

Sistema de Detecção de Quedas em Idosos Usando YOLOv8-Pose e Machine Learning

Cleide Lustosa de Oliveira da Silva¹, Erika Borges Piaui²

¹Faculda de Computação e Informática (FCI)

Universidade Presbiteriana Mackenzie, São Paulo – SP - Brasil

10409459@mackenzista.com.br, 10403716@mackenzista.com.br

Resumo. Este projeto apresenta um sistema automático de detecção de quedas com vídeos utilizando a arquitetura YOLOv8-pose para extração de keypoints (pontos-chave) do corpo humano combinada com algoritmos de machine Lane para a classificação. O sistema foi treinado com 89 vídeos (50 quedas e 39 atividades normais) e alcançou uma acurácia de 93,2% com o modelo Random Forest. O sistema foi disponibilizado através de uma interface web usando Streamlit permitindo a análise dos vídeos. Esse trabalho aborda a importância de sistemas que sejam produzidos para assistência e monitoramento de quedas em idosos e pessoas com mobilidade reduzida.

1. Introdução

As quedas representam um dos principais problemas de saúde pública, especialmente em populações idosas segundo a organização mundial de saúde (OMS), as quedas são a segunda causa da morte não intencional em todo o mundo sendo particularmente preocupantes indivíduos com mais de 65 anos (WHO,2021). O desenvolvimento de sistemas automáticos para detecção de queda pode reduzir significativamente o tempo de resposta em ambientes hospitalares, asilos e residências, potencialmente salvando vidas.

2. Objetivo

2.1 Objetivo Geral

Desenvolver um sistema de detecção de quedas utilizando técnicas para validar e treinar modelos de Machine Learning para detectar quedas em vídeos usando features com YOLOv8-Pose.

2.2 Objetivo Específico

- Extrair keypoints corporais de vídeos por meio do YOLOv8-Pose.
- Desenvolver e comparar algoritmos de Machine Learning aplicados à classificação.

- Validar os modelos utilizando validação cruzada estratificada.
- Disponibilizar uma interface web interativa para utilização do sistema.

3. Metodologia

A classificação de eventos de queda baseia-se na cinemática e geométrica fornecida pela estimativa de pose do YOLOv8. O algoritmo mapeia 17 pontos-chave (keypoints) da estrutura corporal humana (como o ombro, quadril, joelhos e tornozelos), nesse projeto usaremos os pontos da variação do ombro e do quadril. A distinção entre queda e atividades cotidiana (ADLs – *Activities of Daily Living*) é realizada através de 3 vetores principais:

- **Orientação do eixo corporal:** calcula-se o ângulo formado pela linha que conecta o ponto médio dos ombros ao ponto médio dos quadris em relação ao plano vertical.
- **Razão de Aspecto da Bounding Box:** Analisa se a proporção entre a largura e altura da caixa delimitadora do indivíduo. Durante uma queda, a largura da caixa tende a superar a altura, invertendo a proporção observada em uma postura ereta.
- **Velocidade de Deslocamento:** A derivada temporal da posição dos centroides (velocidade) é utilizada para diferenciar uma queda (movimento não intencional e acelerado) de um agachamento ou deitar-se controlado (movimentos lentos e intencionais).

Após a extração das características biomecânicas por meio do modelo YOLOv8, os dados foram submetidos a uma etapa de classificação supervisionada (Pedregosa *et al.*, 2011). Para determinar a abordagem mais eficiente da distinção entre queda e atividades cotidiana foram treinados 11 modelos distintos, variando os algoritmos e os hiperparâmetros. O conjunto de treinamento inclui o algoritmo Random Forest (Breman, 2021), testado em 3 profundidades de árvore diferentes, Arvore de Decisão (*Decision Trees*) em duas configurações distintas, o método *K-Nearest Neighbors* (KNN), avaliado com K igual 3, 5 e 7, redes neurais MLP (*Multi-layer Perceptron*) com duas arquiteturas de camadas ocultas e uma *Support Vector Machine* (SVM) utilizando o kernel de Base Radial, (KWOLEK; KEPSKI, 2014).

