

UNIVERSIDADE DO OESTE DE SANTA CATARINA

ANA PAULA RAMPANELLI

ANÁLISE BIOMECÂNICA DE ARREMESSOS NO BASQUETE COM
VISÃO COMPUTACIONAL

CHAPECÓ, SC

2025

ANA PAULA RAMPANELLI

ANÁLISE BIOMECÂNICA DE ARREMESSOS NO BASQUETE COM VISÃO
COMPUTACIONAL

Projeto de Trabalho de Conclusão de Curso apresentado ao Curso de Engenharia de Computação, Área das Ciências Exatas e da Terra, da Universidade do Oeste de Santa Catarina – Unoesc Campus de Chapecó, como requisito parcial à obtenção do grau de bacharel em Engenharia de Computação.

Chapecó, SC

2025

SUMÁRIO

1 INTRODUÇÃO

A utilização da inteligência artificial IA vem revolucionado de maneira significativa em diferentes setores do mercado. Desde novembro de 2022, com o lançamento do ChatGPT a população teve acesso ao que era exclusivo somente as pessoas que trabalhavam na área, depois desse episódio, a maneira de pesquisar/buscar mudou completamente de patamar, hoje estudos mostram que o novo formato de interagir com os dados está cada vez mais utilizados e aceito pela gerações.

As organizações vêm incorporando essas ferramentas tanto em processos internos, visando maior eficiência operacional, quanto em soluções voltadas ao usuário final, proporcionando novas experiências de interação. No contexto de e-commerce, por exemplo, os mecanismos de busca tradicionais, baseados em filtros rígidos, podem se mostrar limitados e pouco intuitivos.

Nesse cenário, modelos de linguagem de grande escala LLM têm se destacado ao possibilitar a conversão de linguagem natural em consultas estruturadas (text-to-SQL), o que favorece buscas mais personalizadas e acessíveis. Essa evolução contribui para a simplificação da experiência do usuário, tornando o processo de encontrar produtos e serviços mais eficiente.

A empresa Amo Delivery, fundada em 2017 e sediada em Chapecó, consolidou-se como uma plataforma dedicada a melhorar a experiência dos usuários no setor alimentício. Atualmente presente em mais de 300 cidades do Brasil, sua solução de software, denominada “Amo Ofertas”, possibilita ao usuário acessar, de maneira prática e centralizada, ofertas de diferentes restaurantes, trazendo conveniência e agilidade ao processo de escolha.

Diante desse contexto, o presente trabalho propõe o estudo e na implementação de um sistema de busca inteligente no aplicativo Amo Ofertas, com base em modelos de linguagem natural. O sistema permitirá que os usuários realizem pesquisas em linguagem natural, eliminando a necessidade de filtros rígidos e comandos específicos, ao converter as consultas em linguagem SQL estruturada (text-to-SQL). Dessa forma, o usuário poderá buscar por ofertas de forma intuitiva, utilizando expressões semelhantes à conversação cotidiana.

A implementação dessas funcionalidades no contexto de busca e consulta de ofertas dentro da plataforma Amo Ofertas, não irão contemplar modificações nos processos de

pagamento, logística de entrega ou outros módulos já existentes. O foco deste trabalho permanecerá no aprimoramento da experiência de busca do usuário, por meio da aplicação de técnicas de inteligência artificial voltadas à interação natural e personalizada.

1.1 PROBLEMA E JUSTIFICATIVA

A experiência do usuário em plataformas de e-commerce está diretamente relacionada à sua capacidade de encontrar, de forma rápida e intuitiva, os produtos que deseja. Com o avanço da tecnologia, o uso de modelos de linguagem natural LLMS tornou-se uma tendência crescente, oferecendo interações mais inteligentes e próximas da linguagem cotidiana dos consumidores. Estudos indicam que a eficácia dos mecanismos de busca em plataformas digitais impacta de maneira significativa tanto a satisfação do usuário quanto a concretização das vendas (JANSEN; MOLINA, 2006).

