado. Neste trabalho, foi utilizada a rede X3D-S adaptada por Duan et al. (2022) e implementada na biblioteca PySkl [2] de mesma autoria. São disponibilizados também os arquivos de pesos de treinamento dessa rede em diferentes conjuntos de dados, considerandos protocolos de divisão do próprio conjunto escolhido. Essa funcionalidade possibilitou a aplicação de pré-treinamento aos experimentos apresentados no Capítulo 6.

[2] https://github.com/kennymckormick/pyskl

5.3 Preparação do conjunto de dados

A obtenção do conjunto de dados TST para utilização no presente trabalho foi realizada sob demanda, diretamente a partir do site oficial do projeto[3]. Para a reprodução do protocolo experimental apresentado na Seção 4.2.1, foram utilizadas as listas de divisão de dados (*splits*) em formato `.txt` disponibilizadas no repositório indicado pelos criadores do conjunto. Esses arquivos descrevem os subconjuntos de treinamento, validação e teste, de acordo com o protocolo de avaliação escolhido. O protocolo *Cross-Subject* (CS) foi adotado porque favorece a avaliação da capacidade de generalização do modelo, uma vez que garante que o sistema reconheça ações executadas por indivíduos não vistos durante o treinamento.

Foi desenvolvida uma rotina para converter as anotações originais do TST em arquivos `.json`, de forma a adequá-las ao formato esperado pelo PoseConv3D. A rotina foi construída a partir dos arquivos `.csv` das listas de *splits* do TST. Atividades compostas, como *Cook.CleanDishes*, foram renomeadas a partir da substituição do separador "."para "__"(*Cook__CleanDishes*), permitindo a organização correta dos quadros individuais em pastas para entrada no PoseConv3D e resolvendo erros de leitura de diretório. Listas de vídeos para a extração dos esqueletos foram geradas para cada *split* do conjunto TST conforme rotina especificada pela ferramenta PySkl.

A etapa seguinte consistiu na extração dos esqueletos bidimensionais (*2D skeletons*) por meio da rede HRNet. Essa extração foi realizada utilizando a ferramenta PySkl que fornece rotinas específicas para converter vídeos em anotações no formato `.pkl`, compatíveis com o *framework* PoseConv3D, além de possuir implementações de diversos modelos de CNNs. O processo foi aplicado a todos os arquivos de lista de *splits* (treino, validação e teste) do Toyota Smarthome Trimmed, resultando em um conjunto completo de anotações de mapas de calor armazenados em um único arquivo *pickle*. Cada anotação contém, para cada vídeo, informações sobre o número de quadros, dimensões da imagem, rótulo da ação e coordenadas dos pontoa-chave extraídos. Esse conjunto padronizado constitui a base de dados de entrada utilizada nas etapas subsequentes de treinamento e avaliação do modelo.

[3]https://project.inria.fr/toyotasmarthome

6 Experimentos e resultados

Neste capítulo são apresentados e analisados os experimentos realizados para avaliar o desempenho do modelo no reconhecimento de ações humanas. Inicialmente, são conduzidos testes de inferência sobre o conjunto TST utilizando um modelo pré-treinado no conjunto NTU RGB+D. O objetivo é analisar o comportamento da rede no conjunto TST a partir do pré-treinamento em um conjunto distinto, porém semanticamente relacionado, bem como avaliar a viabilidade da estratégia de pré-treinamento adotada.

Em seguida, são conduzidos experimentos com o *framework* PoseConv3D utilizando diferentes subconjuntos do conjunto de dados TST, bem como análises do impacto de diferentes configurações de treinamento. Posteriormente, os experimentos são estendidos ao conjunto completo de dados, permitindo uma avaliação mais abrangente. Por fim, os resultados obtidos são comparados com trabalhos da literatura e é explorado um experimento adicional de agrupamento semântico de classes, com o objetivo de analisar padrões de confusão e similaridade entre as ações.

Os experimentos descritos neste trabalho foram executados em ambiente Linux, utilizando GPUs distintas em função da evolução do *pipeline* experimental e do aumento progressivo do volume de dados. Nos experimentos iniciais e nas avaliações conduzidas sobre conjuntos reduzidos e intermediários foi utilizada uma GPU NVIDIA GeForce GTX 1050 com 2 GB de memória de vídeo (VRAM). Posteriormente, para a realização dos experimentos envolvendo o conjunto completo de dados, foi empregada uma GPU NVIDIA GeForce GTX 1660 com 6 GB de VRAM. Essa mudança permitiu a ampliação do número de vídeos processados por GPU e viabilizou o treinamento em maior escala, mantendo-se a mesma arquitetura e *pipeline* de processamento. As demais configurações de software, incluindo versões de CUDA, *drivers* e bibliotecas auxiliares, seguiram as definições padrão da ferramenta PySkl, conforme disponibilizado por seus desenvolvedores.

6.1 Análise exploratória de inferência cruzada entre NTU RGB+D e TST

O objetivo desta etapa foi analisar qualitativamente o comportamento do modelo PoseConv3D, utilizando a rede X3D pré-treinada no conjunto NTU RGB+D, ao processar vídeos do Toyota Smarthome Trimmed. Foram utilizados 27 vídeos (e suas anotações de mapas de calor) de ações cotidianas comuns no TST — como "levantar-se", "sentar-se", "deitar-se", "comer", "usar o telefone" e "caminhar" — por serem representativas para um mínimo de monitoramento, enquanto outras ações apresentavam alta complexidade e pouca correspondência com o NTU. O objetivo não era avaliar acurácia, mas compreender como o conhecimento adquirido durante o pré-treinamento se manifesta em ações não vistas, explorando correspondências posturais, ambiguidades semânticas e limitações da abordagem baseada apenas em esqueletos 2D. Para cada predição, também foi registrado um valor de confiança, correspondente à probabilidade atribuída à classe prevista pelo modelo, permitindo observar o grau de certeza das inferências em cada vídeo.

A Tabela 6.1 mostra que o modelo fez predições correspondentes a 13 das 60 classes do NTU RGB+D em pelo menos um vídeo do TST. As classes do conjunto NTU RGB+D são apresentadas na Tabela 6.2, as classes detectadas pelo modelo foram destacadas em negrito. O modelo foi capaz de reconhecer significativamente ações com padrões corporais bem definidos, especialmente aquelas relacionadas a mudanças de postura, como "levantar-se" e "deitar-se". É importante notar, entretanto, que valores altos de confiança não implicam que a classe prevista seja semanticamente equivalente à ação do TST; eles indicam apenas que o modelo reconheceu padrões posturais semelhantes aos aprendidos durante o pré-treinamento. Esses resultados fornecem uma visão inicial sobre como o modelo pré-treinado interpreta dinâmicas corporais em um domínio novo, destacando tanto correspondências plausíveis quanto limitações da abordagem baseada apenas em esqueletos 2D.

Neste contexto, foi observada a correspondência entre ações de "sentar-se"e "deitar-se" do TST e classes do NTU RGB+D associadas a posturas curvadas, como "náusea ou vômito" ou "calçar sapato". Essa relação evidencia que o modelo, por basear-se exclu-

Tabela 6.1: Resultados dos testes de inferência com rede pré-treinada no NTU RGB+D.

Vídeo (TST)	**Ação (TST)**	**Duração (s)**	**Tempo de Inferência (s)**	**Classe Prevista (NTU RGB+D)**	**Confiança**
Getup_p14_r03_v02_c04	Levantar-se	2	0.600	Levantar-se (da posição sentada)	0.981
Getup_p02_r00_v06_c05	Levantar-se	1	0.701	Náusea ou vômito	0.810
Getup_p16_r01_v02_c05	Levantar-se	2	0.623	Levantar-se (da posição sentada)	0.771
Leave_p02_r00_v01_c07	Sair	3	0.666	Cambalear	0.760
Sitdown_p02_r00_v04_c04	Sentar-se	2	0.627	Calçar sapato	0.720
Usetablet_p20_r02_v10_c01	Usar tablet	24	0.697	Ler	0.704
Enter_p10_r00_v02_c05	Entrar	5	0.505	Caminhar afastando-se	0.688
Leave_p10_r00_v01_c05	Sair	4	0.620	Cambalear	0.636
Laydown_p14_r00_v02_c04	Deitar-se	5	0.569	Náusea ou vômito	0.610
Laydown_p02_r00_v07_c04	Deitar-se	5	0.603	Náusea ou vômito	0.582
Eat__Attable_p02_r00_v11_c01	Comer à mesa	2	0.665	Ler	0.536
Laydown_p11_r02_v02_c04	Deitar-se	6	0.608	Náusea ou vômito	0.505
Leave_p14_r00_v02_c04	Sair	4	0.658	Abraçar outra pessoa	0.433
Leave_p11_r00_v06_c04	Sair	6	0.687	Abraçar outra pessoa	0.409
Laydown_p10_r01_v03_c04	Deitar-se	5	0.602	Vestir jaqueta	0.405
Enter_p14_r00_v05_c04	Entrar	4	0.494	Caminhar em direção ao outro	0.333
Enter_p11_r00_v03_c04	Entrar	7	0.556	Caminhar afastando-se	0.314
Walk_p16_r00_v01_c04	Caminhar	5	0.781	Calçar sapato	0.306
Enter_p20_r01_v16_c07	Entrar	5	0.887	Tocar as costas (dor nas costas)	0.293
Sitdown_p11_r00_v05_c04	Sentar-se	2	2.294	Calçar sapato	0.290
Getup_p10_r00_v09_c01	Levantar-se	5	0.572	Ler	0.282
Walk_p02_r00_v01_c06	Caminhar	2	0.569	Calçar sapato	0.219
Sitdown_p14_r00_v02_c05	Sentar-se	7	2.044	Ler	0.165
Sitdown_p10_r00_v02_c05	Sentar-se	2	1.524	Náusea ou vômito	0.155
Usetelephone_p02_r00_v01_c06	Usar telefone	8	0.874	Limpar o rosto	0.131
Eat__Attable_p02_r08_v11_c02	Comer à mesa	2	0.744	Sentar-se	0.115
Eat__Attable_p02_r14_v11_c01	Comer à mesa	4	0.768	Usar celular/tablet	0.109

sivamente em informações posturais, tende a associar ações com configurações corporais semelhantes, independentemente do contexto da cena. De modo semelhante, ações como "entrar" e "sair" foram mapeadas para classes como "caminhar afastando-se" ou "caminhar em direção ao outro", o que indica que o modelo reconhece padrões locomotores compartilhados entre os conjuntos. Outras correspondências posturais também se mostraram plausíveis, como "usar tablet" ou "comer à mesa" sendo associadas à classe "ler", todas caracterizadas por leve inclinação do tronco e foco visual direcionado para um objeto próximo.

Por outro lado, algumas predições demonstraram falta de correspondência semântica, evidenciando limitações da abordagem baseada apenas em pose e destacando a diferença quanto a variabilidade de classes entre os dois conjuntos, uma vez que o NTU RGB+D possui praticamente o dobro de ações que o TST. Um exemplo particularmente ilustrativo ocorre nas ações de "sair" (*Leave*), que foram classificadas como "abraçar outra pessoa" ou "cambalear". Embora essas classes pareçam incompatíveis à primeira vista, a análise de quadros dos vídeos em questão, exibida na Figura 6.1, revela que, na maioria dos casos (*Leave_p02_r00_v01_c07, Leave_p14_r00_v02_c04 , Leave_p11_r00_v06_c04*) o indivíduo interage fisicamente com a porta ao abrir ou empurrá-la. Esse gesto altera a configuração

Tabela 6.2: Classes do NTU RGB+D com ações detectadas nas amostras do conjunto TST destacadas em negrito.

ID	Ação	ID	Ação	ID	Ação
A1	Beber água	A21	Tirar chapéu/boné	A41	Espirrar/tossir
A2	Comer refeição/lanche	A22	Comemorar	A42	**Cambalear**
A3	Escovar os dentes	A23	Acenar com a mão	A43	Cair
A4	Escovar o cabelo	A24	Chutar algo	A44	Tocar a cabeça (dor de cabeça)
A5	Deixar cair	A25	Alcançar o bolso	A45	Tocar o peito (dor no coração/estômago)
A6	Pegar objeto	A26	Pular em um pé	A46	**Tocar as costas (dor nas costas)**
A7	Arremessar	A27	Saltar	A47	Tocar o pescoço (dor no pescoço)
A8	**Sentar-se**	A28	Fazer/atender chamada telefônica	A48	**Náusea ou vômito**
A9	**Levantar-se (da posição sentada)**	A29	**Usar celular/tablet**	A49	Usar ventilador (sentindo calor)
A10	Bater palmas	A30	Digitar no teclado	A50	Socar/tapar outra pessoa
A11	**Ler**	A31	Apontar com o dedo	A51	Chutar outra pessoa
A12	Escrever	A32	Tirar selfie	A52	Empurrar outra pessoa
A13	Rasgar papel	A33	Verificar horário no relógio	A53	Dar tapinha nas costas de outra pessoa
A14	**Vestir jaqueta**	A34	Esfregar as mãos	A54	Apontar para outra pessoa
A15	Tirar jaqueta	A35	Assentir com a cabeça	A55	**Abraçar outra pessoa**
A16	**Calçar sapato**	A36	Negar com a cabeça	A56	Entregar algo a outra pessoa
A17	Tirar sapato	A37	**Limpar o rosto**	A57	Tocar o bolso de outra pessoa
A18	Colocar óculos	A38	Saudar (continência)	A58	Apertar as mãos
A19	Tirar óculos	A39	Juntar as palmas das mãos	A59	**Caminhar em direção ao outro**
A20	Colocar chapéu/boné	A40	Cruzar as mãos à frente	A60	**Caminhar afastando-se**

dos braços e do tronco, produzindo uma postura semelhante à de outras ações do NTU, o que explica as predições com confiança relativamente alta, apesar da diferença semântica entre as classes. E no vídeo *Leave_p10_r00_v01_c05* ocorre oclusão de uma das pernas, o que também pode ter contribuído para a predição de "cambalear". Situações similares ocorreram em predições como "usar telefone" sendo classificado como "limpar o rosto" ou "caminhar" sendo confundido com "calçar sapato". Esses casos reforçam que, na ausência de pistas contextuais, como objetos manipulados ou elementos estruturais do ambiente, o modelo tende a confundir ações distintas que compartilham padrões corporais semelhan-

tes.

Figura 6.1: Exemplos de quadros dos vídeos da classe *"Leave"* ilustrando situações que influenciaram as predições do modelo, incluindo interação com objetos do ambiente, oclusões parciais do corpo e posturas ambíguas.

De modo geral, as classificações com maior confiança indicam que a rede consegue identificar corretamente padrões biomecânicos fundamentais, validando a eficácia da representação baseada em mapas de calor. Já os erros em amostras de menor confiança refletem o desafio natural de interpretar ações complexas e com baixa variabilidade postural e movimentos de pequena amplitude, apenas a partir da postura, sem o suporte de informações visuais complementares. Além disso, parte dessas inconsistências decorre da inexistência de uma correspondência direta entre as ações do Toyota Smarthome e as classes disponíveis no modelo pré-treinado no NTU RGB+D. Uma análise nominal dos 31 vídeos avaliados revela que apenas 8 ações possuem equivalente direto ou semanticamente próximo no NTU RGB+D: *Getup* (Levantar-se da posição sentada, A9), *Sitdown* (Sentar-se, A8), *Eat.Attable* e *Eat.Snack* (Comer refeição/lanche, A2), *Readbook* (Ler, A11), *Usetablet* e *Usetelephone* (Usar celular/tablet, A29), e *Walk* (Caminhar afastando-se, A60 / Caminhar em direção ao outro, A59). As demais ações, como cozinhar, beber

Figura 6.2: Exemplo de oclusão do indivíduo durante a ação de “sentar-se”.

de copo ou lata, cortar pão e assistir TV, não possuem equivalente nominal no NTU. Dessa forma, mesmo quando o modelo apresenta níveis relativamente altos de confiança em suas predições, grande parte delas não reflete correspondência semântica direta, mas sim similaridade postural com as classes aprendidas.