Arquitetura escolhida para a tarefa de estimativa de pose foi o YOLOv8-Pose, desenvolvida pela Ultralytics. Trata-se de um modelo que oferece desempenho superior em métricas de precisão e latência, o modelo é versátil e super suporta tarefas de detecção de objetos, segmentação de instância e é significativamente relevante para a estimativa de poses. A capacidade do modelo torna uma escolha robusta para aplicações que exigem processamento em tempo real e alta confiabilidade na identificação de pontos chaves do corpo humano (ULTRALYTICS, 2024).

3.1 Dataset

Utilizou-se um dataset com 89 vídeos, disponibilizados por Ekram Alam (2024). Sendo dividido em:

- 50 vídeos de quedas, controlado em ambientes

- 39 vídeos de comportamento normal do cotidiano como sentar, caminhar e levantar.

3.2 Pipeline de Processamento

O pipeline completo do desenvolvimento do projeto inclui as seguintes etapas:

- Processamento de vídeos com YOLOv8-Pose
- Extração de keypoints corporais
- Cálculo de 8 features relacionadas ao movimento vertical e à postura final
- Normalização dos Dados (StandardScaler)
- Treinamento e comparação entre diferentes modelos de Machine Learning

3.3 Features Desenvolvidas

As features incluem medidas de como velocidade de deslocamento de quadril, variação de altura, distância entre o ombro e o quadril e um indicador de postura final (em pé/deitado) entre outros atributos essenciais para a detecção de queda.

3.4 Modelos Avaliados

Foram treinados 11 modelos com diferentes parâmetros:

- Random Forest (3 profundidades)
- Decision Tree (2 configurações)
- KNN ($k=3, 5, 7$)
- MLP (duas arquiteturas)
- SVM (kernel RBF)

3.5 Critérios de Validação

A validação seguiu o método de validação cruzada estratificada com 5 folds, medindo acurácia, precisão, recall e F1-score.

A validação seguiu o método de avaliação cruzada estratificada com 5 folds, medindo acurácia, precisão, recall e F1-score.

4. RESULTADOS

4.1 Desempenho Geral

O modelo com melhor desempenho foi o Random Forest com profundidade 10. Dentre os 11 modelos avaliados, o Random Forest apresentou o melhor desempenho geral, atingindo 93,2% de acurácia. Esse modelo é um classificador que opera combinando múltiplas árvores de decisão e sua vantagem aponta para um classificador mais robusto

em relação ao ruído (Breman, 2021). O que torna ideal para o problema de detecção de queda quando extraídas as features da pose usando a velocidade do quadril ou a posição do ombro.

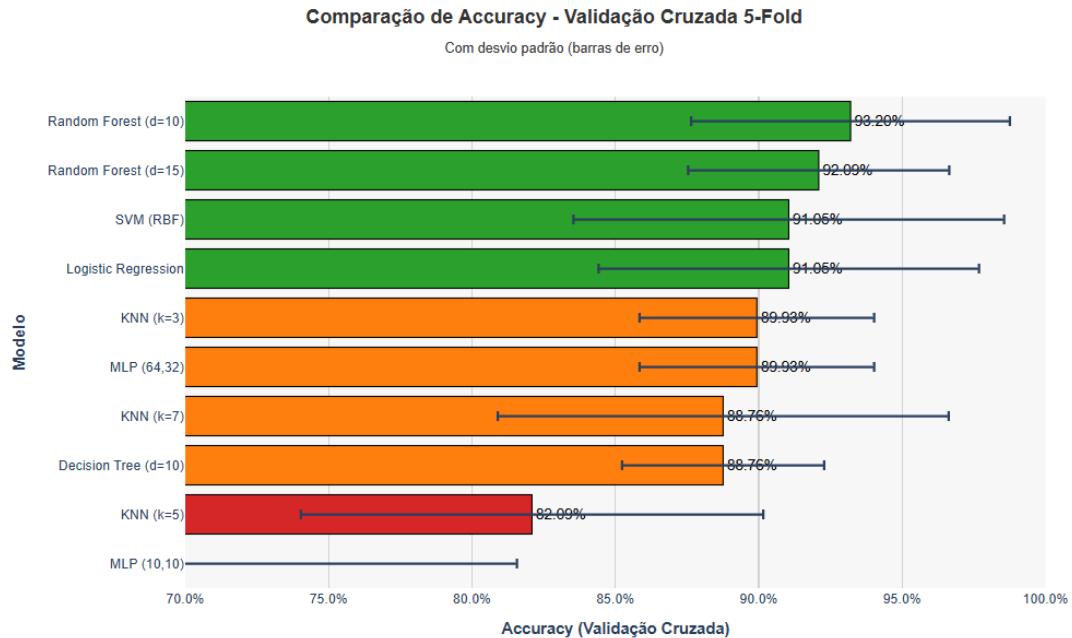


Figura 1: Gráfico de comparação de modelos

Fonte: Fonte: Colcab Autoras: detector_quedas.ipynb

4.2 Métricas de Avaliação

O desempenho do modelo foi avaliado por meio de 5 considerações complementares:

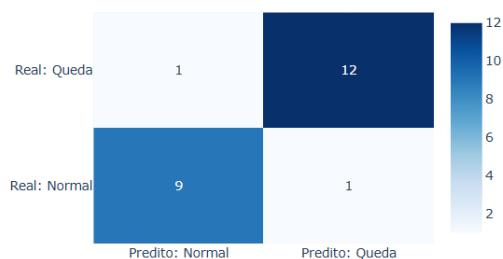
- Precisão de 93,2% - Taxa geral de previsões corretas, relatada que o modelo acerta.
- Precisão de 94% - Das vezes que o modelo previu uma queda, acertou em 94% dos casos. Isto implica em apenas que 6% de alarmes falsos.
- Recall de 94% - Das quedas que realmente ocorreram, o modelo detectou 94% deixando apenas 6% não bloqueados.
- F1-Score de 94% - Métricas harmônicas entre Precisão e Recall, representam o melhor balanço entre falsos alarmes de quedas sem perder as quedas reais.
- ROC-AUC de 95% - Valor próximo a 1.0 indica excelente capacidade de discriminação entre quedas e atividades normais.

4.3 Análise de Erros

Os principais erros identificados incluem:

- Falso positivos ocasionados provavelmente por motivos bruscos
- Falsos negativos em quedas lentas com pouca visibilidade

Matriz de Confusão - Random Forest (d=10) (Hold-Out)



Relatório de Classificação:

	precision	recall	f1-score	support
Normal	0.90	0.90	0.90	10
Queda	0.92	0.92	0.92	13
accuracy			0.91	23
macro avg	0.91	0.91	0.91	23
weighted avg	0.91	0.91	0.91	23

Figura 2: Matriz de Confusão

Fonte: Colcab Autoras: detector_quedas.ipynb

5. Implementação Deploy

5.1 Arquitetura Geral

A arquitetura do sistema integra:

- YOLOv8-Pose
- Módulo de pré-processamento
- Classificador Random Forest
- Interface web em Streamlit

5.2 Interface Streamlit

A interface permite que:

- Upload de vídeos
- Execução do detector
- Visualização da featuars
- Relatórios de desempenho dos modelos

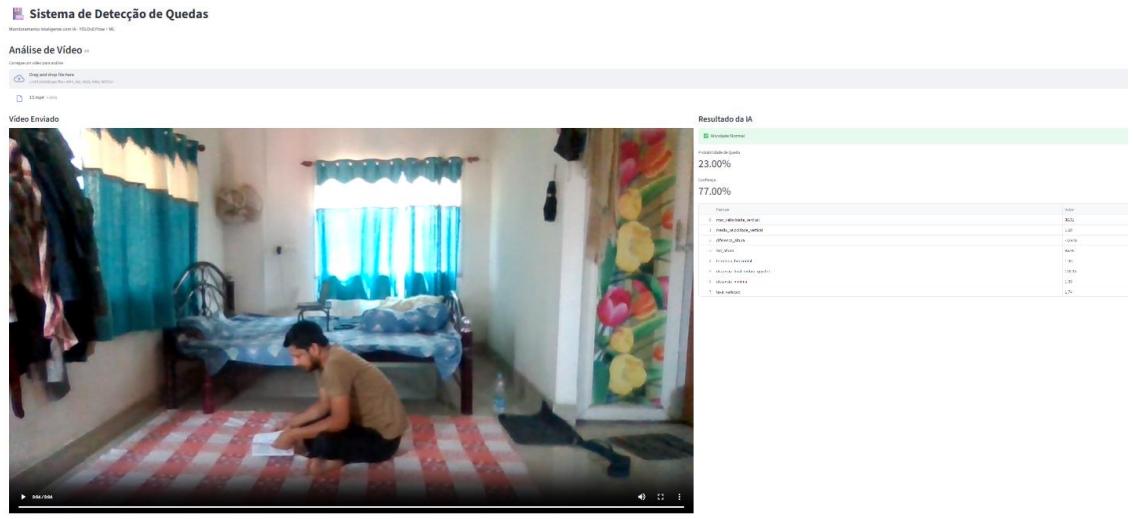


Figura 3: Sistema com vídeo de não queda

Fonte: Sitema de detecção de queda local: [Detecção de Quedas](#)