Empresas consolidadas, como Google, Nubank e C6 Bank, já incorporam tecnologias de processamento de linguagem natural PLN em seus serviços, elevando o nível da experiência do usuário. No setor de e-commerce, organizações como Amazon, Shopify e iFood vêm explorando recursos de aprendizado de máquina (do inglês ,Machine Learning ML) para tornar a busca mais personalizada e acessível, permitindo que os usuários encontrem produtos de forma simples e eficiente.

Apesar do crescimento do comércio eletrônico e da evolução dos sistemas de busca baseados em IA, muitos aplicativos ainda oferecem experiências limitadas ao usuário, especialmente em relação à forma de localizar produtos. Os filtros de pesquisa são essenciais para permitir que o usuário localize o que deseja com base em critérios específicos para sua solicitação como, o preço, ingredientes, distância, entre outros atributos.

O aplicativo AMO Ofertas é utiliza filtros rígidos e pouco flexíveis, o que compromete a experiência do usuário e pode limitar sua competitividade no mercado. Diante disso, este trabalho justifica-se pela necessidade de implementar um modelo de linguagem natural que permita buscas mais inteligentes e personalizadas, ampliando a satisfação do consumidor, fortalecendo a posição da empresa no setor e possibilitando uma concorrência mais equilibrada com players de maior porte.

1.2 OBJETIVOS

Nesta seção serão abordados os objetivos gerais e específicos a serem buscados no decorrer da execução do trabalho proposto.

1.2.1 Objetivo geral

Implementar e avaliar um modelo de linguagem natural /lmmLLM no aplicativo AMO Ofertas, com o propósito de aprimorar o mecanismo de busca, tornando-o mais intuitivo e eficiente, de modo a melhorar a experiência do usuário e aumentar a competitividade da plataforma no mercado de e-commerce.

1.2.2 Objetivos específicos

Os objetivos específicos do projeto seguem a ordem cronológica de execução das atividades, conforme descrito a seguir:

- Estudar a estrutura do sistema de busca existente e documentar as limitações dos filtros rígidos atualmente oferecidos no sistema Amo Ofertas.
- Coletar logs/chaves de pesquisas realizadas no aplicativo ou aplicar questionários/testes com usuários para identificar padrões de busca, como termos mais usados, erros frequentes afim de analisar dados de busca dos usuários.
- Realizar um estudo comparativo de desempenho dos modelos de linguagem natural, requisitos computacionais e aderência ao português, justificando a escolha do modelo mais adequado para o contexto da Amo Ofertas.
- Implementar uma versão experimental em ambiente controlado (Sandbox) para realizar testes isolada para treinar e validar o modelo escolhido, simulando consultas reais de usuários.
- Treinar e ajustar o modelo para o domínio do e-commerce da Amo Ofertas afim de melhorar a relevância e precisão das respostas de busca.
- Implementar o protótipo dentro do aplicativo, disponibilizar a busca inteligente para um grupo de usuários-teste e coletar feedback qualitativo e quantitativo.
- Medir métricas como precisão das buscas, tempo médio para encontrar produtos, satisfação do usuário e comparar com o sistema de busca anterior.

2 REVISÃO DA LITERATURA

Este capítulo apresenta a revisão da literatura sobre os principais conceitos que fundamentam este trabalho, abordando as tecnologias e os estudos relevantes para a análise de movimento no esporte.

2.1 BASQUETEBOL

O basquetebol foi criado em 1891 por James Naismith, professor de Educação Física na Associação Cristã de Moços (ACM) de Springfield, Massachusetts, com o objetivo de desenvolver um esporte que pudesse ser praticado em ambientes fechados durante o inverno rigoroso. Segundo o Comitê Olímpico do Brasil, "o basquete chegou ao Brasil poucos anos depois através do americano Augusto Shaw, que em 1892 graduou-se como bacharel em Artes pela Universidade de Yale e tomou contato pela primeira vez com o basquete. Dois anos depois, desembarcou no Brasil para dar aulas no Colégio Mackenzie, em São Paulo. Além dos livros, trouxe também uma bola de basquete na bagagem"(COB, 2024).