Sobre a duração dos vídeos e dos tempos de inferência, a análise conjunta revela que o tempo total do arquivo não é o principal fator que determina o custo computacional do modelo. Embora vídeos mais longos possam sugerir maior processamento, os resultados mostram que a inferência depende predominantemente da qualidade e da completude das poses extraídas em cada quadro. Um caso ilustrativo é o vídeo *Sitdown_p11_r00_v05_c04*, que possui apenas 2 segundos de duração, mas apresentou um dos maiores tempos de inferência (2.294 s). Esse comportamento ocorre porque o indivíduo permanece fora do campo de visão na maior parte da sequência, como pode ser visto na Figura 6.2, o que dificulta a detecção de pose e leva o extrator a realizar tentativas adicionais para localizar ou ajustar pontos corporais ausentes, aumentando substancialmente o tempo de processamento. Assim, observa-se que o tempo de inferência está mais relacionado à complexidade da estimativa de pose diante de oclusões e quadros incompletos do que ao tempo total do vídeo.

Os resultados apresentados nesta seção reforçam o potencial do PoseConv3D como ponto de partida promissor para o reconhecimento de ações no Toyota Smarthome, ao mesmo tempo em que destacam a necessidade de um ajuste fino para especializar o modelo às características específicas desse conjunto de dados.

6.2 Experimentos com o modelo treinado no TST

Após a análise exploratória do comportamento do modelo pré-treinado, esta seção apresenta os experimentos realizados com o PoseConv3D treinado especificamente no conjunto de dados do Toyota Smarthome (TST). Embora a rede utilize pesos inicializados a partir do pré-treinamento no NTU RGB+D, o foco desta etapa passa a ser a avaliação do desempenho do modelo após sua adaptação supervisionada às classes e às características próprias do TST.

Os experimentos aqui descritos têm como objetivo investigar a capacidade de aprendizado do modelo quando exposto a dados do domínio alvo, analisando tanto o impacto do volume e da distribuição das amostras quanto a evolução do desempenho ao longo de diferentes configurações de treinamento. Diferentemente da etapa anterior, em que as predições refletiam apenas a transferência direta de padrões aprendidos no NTU RGB+D, os resultados apresentados nesta seção correspondem a um cenário de treinamento efetivo, no qual o modelo passa a aprender associações semânticas específicas das ações presentes no TST.

Dessa forma, esta etapa permite avaliar não apenas a eficácia da arquitetura PoseConv3D no contexto do Toyota Smarthome, mas também o papel do pré-treinamento como ponto de partida para a convergência e para a extração de padrões biomecânicos relevantes em um conjunto de dados menor, mais desbalanceado e com ações de maior similaridade postural.

6.2.1 Métricas de avaliação

A avaliação do desempenho do modelo foi realizado a partir das seguintes métricas:

- **Acurácia Global:** corresponde à proporção total de vídeos corretamente classificados pelo modelo em relação ao número total de amostras no conjunto de teste. Essa métrica reflete o desempenho geral do classificador, sendo sensível à distribuição das classes.

- **Acurácia Média (por Classe):** definida como a média das acurácias calculadas individualmente para cada classe de ação no conjunto de teste. Diferentemente da

acurácia global, essa métrica atribui o mesmo peso a todas as classes, independentemente do número de amostras, permitindo uma avaliação mais equilibrada do desempenho do modelo.

- **Diferença Absoluta (DIF):** corresponde à diferença, em pontos percentuais (p.p.), entre a acurácia global e a acurácia média por classe. Essa métrica foi utilizada como um indicador direto do efeito do desbalanceamento entre classes, permitindo quantificar o quanto o desempenho global do modelo é influenciado por classes com maior número de amostras. Valores elevados de DIF indicam maior discrepância entre o desempenho médio por classe e o desempenho agregado do modelo.

- **Desvio padrão entre classes (DP):** Além da acurácia média por classe, foi considerado o desvio padrão da acurácia média por classe como forma de quantificar a variabilidade do desempenho do modelo entre as diferentes ações. Essa métrica mede o grau de dispersão das acurácias individuais das classes em torno da média, permitindo avaliar o quão uniforme é o comportamento do classificador ao longo do conjunto de ações. Valores elevados de desvio padrão indicam que o modelo apresenta desempenho desigual entre as classes, geralmente associado ao desbalanceamento do conjunto de dados ou à presença de ações com padrões posturais mais ambíguos e de difícil distinção.

6.2.2 Experimentos com subconjuntos do conjunto de dados

Para definir a configuração experimental de referência (*baseline*), foram conduzidos experimentos com subconjuntos de tamanho reduzido do conjunto TST, organizados de forma incremental. Embora esses subconjuntos representem apenas uma fração do conjunto completo, eles foram construídos com volumes progressivamente maiores entre si, permitindo analisar o impacto do aumento gradual de dados e da redistribuição das classes no desempenho do modelo.

Os subconjuntos avaliados, renomeados para indicar a origem e o tamanho, incluíram: Toyota[1.270], Toyota[1.645], Toyota[1.827], Toyota[1.922] e Toyota[1.753], além do conjunto completo com 16.115 amostras (Toyota[16.115]). A Tabela 6.3 detalha a

evolução do número de amostras nos conjuntos de treino, validação e teste ao longo dos diferentes estágios experimentais, reforçando a lógica incremental adotada e o equilíbrio buscado entre classes.

Tabela 6.3: Evolução do número de amostras nos conjuntos de treino, validação e teste para os subconjuntos do TST.

Subconjunto	**Treino**	**Validação**	**Teste**	**Estágio do conjunto**
Toyota[1.270]	910	160	200	Conjunto inicial
Toyota[1.645]	1285	160	200	Expansão do conjunto de treino
Toyota[1.827]	1285	160	382	Expansão do conjunto de teste
Toyota[1.922]	1285	255	382	Ajuste da proporção treino/validação para 80%/20%
Toyota[1.753]	1192	255	306	Conjunto com teto de 40 amostras por classe no treino e 10 para validação

A Tabela 6.4 resume os resultados obtidos, incluindo a acurácia global, a média das acurácias entre classes no conjunto de teste e a diferença. Essa abordagem incremental e a apresentação das métricas citadas permitem compreender o impacto do tamanho do conjunto de dados, do balanceamento entre classes e da evolução da maturidade do conjunto na performance do modelo.

Tabela 6.4: Comparação entre acurácia global, acurácia média e diferença absoluta entre acurácias no conjunto de teste para diferentes subconjuntos do TST.

Subconjunto	**Acurácia Global (%)**	**Acurácia Média (%)**	**DIF (p.p.)**
Toyota[1.270]	51,0	43,9	7,1
Toyota[1.645]	52,0	41,8	10,2
Toyota[1.827]	50,0	48,1	1,9
Toyota[1.922]	47,9	46,4	1,5
Toyota[1.753]	49,7	49,7	0,0

O subconjunto Toyota[1.270] teve como objetivo principal a validação do *pipeline* de processamento e a verificação da convergência inicial do modelo. A quantidade de amostras por classe no conjunto de treino variava de 16 a 83, com média de 27 amostras. Nessa etapa, buscou-se assegurar que a arquitetura PoseConv3D era capaz de processar corretamente as sequências de poses extraídas, mesmo diante de uma distribuição altamente desigual entre as classes. Não houve, nesse estágio, exploração sistemática de hiperparâmetros, sendo utilizadas configurações pré-definidas da arquitetura X3D originalmente ajustadas para o conjunto NTU RGB+D, com adaptações restritas ao número

de classes e à quantidade de vídeos processados por GPU. Em função do volume reduzido de dados, os resultados obtidos foram considerados estritamente preliminares e utilizados apenas como verificação funcional do *pipeline*. Posteriormente o conjunto de dados passou a ser reorganizado de forma incremental, com o objetivo de manter uma divisão aproximada de 80% das amostras destinadas ao treinamento e 20% à validação. Esse processo de ampliação do subconjunto foi conduzido considerando, adicionalmente, a manutenção de uma distribuição o mais equilibrada possível de amostras entre as classe.

Nos subconjuntos de 1.645 e 1.827 amostras, foi introduzida uma primeira tentativa de balanceamento da distribuição, com a imposição de um teto de 42 amostras por classe, com quantidade mínima de 23 amostras por classe. A exceção foi a classe *Walk*, que permaneceu com 83 amostras desde o primeiro subconjunto, devido à sua natureza distinta e ao fato de não apresentar confusões relevantes com outras ações. Apesar desse ajuste, até o Toyota[1.645] o modelo ainda apresentava desempenho global limitado, com taxas de predição muito baixas para diversas classes. Como consequência, algumas categorias sequer apareciam na matriz de confusão, uma vez que não eram preditas em nenhuma instância, o que inviabilizava uma análise qualitativa completa nesse estágio e motivou a não exibição dessas matrizes.

A partir do subconjunto Toyota[1.827] a matriz de confusão passou a estar completa, possibilitando-se identificar padrões de erro mais consistentes, como exibido na Figura 6.3. Observou-se que as principais confusões não estavam relacionadas ao desbalanceamento residual das classes, mas sim à similaridade semântica e postural entre determinadas ações. Em particular, verificaram-se confusões recorrentes entre classes que compartilham a mesma dinâmica corporal básica e diferem predominantemente pelo objeto com o qual o indivíduo interage. Esse comportamento já era esperado e foi evidente em grupos de ações como *Drink.Frombottle*, *Drink.Fromcan*, *Drink.Fromcup* e *Drink.Fromglass*, todas caracterizadas por movimentos semelhantes dos membros superiores em direção à região da face, variando apenas pelo tipo de recipiente manipulado. Padrão análogo foi observado entre as classes *Pour.Frombottle*, *Pour.Fromcan* e *Pour.Fromkettle*, bem como entre ações de preparação de alimentos e bebidas, como *Cook.Cleandishes*, *Cook.Cleanup* e *Cook.Cut*, cujas diferenças semânticas estão fortemente associadas ao contexto e aos

objetos presentes na cena, e não à postura corporal isolada. De forma semelhante, atividades envolvendo leitura e uso de dispositivos eletrônicos, incluindo *ReadBook*, *UseLaptop*, *UseTablet* apresentaram confusões frequentes. Essas ações compartilham uma postura predominantemente estática, com leve inclinação do tronco e orientação do olhar para um objeto à frente do corpo, o que dificulta a distinção baseada exclusivamente em informações de pose.

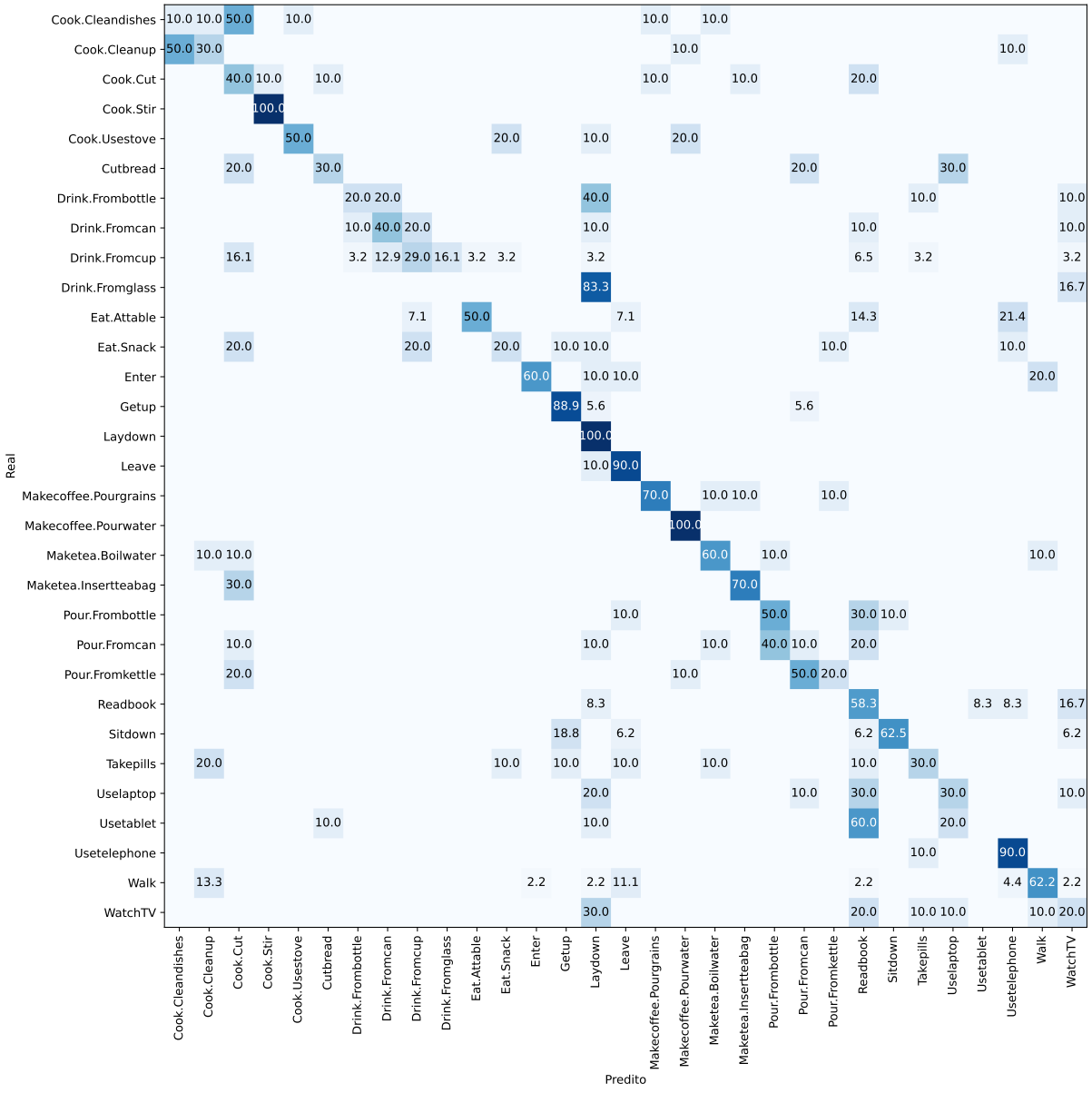


Figura 6.3: Matriz de Confusão do conjunto de teste do subconjunto Toyota[1.827].

Ainda nesse contexto, destaca-se a classe *WatchTV* que apresentou confusões significativas com diversas outras ações, como *Sitdown*, *Readbook*, e com atividades de uso de dispositivos eletrônicos (*Uselaptop*, *Usetablet*). Há ocorrência de confusão até mesmo

com as classes de ação de ingestão de bebidas (*Drink* e suas variações), também devido à postura semelhante adotada pelo indivíduo. Além disso, a natureza concorrente de algumas atividades contribui para essas confusões, já que é possível realizar outras ações simultaneamente à observação da televisão, e o modelo, ao se basear exclusivamente em informações de pose sem considerar o contexto ou objetos presentes na cena, não consegue discernir essas situações.

Com a implementação da divisão estrita de 80%/20% no subconjunto Toyota[1.922], observou-se uma maior convergência entre a acurácia global e a acurácia média por classe. Mesmo com a classe *Walk* ainda superando o teto inicialmente planejado, a aproximação entre essas métricas indicou um aprendizado mais equilibrado. A análise da matriz de confusão apresentada na Figura 6.4 reforçou que o desbalanceamento não era o principal fator limitante do desempenho do modelo nesse estágio. Embora *Walk* possuísse o maior volume de dados, ela não concentrava confusões relevantes com outras classes.