Carregue um vídeo para análise

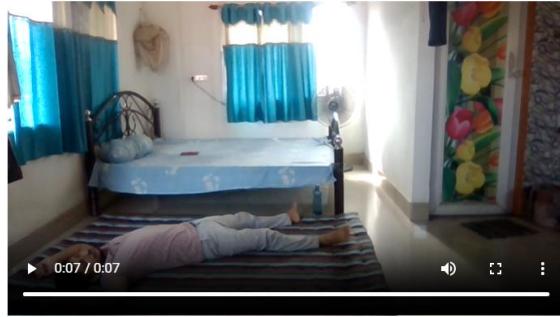
Cloud

Drag and drop file here
Browse files

Limit 200MB per file • MP4, AVI, MOV, MKV, MPEG4

25.mp4 7.2MB X

Vídeo Enviado



Resultado da IA

⚠️ QUE DA DETECTADA!

Probabilidade de Queda

100.00%

Confiança

100.00%

Feature	Valor
0 max_velocidade_vertical	40.34

Figura 4: Sistema com vídeo de queda

Fonte: Sitema de detecção de queda local: [Detecção de Quedas](#)

5.3 Disponibilização

O sistema foi implantado localmente e permite o acesso ao código-fonte, modelos treinados, script de pré-processamento e documentação de uso que estarão acessíveis por meio de um repositório no GitHub.

O repositório inclui instruções completas para a instalação dos pacotes necessários configuração do ambiente, preparação do dataset e execução do sistema de detecção de quedas, permitindo que qualquer usuário recria o ambiente localmente.

Os passos incluem:

- Instalação de dependências (Python, bibliotecas e ambientes virtuais);
- Estruturação das pastas de vídeos;
- Execução dos notebooks de treinos e testes.

O link para o repositório GitHub será disponibilizado juntamente ao relatório final, garantindo acesso aberto e replicável ao projeto.

6. CONCLUSÃO

Esse projeto demonstrou a viabilidade de sistemas híbridos para detecção de queda usando YOLOv8-Pose combinado com algoritmos convencionais de Machine Learning. O modelo Random Forest apresentou desempenho superior, atingindo 93,2% de acurácia. A solução desenvolvida pode servir como base para aplicações com assistência de prevenção de quedas em ambientes residenciais, hospitalares e instituições de cuidado para pessoas com mobilidade reduzida e idosos.

Referências

- Breiman, L. Random Forests. Machine Learning 45, 5–32 (2001). doi.org/10.1023/A:1010933404324.]
- Ekram, Alam. (2024). ekramalam/GMDCSA24-A-Dataset-for-Human-Fall-Detection-in-Videos: 2.0 (v2.0). Zenodo. <https://doi.org/10.5281/zenodo.12921216>.
- KWOLEK, Bogdan; KEPSKI, Michal. Human fall detection using depth maps and accelerometers. IEEE Transactions on Consumer Electronics, v. 60, n. 3, p. 489-496, ago. 2014. doi.org/10.1109/j.cmpb.2014.09.005.
- PEDREGOSA, et al., 2011. Scikit-learn: Machine Learning in Python. Journal of Machine Learning Research, 12, 2825-2830. Disponível em: <https://dl.acm.org/doi/10.5555/1953048.2078195>.
- STREAMLIT, 2024. Streamlit: The fastest way to build and share data apps. Disponível em: <https://streamlit.io>.
- ULTRALYTICS. YOLOv8 Docs., 2024. Disponível em: <https://docs.ultralytics.com/pt/models/yolov8/>. Acesso em: 17 nov. 2025.
- World Health Organization (WHO, 2021). Falls: Key facts. Disponível em: <https://www.who.int/news-room/fact-sheets/detail/falls>. Acesso em 16 de nov. 2025.