As regras do basquetebol são regulamentadas pela Federação Internacional de Basquetebol (FIBA). A quadra oficial mede 28 metros de comprimento por 15 metros de largura, e o aro está fixado a 3,05 metros do solo. Cada equipe é composta por cinco jogadores em quadra, e o jogo é dividido em quatro períodos de 10 minutos. Entre as principais normas, destaca-se o tempo de posse de bola de 24 segundos e o reinício de tempo em situações específicas. Segundo as Regras Oficiais da FIBA 2020, “após a bola tocar o aro da cesta dos oponentes, o relógio de 24 segundos deverá ser reprogramado para: 24 segundos, se a equipe adversária obtiver o controle da bola; 14 segundos, se a equipe que recupera o controle da bola for a mesma equipe que tinha o controle da bola antes dela tocar o aro” (CBB, 2020, p. 30).

Os fundamentos técnicos do basquetebol incluem o drible, o passe, o arremesso, o rebote e a marcação. O arremesso é considerado o fundamento mais importante, pois define a pontuação final do jogo. De acordo com o portal Impulsiona, “ao conhecer o basquete, é importante ter em mente que apenas 3 movimentos são responsáveis por, talvez, mais de 80% do tempo de jogo, criando toda a dinâmica da partida. Podendo ser realizados de várias formas, em separado, combinados entre si ou com outros movimentos, são eles: drible, passe e arremesso” (IMPULSIONA, 2023).

2.2 ANÁLISE DE DESEMPENHO ESPORTIVO

A análise de desempenho esportivo é uma prática fundamental que visa compreender e otimizar o rendimento de atletas e equipes por meio da coleta, interpretação e aplicação de dados objetivos. Essa abordagem multidisciplinar envolve a avaliação de aspectos físicos, técnicos, táticos e psicológicos, permitindo uma compreensão holística do desempenho esportivo.

Segundo De Rose Júnior, Gaspar e Assumpção (2005), “a avaliação do desempenho esportivo, por meio de diferentes indicadores (físico, técnico, tático, psicológico) de jogo, constitui-se em um método válido, objetivo e fidedigno”.

Além disso, a tecnologia tem desempenhado um papel crucial na evolução da análise de desempenho. Conforme destacado por Lamas e Morales (2022), “os registros de análises do desempenho nos esportes coletivos indicam extensa tradição de desenvolvimento metodológico, favorecida pela crescente disponibilidade de dados e técnicas computacionais”.

2.2.1 Análise de Desempenho no Basquetebol

No contexto do basquetebol, a análise de desempenho tem se mostrado essencial para o desenvolvimento de estratégias e melhoria do rendimento individual e coletivo. A utilização de métricas específicas, como eficiência nos arremessos, número de assistências, rebotes e turnovers, proporciona uma visão detalhada das ações em quadra.

De acordo com Canan, Mendes e Silva (2015), “arremessos de dois pontos tentados, arremessos de dois pontos convertidos, porcentagem de acerto de dois pontos, total de pontos feitos, porcentagem de acerto total, rebotes defensivos, rebotes totais e assistências foram considerados significativos para obtenção da vitória no jogo”.

Ferramentas como o *Basketball Stats Assistant* permitem que treinadores e analistas acompanhem estatísticas em tempo real, facilitando a tomada de decisões durante as partidas. O aplicativo se destaca pela interface intuitiva e pelo detalhamento das estatísticas registradas durante os jogos, fornecendo uma gama ampla de funcionalidades que apoiam diretamente a análise de desempenho (Basketballstats, 2024).

Entre os principais recursos oferecidos pela ferramenta, destacam-se:

- **Registro intuitivo de ações em quadra:** permite registrar jogadas com o gesto de arrastar e soltar sobre os jogadores no campo;
- **Sequência de ações:** suporte a jogadas subsequentes, como rebotes ou faltas, imediatamente após a jogada principal;
- **Controle do tempo de jogo:** exibição do período e tempo restante, com possibilidade de edição direta no placar;
- **Estatísticas em tempo real:** atualizações dinâmicas sobre ações individuais e coletivas;
- **Correção de dados:** permite editar ou excluir eventos incorretos mesmo após a partida;
- **Mapas de arremesso (*shot charts*):** visualização dos pontos de acerto e erro em cada área da quadra;
- **Estatísticas detalhadas:** coleta de dados como assistências, rebotes, roubos, faltas, tocos, turnovers, minutos jogados, entre outros;
- **Relatórios pós-jogo:** inclui *box score*(placar), gráficos comparativos entre os jogadores ou equipes, *play-by-play*, eficiência e análise de *plus/minus*(Estatística usada para medir o impacto de um jogador num jogo).