Como ajuste final, foi realizada uma correção pontual no conjunto de dados com o objetivo de viabilizar uma divisão dos subconjuntos de treino, validação e teste segundo proporções mais exatas, e de aproximar, tanto quanto possível, uma distribuição uniforme entre as classes. Para isso, adotou-se um teto máximo de 40 amostras por classe, incluindo o ajuste da classe *Walk*, anteriormente com maior número de instâncias. Embora a disponibilidade de dados tenha impedido que todas as classes atingissem esse limite, resultando em uma sub-representação residual em algumas categorias, o subconjunto final obtido (Toyota[1.753]) permitiu a condução dos experimentos sob condições mais próximas do cenário ideal. Nesse contexto, o modelo alcançou acurácia global e acurácia média idênticas, ambas de 49,7%, eliminando a diferença entre essas métricas e garantindo maior comparabilidade entre as classes.

A Figura 6.5 apresenta a matriz de confusão normalizada do modelo treinado com o subconjunto Toyota[1.753]. Apesar da queda na acurácia global, em comparação com o modelo treinado com o subconjunto Toyota[1.922], observa-se uma melhoria no desempenho por classe, evidenciada pelo aumento da acurácia na diagonal principal em 10 das classes avaliadas. O ganho mais expressivo foi observado na classe *Cutbread*, cuja acurácia de 20% no subconjunto anterior foi para 60% no subconjunto Toyota[1.753].

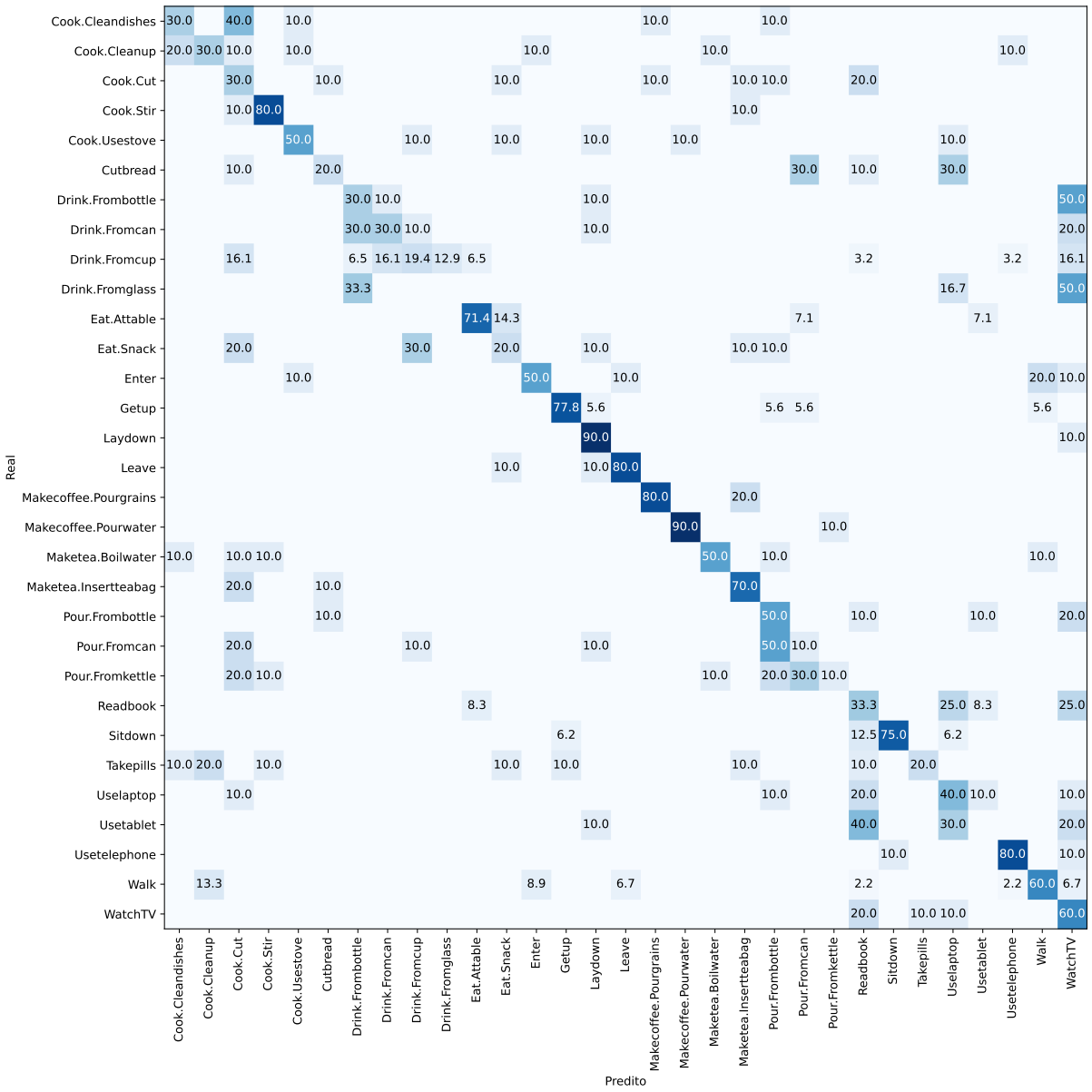


Figura 6.4: Matriz de Confusão do conjunto de teste do subconjunto Toyota[1.922].

Outras classes também apresentaram ganhos relevantes, como *Enter* de 50% para 70% e *Cook.Cleanup* de 30% para 60%. A classe *Laydown* chegou a atingir 100% de acurácia.

Verificou-se também uma redução em confusões específicas entre classes semanticamente semelhantes. No subconjunto anterior, a classe *Cutbread* apresentava confusão de aproximadamente 20% com *Cook.Stir*. De forma semelhante, a confusão entre *Drink.Frombottle* e *Drink.Fromglass* com a classe *Walk* reduziu-se de 50% para 30%, indicando uma melhor separação entre padrões de consumo de bebidas e movimentos de locomoção. Observa-se ainda uma ligeira redução da confusão entre ações envolvendo o uso de dispositivos, como *Usetablet* e *Uselaptop*. Não houve aumento de confusão sig-

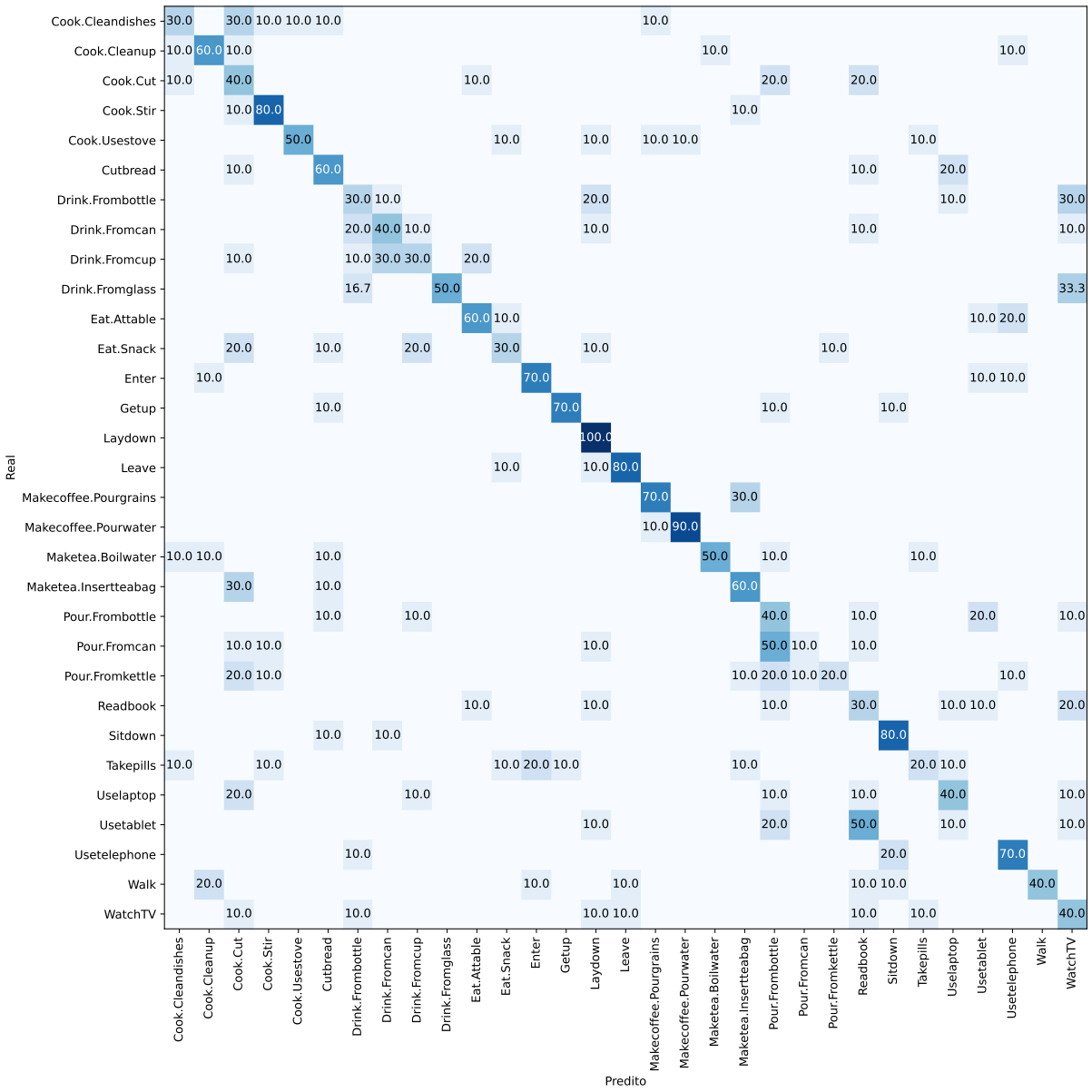


Figura 6.5: Matriz de Confusão do conjunto de teste do subconjunto preliminar Toyota[1.753].

nificativo (igual ou acima de 20 p.p.) entre as classes. Esses resultados sugerem que a estratégia adotada no Toyota[1.753] contribuiu para um aprendizado mais equilibrado entre classes.

A evolução dos experimentos indica que, nos estágio inicial o desempenho do modelo foi fortemente impactado pelo número reduzido de dados disponíveis e pelo elevado desbalanceamento entre as classes. Nessas condições, observou-se uma diferença expressiva entre a acurácia global e a acurácia média (19,2 p.p.), bem como dificuldades na aprendizagem de representações discriminativas para categorias menos representadas, resultando inclusive em predições ausentes para algumas classes. A partir do subconjunto

Toyota[1.827], o modelo passou a apresentar um comportamento mais estável, evidenciado pela consolidação de uma matriz de confusão completa e pela redução consistente da diferença entre as acurácias global e média (de 1,9 p.p para 0,0 p.p.). Nas condições desse experimento, as limitações remanescentes deixaram de estar predominantemente associadas à escassez de dados ou ao desbalanceamento extremo, passando a refletir, sobretudo, a elevada similaridade semântica entre determinadas ações, em especial aquelas que se diferenciam apenas pelo objeto de interação. Nesse contexto, o subconjunto Toyota[1.753] representa um ponto de equilíbrio relevante. Embora não maximize a acurácia global, ele fornece uma avaliação mais justa e informativa da capacidade do modelo, ao assegurar desempenho uniforme entre classes e minimizar distorções causadas por distribuições assimétricas. Esse cenário foi, portanto, adotado como base para as análises subsequentes de configuração e para a interpretação dos limites do PoseConv3D no conjunto de dados TST.

6.2.3 Análise de hiperparâmetros

Com o objetivo de investigar o impacto de diferentes configurações de treinamento no desempenho do modelo, foram conduzidos experimentos no subconjunto Toyota[1.753], definido como *baseline* experimental a partir dos experimentos anteriores. Os experimentos contemplaram variações na estratégia de pré-treinamento, taxa de aprendizado, número de vídeos processados por GPU e quantidade de épocas de treinamento.

Embora a análise central dos resultados se dê em torno da rede X3D, definida como *backbone* principal deste trabalho, a arquitetura C3D Light (LIN et al., 2014b), também foi avaliada de forma breve. Esta rede foi utilizada como um dos modelos de referência no estudo original do PoseConv3D, também com uma versão adaptada implementada e igualmente validada no conjunto NTU RGB+D pelos autores. A inclusão da C3D Light teve como motivação técnica a investigação de um compromisso alternativo entre custo computacional e capacidade representacional. A rede era a segunda mais leve entre as redes que Duan et al. (2022) utilizaram no trabalho original em termos de quantidade de parâmetros e FLOPs. Sua estrutura apresenta maior profundidade quando comparada à X3D nas configurações adotadas, o que poderia, em princípio, favorecer a mo-

delagem de padrões espaço-temporais mais complexos, ainda que com aumento moderado no custo de treinamento. Dessa forma, a arquitetura foi incorporada aos experimentos com o objetivo de verificar se esse acréscimo estrutural seria capaz de competir com a X3D no contexto do conjunto Toyota[1.753]. Assim como a X3D, a C3D Light foi avaliada com e sem inicialização a partir de pesos pré-treinados no NTU RGB+D, mantendo-se os demais parâmetros fixos.

As taxas de aprendizado analisadas foram 0,01 e 0,005. O número de vídeos por GPU corresponde ao tamanho do lote (*batch size*) por GPU, indicando quantas amostras de vídeo são processadas simultaneamente em cada iteração de treinamento. Os valores desse parâmetro foram 8 e 12. Para cada configuração, o treinamento foi conduzido por 24, 40 ou 60 épocas, e o desempenho foi avaliado por meio da acurácia global e da acurácia média, tanto no conjunto de validação quanto no conjunto de teste, e também a diferença absoluta entre os valores das métricas dos respectivos conjuntos (DIF). Adicionalmente, foi registrado o tempo total de treinamento de cada experimento, visando caracterizar o custo computacional associado às diferentes configurações.

Os resultados obtidos a partir dessas variações são apresentados nas Tabelas 6.5 e 6.6, servindo como base para a análise comparativa e para a definição da configuração adotada nos experimentos subsequentes. Nas tabelas em questão, "Exp" refere-se ao identificador do experimento (ou configuração) e "LR" refere-se à taxa de aprendizado (*learning rate*), e ressalta-se que os modelos X3D e C3D Light citados referem-se às implementações adaptadas pelos autores do PoseConv3D. A tabela também apresenta os tempos de treinamento em cada configuração.

Tabela 6.5: Configurações experimentais e desempenhos obtidos no conjunto Toyota[1.753].

Exp	**Modelo**	**Pré-treinamento**	**LR**	**Vídeos/GPU**	**Épocas**	**Validação (%)**		**Teste (%)**		**Duração**
						Acc. Global	**Acc. Média**	**Acc. Global**	**Acc. Média**	
1	X3D	Sim	0,01	8	40	60,9	52,9	48,4	47,7	5h
2	X3D	Sim	0,01	12	40	60,9	53,1	46,7	46,1	3h
3	X3D	Não	0,01	12	40	53,1	45,7	41,8	42,2	3h
4	X3D	Sim	0,005	12	40	59,8	53,6	**50,3**	**49,7**	5h
5	X3D	Sim	0,005	12	60	59,2	51,7	49,7	49,3	7h
6	C3D Light	Sim	0,01	8	24	59,8	52,6	48,4	48,2	8h20
7	C3D Light	Não	0,01	8	24	59,8	54,9	45,8	45,2	8h30
8	C3D Light	Sim	0,005	8	40	57,0	50,7	46,7	46,1	14h

No que se refere à taxa de aprendizado, nos Experimentos 2 e 4, observa-se que ambas as configurações convergem de forma estável ao longo das 40 épocas para as taxas

Tabela 6.6: Diferença absoluta entre acurácias de validação e teste (em pontos percentuais) para diferentes configurações no conjunto Toyota[1.753].

Exp	Modelo	Pré-treinamento	LR	Vídeos/GPU	Épocas	Validação / Teste (p.p.)	
						DIF Acc. Global	DIF Acc. Média
1	X3D	Sim	0,01	8	40	12,52	5,11
2	X3D	Sim	0,01	12	40	14,16	6,98
3	X3D	Não	0,01	12	40	11,24	3,52
4	X3D	Sim	0,005	12	40	**9,45**	**3,89**
5	X3D	Sim	0,005	12	60	9,55	2,49
6	C3D Light	Sim	0,01	8	24	11,40	4,39
7	C3D Light	Não	0,01	8	24	14,03	9,71
8	C3D Light	Sim	0,005	8	40	10,25	4,54

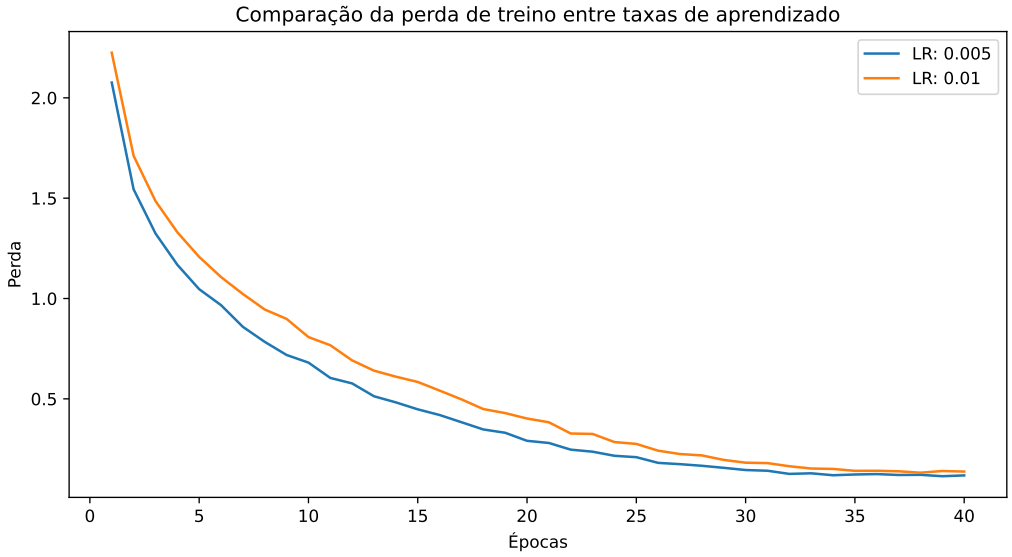
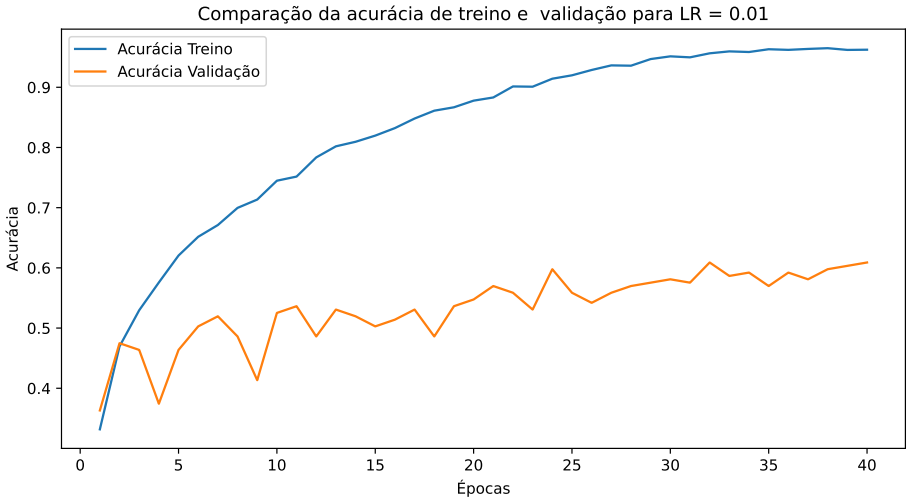


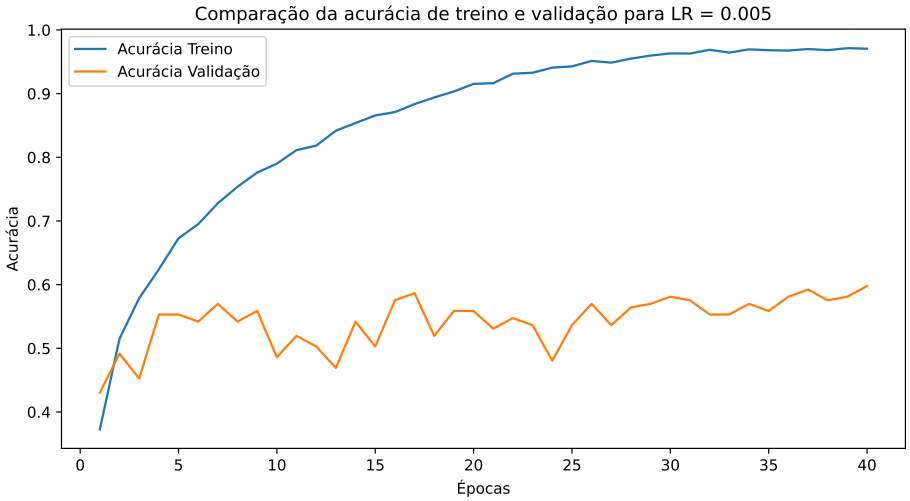
Figura 6.6: Curvas de perda de treino com taxas de aprendizado 0,01 (Experimento 1) e 0,005 (Experimento 4).

0,01 e 0,005 (Figuras 6.6 e 6.7). No entanto, no experimento conduzido com taxa de aprendizado igual a 0,01 esse comportamento não se refletiu em melhor desempenho no conjunto de teste, sendo observada uma queda de aproximadamente 3,6 p.p. na acurácia global e de 2 p.p. na acurácia média em relação às configurações com taxa de aprendizado de 0,005. Embora a taxa de aprendizado 0,01 tenha produzido acurácias elevadas no conjunto de validação (global de 60,9% e média 52,9%), a discrepância em relação ao conjunto de teste, evidenciada por um uma diferença de 14,16 p.p. na acurácia global e 6,98 p.p. na acurácia média, sugere um processo de ajuste excessivo dos pesos (*overfitting*), comprometendo a capacidade de generalização do modelo. Em contraste, a taxa de aprendizado de 0,005 apresentou convergência mais rápida nas primeiras épocas, além de melhor equilíbrio entre os resultados de validação e teste, reduzindo a discrepância entre a acurácia global (9,45%) e a acurácia média por classe (3,89%). Esse padrão de

desempenho entre as duas taxas de aprendizado se repete nos demais experimentos com o modelo X3D apresentados na tabela.



Curvas de acurácia de treino e validação para o experimento 2 (taxa de aprendizado 0,01).



Curvas de acurácia de treino e validação para o experimento 4 (taxa de aprendizado 0,005).

Figura 6.7: Curvas de acurácia de treino e validação para os experimentos 2 e 4 com diferentes taxas de aprendizado.

A análise do impacto do tamanho do lote por GPU nos Experimentos 1 e 2, indica que a configuração com 12 vídeos apresentou convergência mais rápida nas primeiras épocas em relação à de 8 vídeos (Figura 6.8). Esse comportamento é evidenciado pela queda mais acentuada da perda de treino nas primeiras iterações, assim como por valores de perda consistentemente menores ao longo de todo o processo de otimização. Tais resultados podem ser compreendidos a partir da relação entre o tamanho do lote, o número

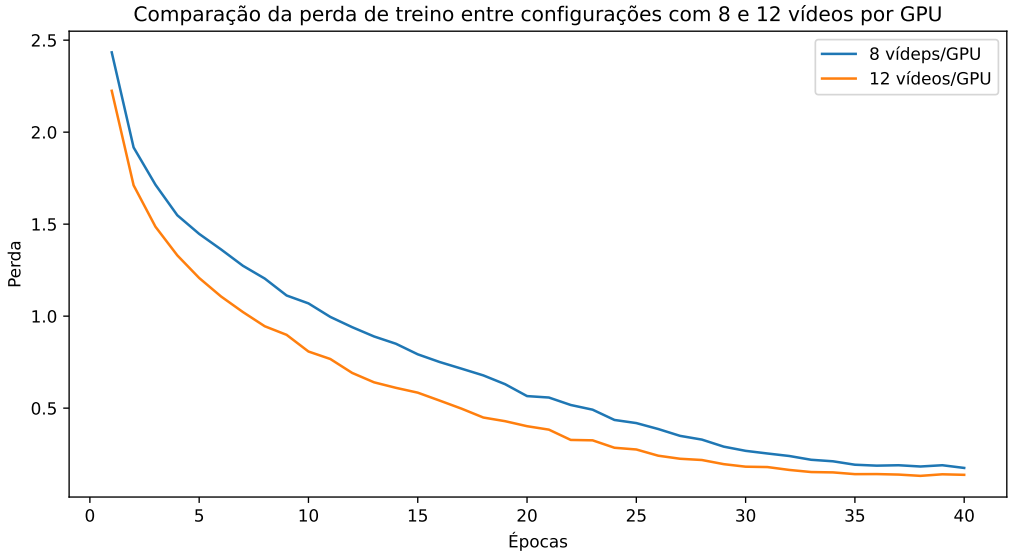


Figura 6.8: Curva de perda de treino com as configurações de 8 e 12 vídeos por GPU (Experimentos 1 e 2, respectivamente).

de iterações por época e a taxa de aprendizado.

Como formalizado na Equação 6.1 onde N_{iter} é o número total de iterações durante o treinamento; E é o número de épocas; N é o número total de amostras no conjunto de dados; B é o tamanho do lote (*batch size*), para um conjunto de dados fixo, o aumento do tamanho do lote reduz o número de atualizações de pesos realizadas a cada época, tornando as estimativas de gradiente menos ruidosas. Dessa forma, quando combinado com uma taxa de aprendizado relativamente elevada, esse regime de otimização permite passos maiores no espaço de parâmetros que permanecem mais estáveis, acelerando a convergência inicial da função de custo.

$$N_{\text{iter}} = E \times \left\lceil \frac{N}{B} \right\rceil \tag{6.1}$$

Entretanto, ao confrontar esse comportamento com as métricas de desempenho, observa-se que a convergência mais rápida e a menor perda de treino não se traduziram em melhor capacidade de generalização. Conforme os dados apresentados na Tabela 6.5, ambas as configurações atingiram o mesmo valor de acurácia global no conjunto de validação (60,89%). No conjunto de teste, contudo, a configuração com 12 vídeos por GPU apresentou desempenho inferior, com redução da acurácia de 48,37% para 46,73%. Tendência semelhante é observada na acurácia média por classe, que apresentou leve aumento na

validação (de 52,85% para 53,11%), acompanhado por queda no teste (de 47,74% para 46,13%). Esse comportamento indica que a boa convergência e o ajuste acentuado no treinamento fizeram o modelo se especializar nos dados de treino, gerando sobreajuste. Nesse contexto, a redução da taxa de aprendizado mostrou-se eficaz para melhorar a generalização, pois passos menores nos ajustes de peso tornam o treinamento mais estável. Como pode ser observado na Tabela 6.5, no Experimento 4 o uso de uma taxa de aprendizado de 0,005 com 12 vídeos por GPU e mantendo-se a quantidade de épocas dos demais experimentos, houve aumento tanto na acurácia global (50,3%) quanto na acurácia média (49,7%) no conjunto de teste, além da menor diferença de acurácia global (9,45 p.p.).

Em relação ao número de épocas, os resultados dos Experimentos 4 e 5 indicam que a extensão do treinamento de 40 para 60 épocas não resultou em ganhos consistentes de desempenho. Pelo contrário, a configuração com 60 épocas apresentou uma leve queda tanto na acurácia de validação e teste quanto na acurácia média por classe, ao mesmo tempo em que implicou um aumento expressivo do custo computacional, passando de 5h para 7h de processamento. Esse comportamento sugere que o modelo já havia atingido um ponto de saturação por volta das 40 épocas, conforme evidenciado pela estabilização da curva de perda de treino na Figura 6.9, correspondente ao Experimento 5. A partir desse ponto, a continuidade do treinamento passou a provocar sobreajuste, sem benefícios adicionais em termos de generalização. Esse cenário é corroborado pela alta volatilidade e ausência de tendência de subida na acurácia de validação do Experimento 5 após a época 40, como exibido na Figura 6.10.

Quanto ao uso de pré-treinamento, a interpretação dos seus efeitos deve considerar não apenas a capacidade de generalização do modelo, mas também o custo computacional associado ao processo de treinamento. Ao comparar os Experimentos 2 e 3, observa-se que a configuração com pré-treinamento (experimento 2) apresenta desempenho significativamente superior no conjunto de teste, com acurácia global de 46,7% frente a 41,8% e acurácia média de 46,7% frente a 42,2%. Esse ganho pode ser atribuído ao fato de que o modelo pré-treinado é ajustado especificamente ao domínio do problema em estudo, refinando representações previamente aprendidas e aproveitando similaridades entre o conjunto NTU RGB+D e o conjunto Toyota Smarthome.

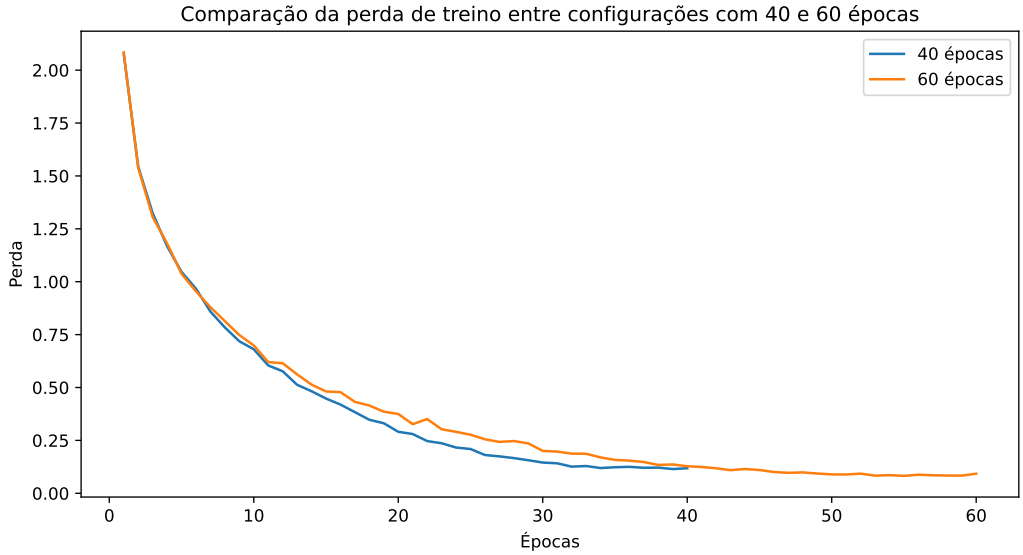


Figura 6.9: Curvas de perda de treino para 40 épocas (Experimento 4) e 60 épocas (Experimentos 5).

Em contrapartida, a diferença de acurácia global 14,16 p.p. frente à 11,24 p.p. e de acurácia média 6,98 frente à 3,52 p.p mostra-se ligeiramente maior na configuração com pré-treinamento. Esse comportamento, pode ser explicado pela dinâmica de convergência do treinamento, exibida na Figura 6.11. Na Figura observa-se que o modelo pré-treinado inicia o processo de otimização em níveis de perda mais baixos e converge de maneira mais rápida e estável, enquanto o treinamento do zero apresenta uma fase inicial prolongada de adaptação, evidenciando a necessidade de um maior número de épocas para atingir desempenho comparável. Quando o número de épocas é limitado, o treinamento sem aprendizado prévio pode não alcançar um regime suficientemente maduro para capturar de forma consistente as variações intra-classe, afetando principalmente a acurácia média.

Uma avaliação mais equilibrada foi possível nos experimentos 9 e 10, com configurações sem pré-treinamento treinadas com taxa de aprendizado na escala de 10^{-3} (0,005 e 0,001), estendidas em 80 e 160 épocas. Os resultados são apresentados na Tabela 6.7. Mesmo com aumento significativo do número de épocas, os valores máximos obtidos dos experimentos foram de 51,95% de acurácia global e acurácia média por classe de 46,79%, no conjunto de validação, valores consideravelmente inferiores aos obtidos pelas configurações equivalentes com pré-treinamento. Esses resultados indicam que o aumento do número de épocas, isoladamente, não é suficiente para compensar a ausência

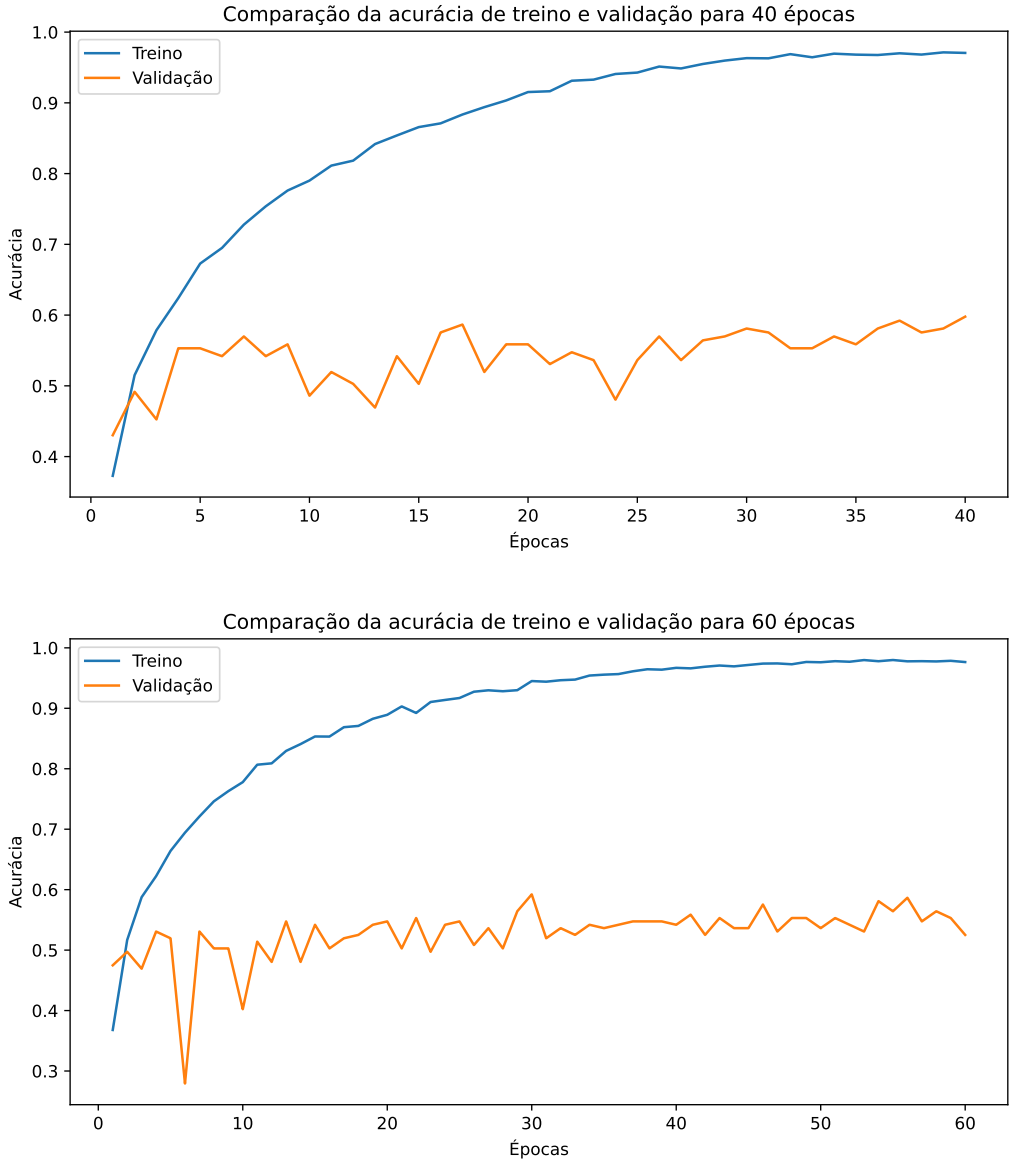


Figura 6.10: Curva de acurácia de treino e validação para 40 épocas (Experimento 4) e 60 épocas (Experimento 5).

de pré-treinamento, sendo ainda inviável em termos de custo computacional devido ao prolongado tempo de treinamento necessário para que o modelo comece a convergir.

Experimentos adicionais com taxa de aprendizado reduzida para 0,001 e até 160 épocas reforçam essa observação. Mesmo sob um regime de treinamento prolongado, o modelo não apresentou convergência adequada no conjunto de treino, atingindo aproximadamente 50% de acurácia, enquanto os valores no conjunto de validação permaneceram abaixo de 45%. Isso evidencia que a simples redução da taxa de aprendizado torna o processo de otimização excessivamente lento e computacionalmente inviável no contexto experimental considerado.

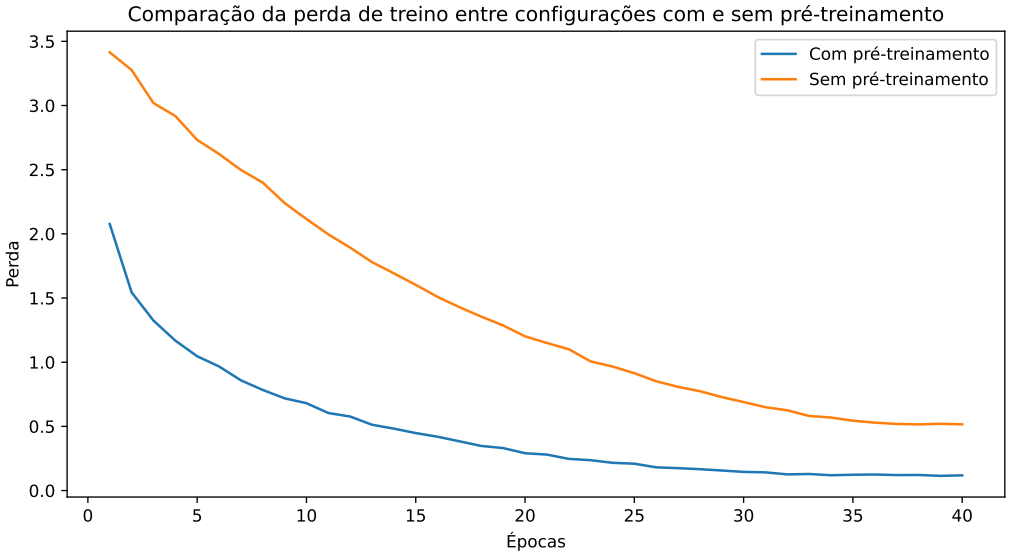


Figura 6.11: Curvas de perda do conjunto de treino para as configurações com e sem pré-treinamento (Experimentos 2 e 3 respectivamente).

Tabela 6.7: Resultados adicionais sem pré-treinamento considerando apenas treino e validação.

Exp	**Modelo**	**LR**	**Vídeos por GPU**	**Épocas**	**Acc. Treino**	**Acc. Validação**
9	X3D	0,005	24	80	93,75%	51,95% (global) 46,79% (média)
10	X3D	0.001	12	160	53,33%	43,01% (global) 37,44% (média)

Por outro lado, as configurações com pré-treinamento a partir do conjunto NTU RGB+D apresentaram convergência mais rápida e estável, alcançando níveis superiores de acurácia com um número reduzido de épocas e taxas de aprendizado moderadas. A melhor configuração obtida de todos os experimentos é a do Experimento 4 com taxa de aprendizado de 0,005, 12 vídeos por GPU e 40 épocas. O modelo atingiu 50,3% de acurácia global e 49,7% de acurácia média no conjunto de teste em apenas 5 horas de treinamento, apresentando também 53,6% de acurácia média e 59,8% de acurácia global no conjunto de validação. Além disso, essas configurações resultaram nos menores valores de DIF entre treinamento e validação, indicando melhor equilíbrio entre desempenho e capacidade de generalização. O uso de pré-treinamento configura-se, portanto, como a alternativa de melhor custo-benefício para o treinamento do Pose-X3D-S no conjunto TST.

Ao comparar os *backbones* avaliados, o X3D apresentou desempenho superior ao

C3D Light na maior parte das configurações experimentais, especialmente quando combinado com pré-treinamento e taxa de aprendizado reduzida. Embora o C3D Light tenha alcançado acurácias competitivas no conjunto de validação nos Experimentos 6 e 7, inclusive comparáveis à melhor configuração do X3D, seu desempenho no conjunto de teste permaneceu inferior. Além disso, o C3D Light exigiu tempos de treinamento significativamente maiores, em alguns casos mais que o dobro, sem apresentar ganhos proporcionais de desempenho, comprometendo sua viabilidade prática no contexto avaliado.

Dessa forma, a rede X3D se consolidou como a melhor opção, e sua configuração mais eficiente foi adotada como base para os experimentos subsequentes, por oferecer o melhor equilíbrio entre desempenho, estabilidade do aprendizado e viabilidade computacional, sendo particularmente adequada ao cenário de reconhecimento de ações humanas baseado exclusivamente em informações de pose.

6.2.4 Experimentos sobre o conjunto completo

Os experimentos conduzidos com o conjunto completo de dados são centrais neste trabalho, uma vez que a comparação com resultados reportados na literatura pressupõe a utilização de todo o volume de dados disponível. Assim, após a validação inicial do *pipeline* experimental em subconjuntos reduzidos, o treinamento e a avaliação do modelo foram estendidos ao conjunto completo, respeitando o protocolo original definido para o conjunto e citado na Seção 5.3.

Nestes experimentos o *backbone* X3D (Pose-X3D-S) foi adotado, inicializado a partir de pesos previamente ajustados no conjunto NTU RGB+D, caracterizando um cenário de pré-treinamento sobre um domínio distinto. O treinamento foi realizado com taxa de aprendizado igual a 0,005, utilizando 32 vídeos por GPU e um total de 20 épocas. O aumento na quantidade de vídeos por GPU em comparação aos experimentos anteriores foi possibilitado pela atualização da placa de vídeo, citada na introdução deste capítulo. Não foram aplicadas, nesse estágio, estratégias adicionais de balanceamento de classes ou aumentos de dados direcionados, sendo mantida a distribuição original do conjunto de treinamento. Os conjuntos de validação e teste também permaneceram inalterados, assegurando a consistência da avaliação. Os resultados obtidos neste experimento são

apresentados na Tabela 6.8 e serviram de referência para as análises e ajustes experimentais subsequentes.

Tabela 6.8: Resultados da aplicação do *framework* PoseConv3D ao conjunto Toyota Smarthome Trimmed completo.

Validação (%)		**Teste (%)**			**Duração**
Acc. Global	**Acc. Média**	**Acc. Global**	**Acc. Média**	**DP**	
77,12	52,14	72,26	54,51	25,88	12h

Observa-se que o modelo atingiu 77,12% de acurácia global em validação, indicando boa capacidade de ajuste aos dados vistos durante o treinamento. No entanto, a acurácia média por classe foi de 52,14% e desvio padrão de acurácia média por classe de 25,88%, evidenciando que o desempenho varia significativamente entre classes, principalmente devido ao desbalanceamento e à dificuldade de predizer ações com posturas semelhantes. No conjunto de teste, a acurácia global de 72,26% e a acurácia média de 54,51% sugerem que o modelo generaliza de forma razoável, mantendo desempenho consistente em dados não vistos. De forma geral, o resultado indica que o PoseConv3D consegue capturar padrões espaciais e temporais das ações, mas ainda apresenta limitações em atividades com posturas semelhantes ou ações concorrentes, refletindo a dificuldade inerente ao reconhecimento de ações puramente a partir de representações de pose. A análise qualitativa da matriz de confusão obtida no conjunto de teste (Figura 6.12) reforça essas observações, evidenciando que as confusões entre classes seguem padrões semelhantes aos já identificados.

Especificamente, na atividade *Drink.Fromcup*, observou-se que seu elevado número de amostras no conjunto de treinamento introduzia um viés amostral. A classe *WatchTV*, como já destacado em análises dos subconjuntos preliminares, mesmo em menor quantidade de amostras apresentou baixa variabilidade das poses associadas à ação, caracterizada por movimentos sutis e postura predominantemente estática, o que resultou em dificuldades de generalização do modelo. De modo semelhante, a classe *Readbook* também manteve grande confusão com diversas classes que o indivíduo realiza sentado e, ainda, interagindo com um objeto como *Usetablet*, *Uselaptop*, *Eat.Attable* entre outras.

Na tentativa de mitigar esses problemas, foram avaliadas diferentes estratégias de pré-processamento aplicadas exclusivamente ao conjunto de treinamento, com o obje-

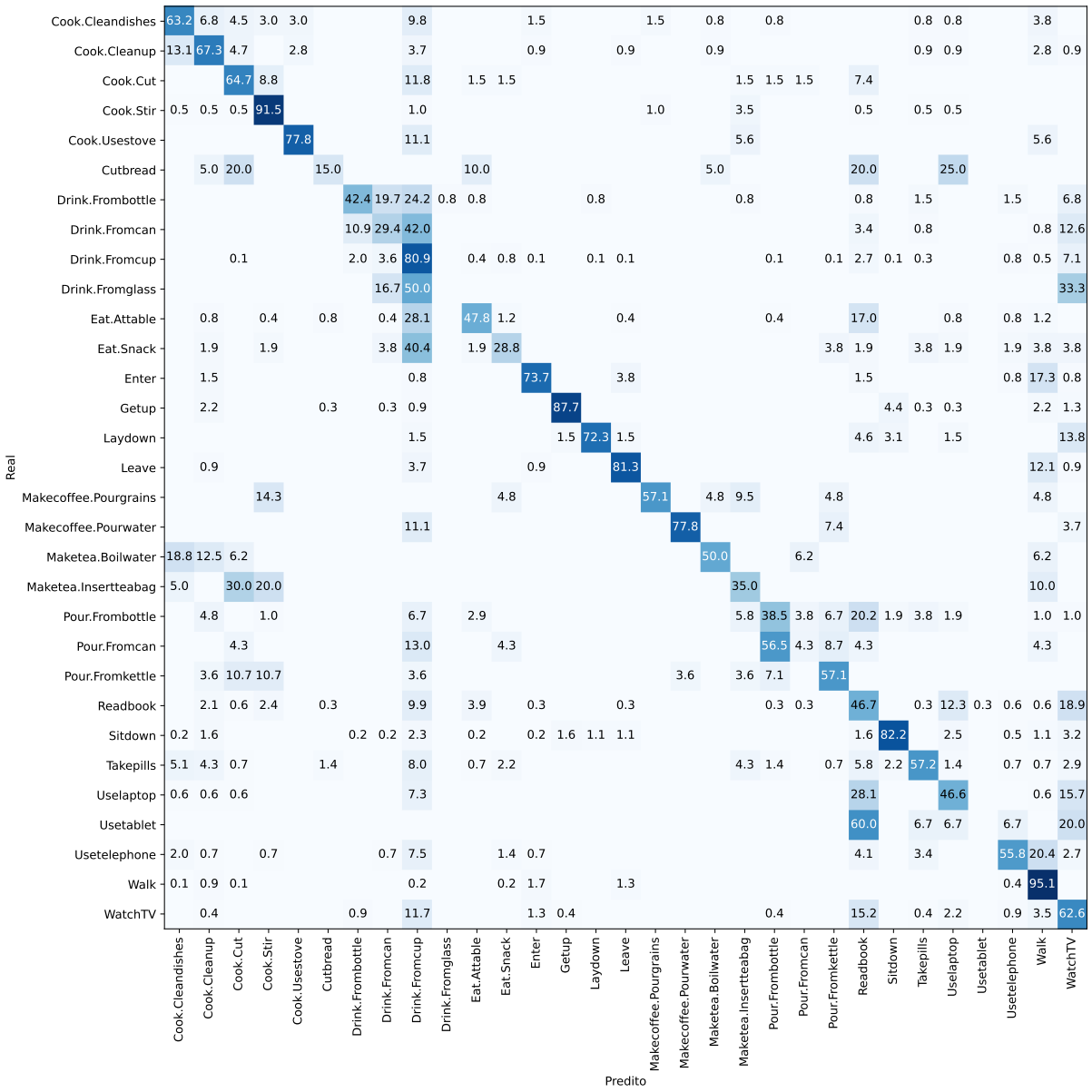


Figura 6.12: Matriz de confusão do conjunto de teste do TST completo.

tivo de reduzir os efeitos do desbalanceamento entre classes e aumentar a variabilidade das ações menos representadas. As estratégias investigadas incluíram: (1) ajuste da distribuição das classes mais frequentes e com maior grau de confusão, em particular *Drink.Fromcup*, *Readbook* e *WatchTV*; e (2) a imposição de um limite máximo de 400 amostras por classe.

O conjunto de treino apresenta uma média de aproximadamente 215 amostras por classe, contudo, devido ao forte desbalanceamento da distribuição, essa média não é um indicador robusto da representatividade real das classes, uma vez que a maioria delas possui menos de 400 amostras, enquanto apenas seis classes atingem ou ultrapassam esse

valor. Além disso, observa-se que o desbalanceamento é fortemente concentrado em duas classes dominantes(*Drink.Fromcup* e *Walk*), com mais de 1000 amostras cada. Embora a limitação do número de amostras por classe reduza o viés associado a essas classes majoritárias, uma redução excessiva poderia comprometer classes já bem definidas pela maior disponibilidade de dados, levando à perda de variabilidade intra-classe. Nesse contexto, o teto de 400 amostras foi definido como um compromisso entre a mitigação do desbalanceamento e a preservação da representatividade estatística das classes mais frequentes, afetando principalmente as classes dominantes e mantendo a distribuição original da maioria das demais.

Paralelamente à aplicação das duas estratégias, foi adotada uma estratégia de aumento de dados baseada exclusivamente em transformações geométricas aplicadas às próprias amostras do conjunto de treino, sem a introdução de novas instâncias independentes ou dados sintéticos externos. Essas transformações foram direcionadas exclusivamente às classes sub-representadas no conjunto de treinamento. A seleção dessas classes foi fundamentada em uma análise quantitativa da métrica de precisão por classe, associada à distribuição de amostras nos conjuntos de treino, validação e teste, conforme apresentado na Tabela 6.9. A precisão mede quão confiáveis são as predições do modelo para uma classe específica, considerando apenas as amostras classificadas como pertencentes a essa classe. Com base nesses critérios, foram identificadas como classes críticas *Cutbread*, *Drink.Fromglass*, *Maketea.Insertteabag*, *Pour.Fromcan* e *Usetablet*, caracterizadas por baixos valores de precisão e, em alguns casos, pela escassez ou ausência de amostras no conjunto de validação. O objetivo principal dessa estratégia foi mitigar o impacto da limitada representatividade dessas classes e ampliar a variabilidade intra-classe, favorecendo um processo de aprendizado mais robusto.

Tabela 6.9: Precisão por classe crítica e distribuição de amostras nos conjuntos de treino, validação e teste.

ID	**Classe**	**Precisão (%)**	**Treino**	**Validação**	**Teste**
6	*Cutbread*	33,33	23	2	20
10	*Drink.Fromglass*	0,00	40	19	6
20	*Maketea.Insertteabag*	21,88	30	6	20
22	*Pour.Fromcan*	12,50	34	2	23
28	*Usetablet*	0,00	34	0	15

Os aumentos adotados foram pensados para atuação direta sobre a estrutura geométrica das sequências de pose, permitindo simular variações realistas na execução das atividades humanas monitoradas. O primeiro tipo de aumento consiste na aplicação de rotações aleatórias de pequena magnitude, limitadas a até 15 graus, sobre o esqueleto humano. Essa transformação busca representar variações naturais na orientação corporal do indivíduo em relação à câmera, comuns em cenários reais de monitoramento doméstico, nos quais o posicionamento do sujeito raramente é perfeitamente alinhado ao plano de captura. Ao introduzir essa variabilidade, o modelo é incentivado a aprender representações mais invariantes à orientação espacial da pose.

O segundo aumento corresponde à variação de escala da pose, na qual todas as articulações são ampliadas ou reduzidas de forma proporcional. Essa operação simula diferenças na distância entre o indivíduo e a câmera, bem como variações antropométricas entre sujeitos distintos. Do ponto de vista do aprendizado, esse aumento reduz a dependência do modelo em relação a dimensões absolutas do esqueleto, favorecendo a captura de relações espaciais relativas entre as articulações ao longo do tempo.

Adicionalmente, foi introduzida uma perturbação controlada nas coordenadas dos pontos-chave, caracterizada pela adição de ruído de baixa intensidade. Esse tipo de aumento visa modelar imprecisões inerentes aos algoritmos de estimação de pose, que podem ocorrer em função de oclusões, variações de iluminação ou ruído visual. Ao expor o modelo a essas perturbações durante o treinamento, busca-se aumentar sua robustez a pequenas inconsistências nas entradas, reduzindo o risco de *overfitting* a configurações articulares específicas.

Todas as operações de aumento de dados foram aplicadas exclusivamente ao conjunto de treinamento, preservando a integridade dos conjuntos de validação e teste. Essa estratégia permitiu ampliar a diversidade das classes sub-representadas sem introduzir viés artificial na avaliação do desempenho do modelo. Os resultados obtidos para cada estratégia estão apresentados na Tabela 6.10.

A introdução da estratégia de aumento de dados em classes críticas (Tabela 6.9), aliada ao balanceamento das classes *Drink.Fromcup*, *Readbook* e *WatchTV*, foi inicialmente avaliada com 30 épocas de treinamento. Observou-se uma leve redução na acurácia

Tabela 6.10: Resultados obtidos ao aplicar diferentes estratégias de correção de desbalanceamento e aumento de dados no conjunto de treino Toyota Smarthome Trimmed.

Estratégia	**Validação (%)**		**Teste (%)**		**Duração**
	Acc. Global	**Acc. Média**	**Acc. Global**	**Acc. Média**	
1	74,53	54,35	71,67	55,22	18h
2	71,83	50,15	69,00	55,21	13h

de validação, que passou para 74,53%, acompanhada por acurácia média de 54,35%. No conjunto de teste, o desempenho mostrou-se estabilizado, com acurácia global de 71,67% e acurácia média por classe de 55,22%, superando a acurácia média obtida sem balanceamento. Destaca-se que, nessa configuração, a acurácia de teste atingia 55,05% já em 20 épocas, indicando convergência mais precoce. O prolongamento do treinamento para 40 épocas, mantendo a mesma estratégia de aumento de dados, não resultou em ganhos substanciais, sugerindo saturação do desempenho.

A comparação entre as matrizes de confusão do conjunto de teste para o TST completo (Figura 6.12) e para o modelo com a estratégia 1 de balanceamento (Figura 6.13) evidencia que os padrões gerais de confusão entre classes foram amplamente preservados. De forma complementar, observa-se que o balanceamento proporcionou ganhos significativos em algumas classes, enquanto outras mantiveram acertos baixos, especialmente aquelas semanticamente ou biomecanicamente semelhantes, indicando que o aumento de dados e redistribuição das amostras não elimina completamente confusões entre ações similares.

Entre as classes críticas houve ganho apenas na classe *Usetablet*, que passou de 0% para 6,7% de acertos. Por outro lado, a classe *Cutbread* apresentou uma redução no desempenho, com queda de 15% para 10% de acertos, enquanto *Maketea.Insertteabag* manteve-se estável em 35%. A classe *Pour.Fromcan* teve o número de acertos reduzido a 0. Além disso, classes como *Drink.Fromglass* continuaram sem acertos (0%), permanecendo fortemente confundidas com ações semanticamente próximas que dispõem de maior número de amostras, como *Drink.Fromcup*. Esses resultados indicam que, embora o balanceamento contribua para melhorar o desempenho de algumas classes sub-representadas, ele não é suficiente para resolver ambiguidades inerentes a ações com poucos exemplos ou elevada similaridade semântica e biomecânica.

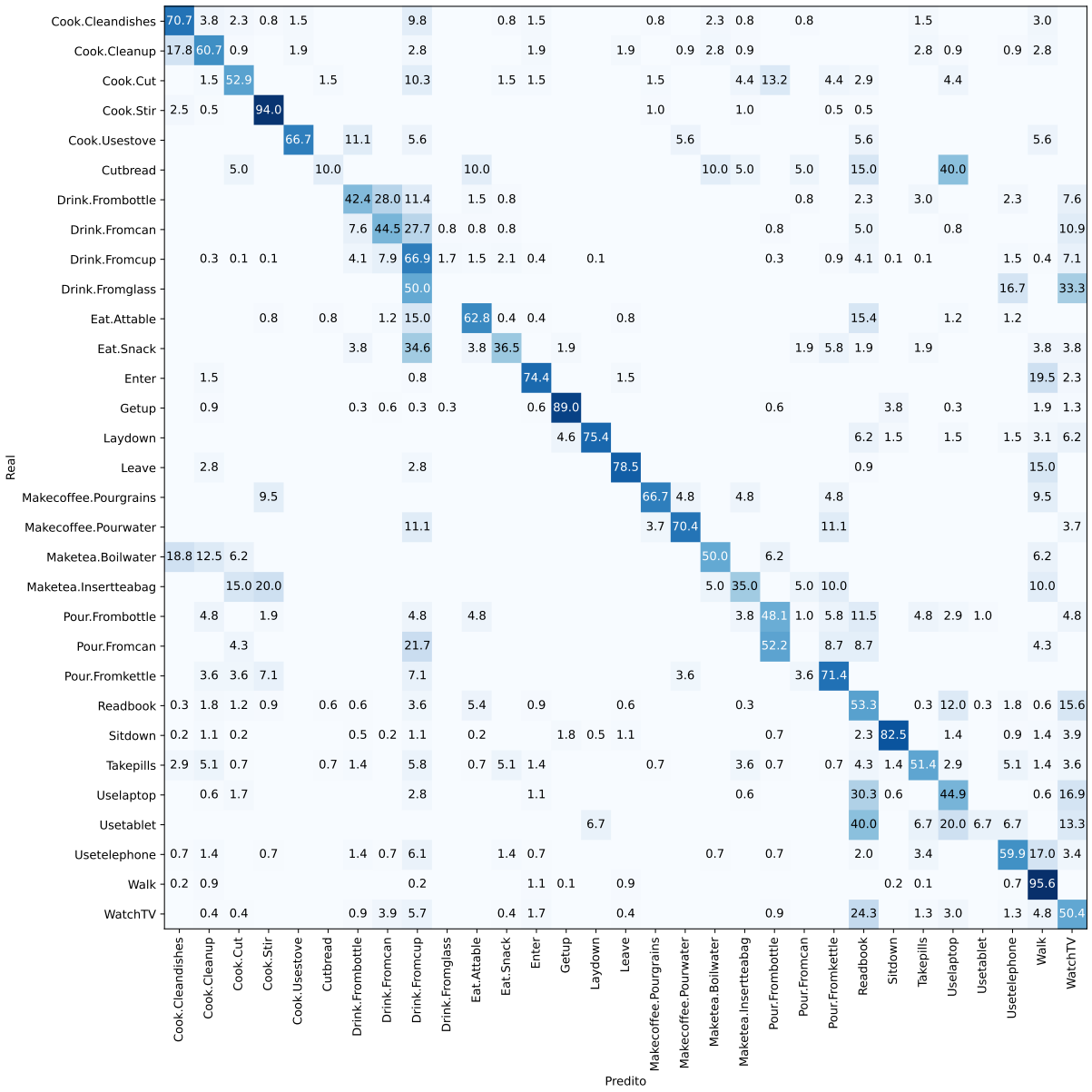


Figura 6.13: Matriz de confusão do conjunto de teste do TST para o modelo com estratégia 1 de balanceamento.

No geral, o maior ganho absoluto ocorreu na classe *Eat.Attable*, com aumento de 38 amostras corretamente classificadas, correspondendo a um ganho relativo de 15,02%. Outras classes também apresentaram ganhos relevantes, como a redução das confusões entre *Drink.Fromcup* e *Drink.Fromcan* (+32 amostras, 4,28%), *Readbook* (+22 amostras, 6,59%), bem como melhorias na separação entre *WatchTV* e *Readbook* (+21 amostras, 9,13%) e na classe *Drink.Fromcan* (+18 amostras, 15,13%). Esses resultados indicam que a estratégia favoreceu o aprendizado de classes menos representadas e contribuiu para uma melhor distinção entre ações semanticamente próximas.

Por outro lado, algumas classes apresentaram perdas consideráveis. A maior

queda foi observada em *Drink.Fromcup*, com redução de 104 amostras corretamente classificadas (-13,92%). Também foram observadas intensificações de confusão envolvendo *Eat.Attable* e *Drink.Fromcup* (-33 amostras, -13,04%), bem como em *WatchTV* (-28 amostras, -12,17%), *Readbook* (-21 amostras, -6,29%) e *Drink.Frombottle* (-17 amostras, -12,88%), todas associadas a confusões com *Drink.Fromcup*. Esses resultados mostram que, ao reduzir a predominância de algumas classes no treinamento, ocorreu uma redistribuição dos erros, com aumento de confusões entre ações que compartilham posturas e contextos semelhantes.

Em síntese, a estratégia 1 de balanceamento promove melhorias expressivas em classes críticas, especialmente em *Eat.Attable* e *Drink.Fromcan*, favorecendo a representação de ações menos frequentes. Simultaneamente, observa-se a ocorrência de compensações naturais, com aumento de confusões em classes que anteriormente eram bem reconhecidas, evidenciando que o balanceamento atua como um mecanismo de redistribuição do erro e equilíbrio do modelo, ainda que não elimine completamente ambiguidades entre classes postural ou semanticamente similares.

A estratégia 2 apresentou redução da acurácia de validação, atingindo 71,83%, bem como queda da acurácia média por classe para 50,15%. No conjunto de teste, entretanto, a acurácia global manteve-se em 69,00%, com acurácia média de 55,21%. Esses resultados indicam que a restrição no número de amostras por classe limita a capacidade de ajuste do modelo durante o treinamento, mas não compromete de forma significativa o desempenho no teste, sugerindo um efeito regularizador implícito. A Tabela 6.11 apresenta a distribuição de amostras por classe após a aplicação do teto global, mostrando que classes dominantes como *Drink.Fromcup, Readbook, Sitdown* e *Walk* foram drasticamente reduzidas, enquanto classes médias e raras foram preservadas. O total de amostras de treino caiu de 8.829 para 5.920, mantendo-se os conjuntos de validação e teste inalterados para garantir comparabilidade.

A comparação entre as matrizes de confusão do balanceamento 2 (Figura 6.14) e do balanceamento 1 (Figura 6.13) evidencia alterações relevantes na distribuição dos erros do modelo. O maior ganho individual foi observado na redistribuição das predições da classe *Walk* para *Leave*, com um acréscimo de 60 amostras (4,85%), indicando maior

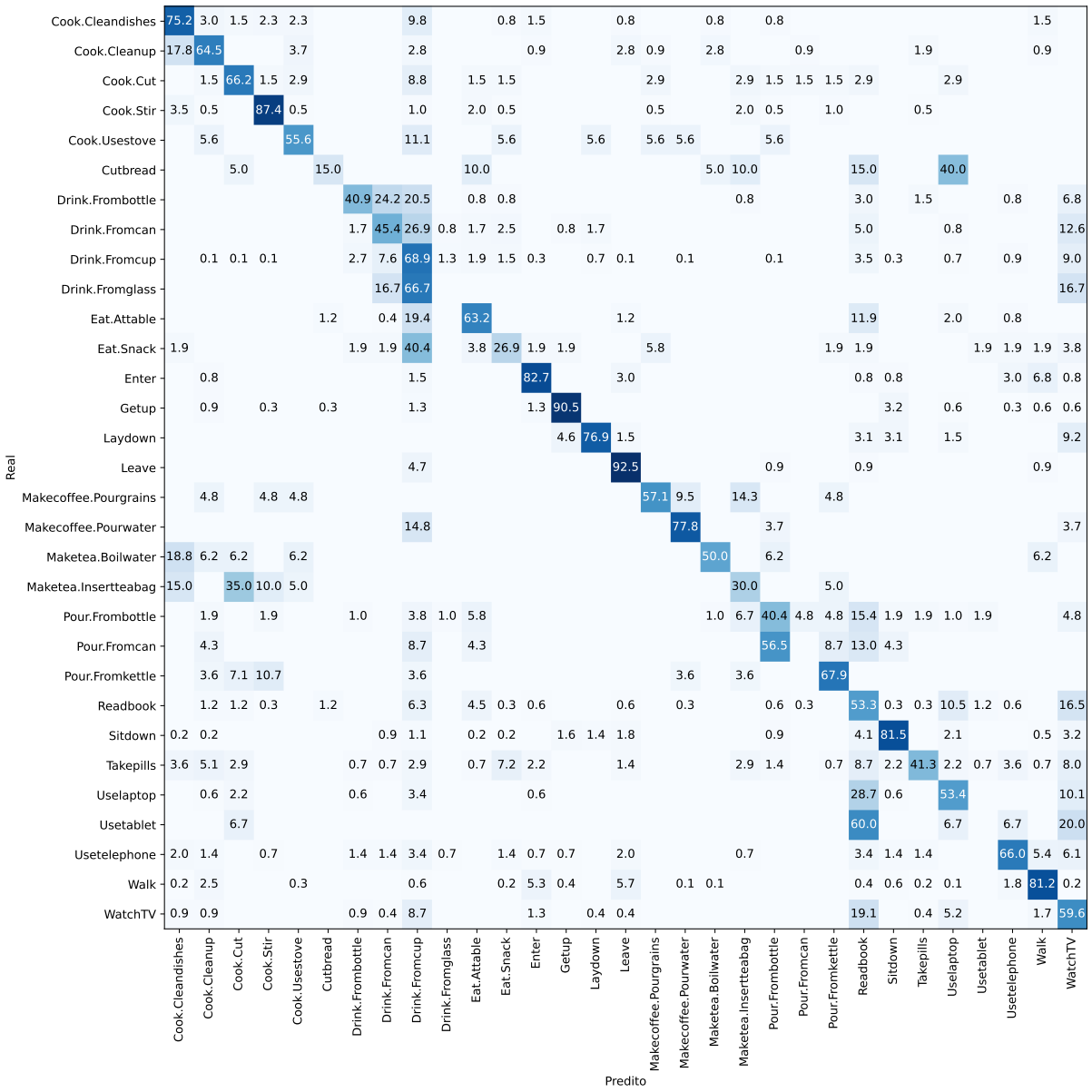


Figura 6.14: Matriz de confusão do conjunto de teste do TST para o modelo com estratégia 2 de balanceamento.

separação entre essas ações. Outros aumentos relevantes na redistribuição de erros incluem as confusões entre *Walk* e *Enter* (+53, 4,28%), e *Walk* e *Cook.Cleanup* (+20, 1,62%).

Em termos de acertos efetivos (diagonal da matriz), destacam-se os ganhos nas classes *WatchTV* (+21, 9,13%) e *Uselaptop* (+15, 8,43%).

Em contrapartida, a maior perda de desempenho foi concentrada na classe *Walk*, que apresentou redução de 178 amostras corretamente classificadas (-14,39%). Além disso, observou-se aumento das confusões de outras classes com *Walk*, incluindo *Usetelephone* (-17, -11,56%), *Enter* (-17, -12,78%) e *Leave* (-15, -14,02%). Por fim, a classe *Takepills* apresentou queda de desempenho, com redução de 14 acertos (-10,14%).

Tabela 6.11: Distribuição de amostras por classe após aplicação de teto global de 400 amostras no conjunto de treino.

Classe	Nome da Classe	Treino	Validação	Teste	Total
1	*Cook.Cleandishes*	225	20	133	378
2	*Cook.Cleanup*	254	19	107	380
3	*Cook.Cut*	93	17	68	178
4	*Cook.Stir*	300	80	199	579
5	*Cook.Usestove*	78	0	18	96
6	*Cutbread*	23	2	20	45
7	*Drink.Frombottle*	209	0	132	341
8	*Drink.Fromcan*	171	35	119	325
9	*Drink.Fromcup*	400	379	747	1526
10	*Drink.Fromglass*	40	19	6	65
11	*Eat.Attable*	333	31	253	617
12	*Eat.Snack*	140	24	52	216
13	*Enter*	282	29	133	444
14	*Getup*	400	78	317	795
15	*Laydown*	79	37	65	181
16	*Leave*	289	20	107	416
17	*Makecoffee.Pourgrains*	35	8	21	64
18	*Makecoffee.Pourwater*	41	8	27	76
19	*Maketea.Boilwater*	37	9	16	62
20	*Maketea.Insertteabag*	30	6	20	56
21	*Pour.Frombottle*	112	60	104	276
22	*Pour.Fromcan*	34	2	23	59
23	*Pour.Fromkettle*	69	10	28	107
24	*Readbook*	400	133	334	867
25	*Sitdown*	400	117	439	956
26	*Takepills*	177	29	138	344
27	*Uselaptop*	184	34	178	396
28	*Usetablet*	34	0	15	49
29	*Usetelephone*	251	53	147	451
30	*Walk*	400	521	1237	2158
31	*WatchTV*	400	73	230	703
Total		**5920**	**1853**	**5433**	**13206**

Nota: observa-se o desbalanceamento no número de amostras de treino, de modo que diversas classes não atingem o teto proposto.

Entre as classes críticas, observa-se ganho de acertos em *Cutbread*, de 10% para 15%, enquanto *Maketea.Insertteabag* apresentou redução de 35% para 30%. Por outro lado, *Drink.Fromglass* manteve-se com 0% de acertos, *Pour.Fromcan* continuou em 0% e *Usetablet* reduziu de 6,7% para 0%. Esses resultados indicam que novamente o balanceamento promoveu redistribuição de erros.

Em síntese, o balanceamento 2 atenuou parcialmente o viés das classes dominantes ao promover uma redistribuição dos erros entre classes funcionalmente semelhantes, alte-

rando os padrões de confusão observados no modelo. Embora a classe *Walk* tenha permanecido como um dos principais polos de erro, observou-se uma redistribuição das predições incorretas envolvendo essa classe, bem como ganhos pontuais de acurácia em classes específicas. Como consequência, verificou-se uma maior homogeneidade na acurácia média por classe, indicando um treinamento mais equilibrado. Ainda assim, algumas classes permaneceram desafiadoras, especialmente aquelas sub-amostradas ou semanticamente próximas, evidenciando limitações inerentes à representação baseada apenas em esqueletos 2D. Esse comportamento sugere que o balanceamento e as estratégias de aumento de dados atuam como mecanismos de regularização, melhorando o equilíbrio do treinamento sem eliminar completamente ambiguidades estruturais do problema.

6.2.5 Comparação com a Literatura

Para comparar os resultados obtidos com o uso do PoseConv3D, considera-se o modelo *Separable Spatio-Temporal Attention* (STA) (DAS et al., 2019), proposto pelos próprios autores do conjunto TST para lidar com os desafios específicos desse conjunto. O STA é guiado por pose 3D e também utiliza informações de aparência (RGB) acopladas a uma 3D-CNN. O modelo funciona acoplando um mecanismo de atenção sobre a 3D-CNN, utilizando as coordenadas 3D do esqueleto humano como entrada para uma LSTM de 3 camadas. A LSTM é uma arquitetura de rede neural recorrente (RNN) capaz de aprender dependências temporais nos dados. Neste caso, a rede direciona a atenção espacial e temporal de forma separada. A dissociação entre atenção espacial e temporal permite que o modelo concentre-se em regiões e momentos relevantes do vídeo, além de proporcionar maior robustez a mudanças de ângulo de câmera.

A Tabela 6.12 apresenta os resultados do modelo STA e os resultados do PoseX3D-S para o conjunto Toyota Smarthome Trimmed obtidos no presente trabalho. Para a comparação de resultados, reitera-se que ambos os modelos apresentados foram treinados sob o protocolo *Cross-Subject* definido para o conjunto TST e apresentado na Seção 4.2.1. Essa escolha assegura a utilização das mesmas listas de vídeos e das mesmas divisões de conjuntos de treinamento, validação e teste entre os trabalhos.

Em termos de custo computacional, o modelo STA apresenta uma complexidade

Tabela 6.12: Comparação de desempenho do conjunto de teste do Toyota Smarthome Trimmed na metodologia proposta neste trabalho com o método da literatura.

Modelo	**Entrada**	**Acurácia Global**	**Acurácia Média**
STA	Pose + RGB	75,3%	54,2%
STA	Pose somente	Não informado	42,5%
PoseConv3D - X3D (Este trabalho)	Pose somente	72,26%	54,51%

significativamente maior que a X3D. O STA exige pré-treinamento separado das redes base I3D (RGB) e LSTM (pose 3D), consumindo cerca de 23 horas, seguido de 5 horas adicionais para o treinamento ponta a ponta do mecanismo de atenção. Além disso, a utilização de múltiplas GPUs (4 GTX 1080 Ti) é necessária para viabilizar esse treinamento, refletindo um alto consumo de recursos de hardware. Em contraste, a X3D treinada apenas com informações de pose 2D em uma única GPU, totalizou 12 horas de treinamento para o conjunto completo, sem necessidade de pré-treinamento separado, demonstrando maior eficiência computacional. O maior gasto de tempo foi para extração de mapas de calor dos 16.115 vídeos. Essa diferença evidencia que, embora o STA possa alcançar maior acurácia ao combinar RGB e pose 3D, ele impõe custos computacionais consideravelmente superiores em comparação ao modelo baseado exclusivamente em pose. Além disso, apesar de o STA com Pose + RGB apresentar a maior acurácia global, a X3D alcançou uma acurácia média significativamente superior comparada ao STA utilizando apenas pose, e ficando relativamente próximo ao STA que combina Pose e RGB. Esse resultado demonstra que a X3D consegue capturar com fidelidade as diferentes classes de ações, mesmo utilizando apenas informações de pose 2D.

Durante a fase de teste, o tempo de processamento de um único vídeo (*forward pass*) com o STA foi de aproximadamente 338 ms. Embora não se tenha medido ainda o tempo de inferência com o *pipeline* do PoseConv3D, o modelo foi executado de forma eficiente para o conjunto completo de vídeos.

6.2.6 Experimento de agrupamento semântico de classes

O objetivo deste experimento foi avaliar o potencial do modelo para um cenário de monitoramento mais geral de idosos, em que o foco está em categorias amplas de comportamento,

sem preocupação com variações finas de cada ação. Para isso, realizou-se inicialmente um agrupamento semântico das classes baseado na classe base de cada ação, definida como a parte do rótulo anterior ao primeiro ponto. Dessa forma, ações como *Cook.Cleandishes* e *Cook.Cleanup* foram agrupadas na classe *Cook*, enquanto *Drink.Fromcup* e *Drink.Frombottle* passaram a pertencer à classe *Drink*. Após o agrupamento, foram definidas 19 classes-base e o modelo foi treinado novamente com a nova rotulação.

A avaliação do modelo sob essa configuração apresentou uma acurácia geral de 77,7% e uma acurácia média por classe de 67,9% no conjunto de teste. Esses resultados indicam que, embora o modelo tenha desempenho satisfatório no reconhecimento global das ações, algumas classes ainda apresentam maior dificuldade de classificação. A análise da matriz de confusão exibida na Figura 6.15 mostra que classes mais frequentes e visualmente distintas, como *Walk*, apresentam alto número de acertos, enquanto classes menos representadas ou visualmente semelhantes a outras, como *Cutbread* e *Eat*, apresentam maior dispersão fora da diagonal principal. No caso da classe *Eat*, a redução na acurácia em relação às suas ações individuais provavelmente está relacionada à variação de postura durante a execução da ação: enquanto comer à mesa geralmente ocorre sentado, consumir *snacks* pode acontecer em pé, deitado ou em movimento, tornando o reconhecimento mais desafiador.

Embora a fusão das classes *Drink* tenha simplificado a classificação, essa estratégia nem sempre é vantajosa, pois pode agrupar ações que o modelo ainda confunde de formas diferentes, mantendo ambiguidades. Uma alternativa seria agrupar classes com base na similaridade de postura ou nos padrões de confusão observados nas matrizes de confusão, permitindo reduzir de forma mais direcionada a complexidade do problema. Outra abordagem complementar seria treinar modelos especialistas para subconjuntos de classes que apresentam alta similaridade postural ou semântica, de modo a manter a granularidade quando necessário, sem comprometer o desempenho global do sistema.

Ainda assim, o agrupamento semântico das classes cumpriu seu papel de reduzir a complexidade do problema de classificação, permitindo que o modelo aprenda padrões gerais de comportamento dos indivíduos. Mesmo com algumas confusões entre classes semelhantes, os resultados indicam que o modelo é capaz de identificar corretamente a

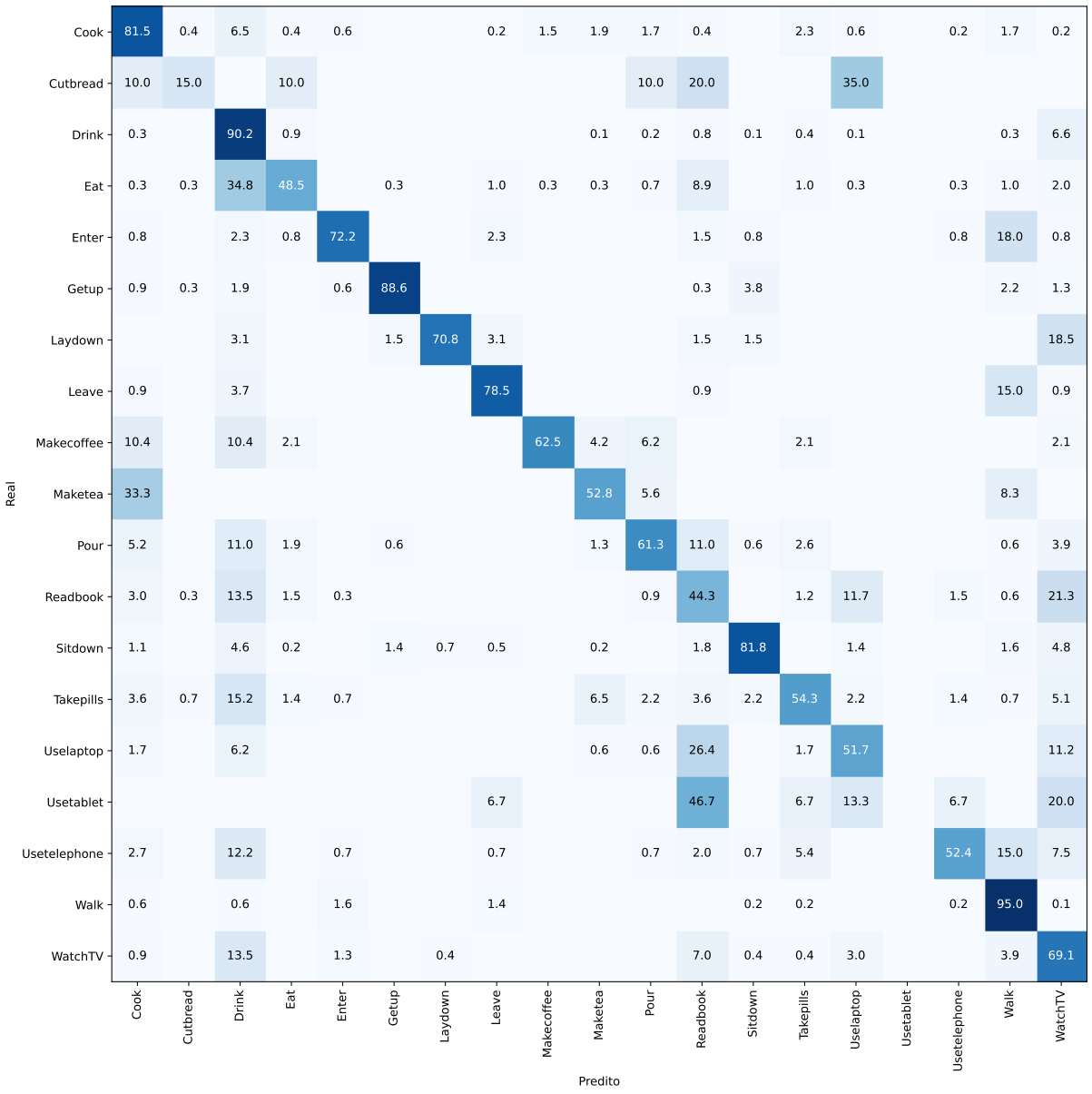


Figura 6.15: Matriz de Confusão do conjunto de teste do TST após agrupamento semântico de classes.

maioria das ações relevantes para o monitoramento geral de idosos, validando a abordagem adotada.

7 Conclusão

Este trabalho investigou a aplicação do modelo PoseConv3D ao conjunto Toyota Smarthome Trimmed (TST) no contexto do reconhecimento de ações humanas voltado ao monitoramento não invasivo de idosos, utilizando exclusivamente informações de vídeo processadas a partir de mapas de calor 2D das articulações. A partir dessa investigação, foi possível demonstrar que o modelo é capaz de monitorar atividades de forma não invasiva por meio de uma rede neural tridimensional baseada em sequências de pose.

Os resultados experimentais indicam que, mesmo em um conjunto desafiador como o TST, caracterizado por elevada variabilidade postural e ações concorrentes, o modelo alcança desempenho consistente, especialmente em ações com variação de postura bem definida, diretamente relevantes para cenários de assistência e prevenção de riscos.

A análise detalhada das matrizes de confusão e das métricas por classe evidenciou que as principais limitações do modelo estão associadas à semelhança semântica e postural entre determinadas ações, em especial aquelas realizadas em posição sentada ou envolvendo interação com objetos semelhantes. Nesse contexto, a aplicação de estratégias de balanceamento e aumento de dados mostrou-se eficaz para mitigar vieses amostrais, reduzir confusões sistemáticas e estabilizar a acurácia média por classe, além de favorecer uma convergência mais precoce do treinamento, mesmo sem impactar significativamente a acurácia global de teste.

O agrupamento semântico das classes permitiu avaliar o modelo sob uma perspectiva mais alinhada ao monitoramento comportamental, demonstrando que a redução da granularidade das ações preserva informações essenciais sobre o estado funcional do indivíduo. Essa abordagem reforça a viabilidade do uso do modelo em aplicações práticas, nas quais a identificação de padrões gerais de comportamento é frequentemente mais relevante do que a distinção entre ações finamente granulares.

Os experimentos também evidenciaram que o pré-treinamento em conjuntos externos, aliado a ajustes criteriosos de hiperparâmetros, é determinante para alcançar um equilíbrio entre capacidade de ajuste e generalização, mesmo em cenários com forte des-

balanceamento de classes.

Comparativamente, a metodologia proposta neste trabalho supera o desempenho de acurácia média por classe reportado no modelo proposto no trabalho original do TST (modelo STA), porém com uma implementação mais simples e de menor custo computacional.

Como perspectivas futuras, planeja-se integrar o modelo a um sistema de monitoramento em tempo real, possibilitando a avaliação do desempenho de inferência e da viabilidade computacional em ambientes domésticos reais. Além disso, a investigação de estratégias avançadas de aumento de dados para classes sub-representadas e a incorporação de informações contextuais da cena, como objetos e *layout* do ambiente, visando a redução de ambiguidades entre ações visualmente semelhantes e ampliação da aplicabilidade do modelo em cenários de assistência à vida diária.

Bibliografia

ABDELGAWAD, A.; YELAMARTHI, K.; KHATTAB, A. Iot-based health monitoring system for active and assisted living. In: *Smart Objects and Technologies for Social Good: Second International Conference, GOODTECHS 2016*. [S.l.: s.n.], 2017. p. 11 – 20.

AGGARWAL, J. K.; RYOO, M. S. Human activity analysis: A review. In: *ACM Computing Surveys (CSUR)*. [S.l.: s.n.], 2011. (3, v. 43).

ALEMáN, J. J.; SáNCHEZ-PI N.AND MARTI, L.; MOLINA, J. M.; GARCIA, A. C. B. A data fusion model for ambient assisted living. In: *Highlights of Practical Applications of Scalable Multi-Agent Systems. The PAAMS Collection: International Workshops of PAAMS 2016*. Servilla, Spain: [s.n.], 2016. p. 301 – 312.

BEAUCHEMIN, S. S.; BARRON, J. L. The computation of optical flow. In: *ACM Computing Surveys*. [S.l.: s.n.], 1995. v. 27.

BOBICK, A. F.; DAVIS, J. W. The recognition of human movement using temporal templates. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence*. [S.l.: s.n.], 2001. (3, v. 23).

BRASIL. *Estratégia Brasil Amigo da Pessoa Idosa: Documento Técnico*. 2018. Brasília, DF: Ministério da Cidadania.

BRASIL. *Boletim temático da biblioteca do Ministério da Saúde: Saúde do Idoso*. 2022. Brasília: Ministério da Saúde.

BRASIL. *Respeito a todas as fases da vida: Junho Violeta*. 2024. ⟨https://www.gov.br/mdh/⟩.

BUZZELLI, M.; ALBé, A.; CIOCCA, G. A vision-based system for monitoring elderly people at home. In: *Applied Sciences*. [S.l.: s.n.], 2020. (1, v. 10).

CICIRELLI, G.; MARANI, R.; PETITTI, A.; MILELLA, A.; D'ORAZIO, T. Ambient assisted living: A review of technologies, methodologies and future perspectives for healthy aging of population. In: *Sensors*. [S.l.: s.n.], 2021. (10, v. 21).

DAI, R.; DAS, S.; SHARMA, S.; MINCIULLO, L.; GARATTONI, L.; BREMOND, F.; FRANCESCA, G. Toyota smarthome untrimmed: Real-world untrimmed videos for activity detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, IEEE, v. 45, n. 2, p. 2533–2550, 2022.

DALAL, N.; TRIGGS, B. Human detection using oriented histograms of flow and appearance. In: *Proc. European Conference on Computer Vision (ECCV)*. [S.l.: s.n.], 2005. v. 1, p. 886 – 893.

DAS, S.; DAI, R.; KOPERSKI, M.; MINCIULLO, L.; GARATTONI, L.; BREMOND, F.; FRANCESCA, G. Toyota smarthome: Real-world activities of daily living. In: *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*. [S.l.: s.n.], 2019. p. 833–842.

DENG, J.; DONG, W.; SOCHER, R.; LI, L.; LI, K.; FEI-FEI, L. Imagenet: A large-scale hierarchical image database. In: *IEEE conference on computer vision and pattern recognition*. [S.l.: s.n.], 2009.

DUAN, H.; ZHAO, Y.; CHEN, K.; LIN, D.; DAI, B. Revisiting skeleton-based action recognition. In: *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. [S.l.: s.n.], 2022. p. 2959–2968.

FEICHTENHOFER, C. X3d: Expanding architectures for efficient video recognition. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. [S.l.: s.n.], 2020. p. 203–213.

FISCHLER, M. A.; ELSCHLAGER, R. A. The representation and matching of pictorial structures. *IEEE Transactions on computers*, IEEE, v. 100, n. 1, p. 67–92, 1973.

GAIKWAD; SUDHIR; BHATLAWANDE, S.; SHILASKAR, S.; SOLANKE, A. A computer vision-approach for activity recognition and residential monitoring of elderly people. In: *Medicine in Novel Technology and Devices*. [S.l.: s.n.], 2023.

GIRSHICK, R. Fast r-cnn. In: *Proceedings of the IEEE international conference on computer vision*. [S.l.: s.n.], 2015. p. 1440–1448.

GONZALEZ, R. C.; WOODS, R. E. Digital image processing 4th edition. In: _____. [S.l.]: Pearson, 2018. ISBN 978-1-292-22304-9.

GOODFELLOW, I.; BENGIO Y.AND COURVILLE, A. Deep learning. In: _____. [S.l.]: The MIT Press, 2016. ISBN 9780262035613.

GUO, P.; MIAO, Z.; SHEN, Y.; CHENG, H. Real time human action recognition in a long video sequence. In: *Proceedings of the 2010 7th IEEE International Conference on Advanced Video and Signal Based Surveillance*. Boston, MA, USA: [s.n.], 2010. p. 248 — 255.

HAYAT, A.; MORGADO-DIAS, F.; BHUYAN, B. P.; TOMAR, R. Human activity recognition for elderly people using machine and deep learning approaches. *Information*, MDPI, v. 13, n. 6, p. 275, 2022.

HUSSAIN; A.; WENBI, R.; SILVA A.L.AND NADHER, M. D.; MUDHISH, M. Health and emergency-care platform for the elderly and disabled people in the smart city. In: *Journal of Systems and Software*. [S.l.: s.n.], 2015. v. 100, p. 253 –263.

IBGE. *Censo Demográfico 2022: Panorama*. 2022. ⟨https://censo2022.ibge.gov.br/panorama/⟩.

KONG, Y.; FU, Y. Human action recognition and prediction: A survey. In: *International Journal of Computer Vision*. [S.l.: s.n.], 2022. (5, v. 130).

KOPPULA, H. S.; GUPTA, R.; SAXENA, A. *Northwestern-UCLA Multiview activity 3D Dataset*. 2024. Disponível em: ⟨https://service.tib.eu/ldmservice/dataset/northwestern-ucla-multiview-activity-3d-dataset⟩.

KRIZHEVSKY, A.; SUTSKEVER, I.; HINTON, G. E. Imagenet classification with deep convo- lutional neural networks. In: *Proc. Advances in Neural Information Processing Systems (NIPS), pages*. [S.l.: s.n.], 2012. p. 1097 – 1105.

LIN, T.-Y.; MAIRE, M.; BELONGIE, S.; HAYS, J.; PERONA, P.; RAMANAN, D.; DOLLÁR, P.; ZITNICK, C. L. Microsoft coco: Common objects in context. In: SPRINGER. *European conference on computer vision*. [S.l.], 2014. p. 740–755.

LIN, T.-Y.; MAIRE, M.; BELONGIE, S.; HAYS, J.; PERONA, P.; RAMANAN, D.; DOLLÁR, P.; ZITNICK, C. L. Microsoft coco: Common objects in context. In: SPRINGER. *European conference on computer vision*. [S.l.], 2014. p. 740–755.

NúñEZ-MARCOS, A.; AZKUNE, G.; ARGANDA-CARRERAS, I. Vision-based fall detection with convolutional neural networks. In: *Wireless communications and mobile computing*. [S.l.: s.n.], 2022.

ONU. *World Population Prospects 2024: Summary of Results*. 2024. UN DESA/-POP/2024/TR/NO. 9. Disponível em: ⟨https://population.un.org/wpp/⟩.

OUDAH, M.; AL-NAJI, A.; CHAHL, J. Computer vision for elderly care based on deep learning cnn and svm. In: *OP Conference Series: Materials Science and Engineering*. [S.l.: s.n.], 2020. (1, v. 1105).

PHUNG, V. H.; RHEE, E. J. A high-accuracy model average ensemble of convolutional neural networks for classification of cloud image patches on small datasets. *Applied Sciences*, MDPI, v. 9, n. 21, p. 4500, 2019.

PLANINC, R.; KAMPEL, M. *Fall Database*. 2012. ACCV Workshop on Color Depth Fusion in Computer Vision. 72 video sequences: 40 falls + 32 ADLs; doi:10.5281/zenodo.3886586.

RAHMANI, H.; MAHMOOD, A.; DU, H.; MIAN, A. *UWA 3D Multiview Activity II Dataset*. 2013. 30 activities, 10 subjects, 4 views with Kinect.

RISPA. *Demografia e Saúde: Contribuição para Análise de Situação e Tendências*. 2009. Brasília: Organização Pan-Americana da Saúde. Série G. Estatística e Informação em Saúde.

ROSENBLATT, F. The perceptron: A probabilistic model for information storage and organization in the brain. In: *Psychological Review*. [S.l.: s.n.], 1958. (6, v. 65), p. 386 — 408.

SHAHROUDY, A.; LIU, J.; NG, T.-T.; WANG, G. Ntu rgb+d: A large scale dataset for 3d human activity analysis. In: *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*. [S.l.: s.n.], 2016. p. 1010–1019.

SHUCHANG, Z. A survey on human action recognition. *arXiv preprint arXiv:2301.06082*, 2022.

SZELISKI, R. Computer vision: Algorithms and applications 2nd edition. In: _____. [S.l.]: Springer, 2022.

TOMPSON, J.; JAIN, A.; LECUN, Y.; BREGLER, C. Joint training of a convolutional network and a graphical model for human pose estimation. *Advances in neural information processing systems*, v. 27, 2014.

TOSHEV, A.; SZEGEDY, C. Deeppose: Human pose estimation via deep neural networks. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. [S.l.: s.n.], 2014. p. 1653–1660.

TURAGA, P.; CHELLAPPA, R.; SUBRAHMANIAN, V.; UDREA, O. Machine recognition of human activities: A survey. In: *IEEE Transactions on Circuits and Systems for Video technology*. [S.l.: s.n.], 2008.

WANG, H.; KLASER, A.; SCHMID, C.; LIU, C.-L. Action recognition by dense trajectories. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence*. [S.l.: s.n.], 2011. p. 3169 – 3176.

WANG, J.; LIU, Z.; WU, Y.; YUAN, J. *MSR DailyActivity3D Dataset*. 2012. Captured via Kinect; 16 activities, 10 subjects. Data channel: RGB, depth, skeleton.

WANG, J.; SUN, K.; CHENG, T.; JIANG, B.; DENG, C.; ZHAO, Y.; LIU, D.; MU, Y.; TAN, M.; WANG, X. et al. Deep high-resolution representation learning for visual recognition. *IEEE transactions on pattern analysis and machine intelligence*, IEEE, v. 43, n. 10, p. 3349–3364, 2020.

WANG, L.; HU, W.; TAM, T. Recent developments in human motion analysis. pattern recognition. In: *Pattern Recognition*. [S.l.: s.n.], 2003. (3, v. 36).

WEINLAND, D.; RONFARD, R.; BOYER, E. Free viewpoint action recognition using motion history volumes. *Computer Vision and Image Understanding*, v. 104, n. 2–3, p. 249–257, 2006.

ZHAI, M.; HUANG, Y.; ZHOU, S.; JIN, Y.; FENG, J.; PEI, C.; WEN, L.; WEN'S, L. Effects of age-related changes in trunk and lower limb range of motion on gait. *BMC musculoskeletal disorders*, Springer, v. 24, n. 1, p. 234, 2023.

ZIN, T. T.; 1, Y. H.; AKAGI, Y.; TAMURA, H.; KONDO, K.; ARAKI, S.; CHOSA, E. Real-time action recognition system for elderly people using stereo depth camera. In: *Sensors*. [S.l.: s.n.], 2021. (17, v. 21).