Figura 1 – Interface do aplicativo Basketball Stats Assistant, utilizado na análise de desempenho em tempo real.



Fonte: Basketball Stats Assistant (2025).

A análise de desempenho no basquete tem evoluído com o apoio de tecnologias como a do *Basketball Stats Assistant* apresentado anteriormente, que permitem automatizar a coleta de métricas que auxiliam no desenvolvimento de táticas e de treinos individuais para cada tipo de atleta. A combinação entre dados objetivos (como distâncias percorridas e ângulos articulares) e dados estatísticos (como aproveitamento de arremessos) potencializa a compreensão do comportamento do atleta durante o jogo e permite

intervenções mais precisas por parte da comissão técnica.

2.3 BIOMECÂNICA

A biomecânica é a ciência que estuda os movimentos do corpo humano aplicando os princípios da mecânica. Ela busca compreender as forças internas e externas que atuam sobre o corpo e os efeitos dessas forças no desempenho atlético e na prevenção de lesões. Segundo Amadio e Serrão (2011), "o objetivo central da biomecânica é o estudo do movimento humano, procedendo sua análise a partir das leis da física, levando em consideração as características do aparelho locomotor".

No contexto esportivo, a biomecânica desempenha um papel fundamental na análise e otimização dos movimentos dos atletas. Por meio da avaliação biomecânica, é possível identificar padrões de movimento inadequados, propor correções técnicas e desenvolver programas de treinamento personalizados, visando melhorar a eficiência e reduzir o risco de lesões. Brandina (2018) destaca que "a biomecânica aplicada ao esporte evidencia as estratégias usadas pelo aparelho locomotor para garantir maior eficiência nos gestos motores esportivos".

Conforme destaca Teferi e Endalew (2020), a biomecânica esportiva pode ser dividida em cinemática — estudo do movimento sem considerar as forças (como ângulos, velocidades e acelerações) — e cinética, que analisa as forças envolvidas nos gestos esportivos. A integração entre essas abordagens permite aos treinadores e analistas identificar pontos fortes e falhas nos movimentos dos atletas.

Knudson (2007) argumenta que, embora fatores fisiológicos, cognitivos e anatômicos influenciem o desempenho, a técnica de execução é o principal aspecto que a biomecânica busca aprimorar. Assim, a análise biomecânica torna-se relevante em esportes nos quais a qualidade técnica dos gestos é determinante — como o arremesso no basquetebol.

Segundo a revisão sistemática de Teferi e Endalew (2020), métodos quantitativos e qualitativos são utilizados na prática. A análise qualitativa, geralmente realizada por observação visual, permite identificar padrões técnicos, enquanto a análise quantitativa recorre a instrumentos como câmeras de alta velocidade, plataformas de força, acelerômetros e softwares de rastreamento para coletar dados objetivos e reproduzíveis. A combinação dessas abordagens gera resultados mais robustos e contextualizados.

Aplicações práticas da análise biomecânica no desempenho esportivo com base nas

referências utilizadas sobre o tema:

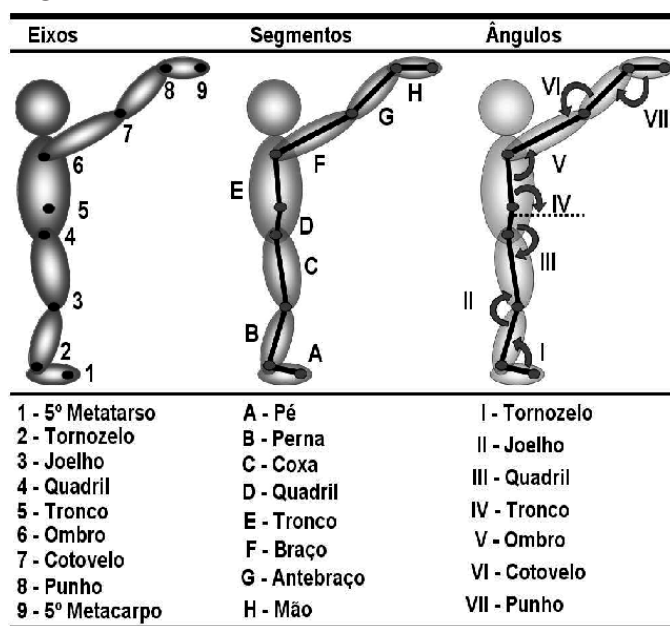
- Identificação de falhas técnicas sutis, imperceptíveis ao olho nu, que comprometem a eficiência do movimento;
- Fornecimento de feedbacks baseados em dados objetivos, permitindo correções específicas e fundamentadas;
- Monitoramento da progressão de habilidades motoras ao longo do tempo, comparando vídeos e métricas biomecânicas;
- Prevenção de lesões, por meio da análise de sobrecargas articulares e padrões de movimento inadequados;
- Otimização do gesto esportivo, ajustando ângulos e tempos de ativação muscular para maior aproveitamento mecânico;

2.3.1 Biomecânica no Basquetebol

No basquetebol, a biomecânica é aplicada para analisar movimentos específicos, como arremessos, saltos e deslocamentos laterais. Estudos indicam que a análise biomecânica pode contribuir para a melhoria da técnica de arremesso, aumentando a precisão e eficiência dos jogadores. Conforme Caseiro (2015), “o lançamento em suspensão permite ao atleta aumentar o ângulo de entrada da bola no cesto, sendo considerado por diferentes investigadores como um dos movimentos mais eficazes no basquetebol”.

A biomecânica no basquetebol é uma ferramenta fundamental para analisar movimentos específicos, como arremessos, saltos e deslocamentos, contribuindo para a melhoria da técnica e da eficiência dos jogadores. Dentre esses movimentos, o arremesso com salto (*jump shot*) tem sido amplamente estudado por sua relevância no jogo moderno. Segundo Caseiro (2015), este tipo de lançamento é considerado um dos mais eficazes por permitir ao atleta aumentar o ângulo de entrada da bola no cesto. A execução bem-sucedida do *jump shot* depende de diversos fatores biomecânicos, onde, conforme apontado por Okazaki et al. (2015), a trajetória da bola, as fases do movimento e a coordenação entre os membros superiores e inferiores influenciam diretamente a sua eficácia.

Figura 2 – Modelo biomecânico utilizado na análise do arremesso no basquetebol.



Fonte: Okazaki et al. (2015).

Principais erros biomecânicos cometidos no arremesso com base em Physiopedia (2025):

- **Alinhamento incorreto do corpo:** inclinação excessiva do tronco ou rotações desnecessárias comprometem o equilíbrio e a precisão do arremesso.;
- **Extensão insuficiente do cotovelo:** limita a altura do lançamento da bola e a transferência de força, afetando o alcance e a precisão.;
- **Liberação fora do tempo ideal:** soltar a bola antes ou depois do pico do salto diminui o controle do arremesso, influenciando na precisão do mesmo;
- **Baixo ângulo de arremesso:** dificulta a trajetória parabólica ideal e reduz a taxa de acerto;
- **Mau posicionamento dos pés:** compromete a base do salto e a estabilidade durante o movimento.

Esses erros impactam diretamente a eficiência dos arremessos e aumentam o risco de lesões, principalmente em articulações como ombro e joelho. A análise biomecânica permite detectar esses desvios e propor correções baseadas em dados objetivos. Segundo Knudson (2007), padrões técnicos otimizados melhoram o desempenho e prolongam a carreira esportiva. Lees (2002) reforça que o uso da biomecânica permite maior precisão nas correções técnicas, superando a subjetividade da observação visual comum em treinos.

2.4 VISÃO COMPUTACIONAL

A visão computacional é um campo da () que permite que máquinas interpretem e compreendam imagens e vídeos do mundo real. Essa tecnologia utiliza algoritmos avançados para processar dados visuais, identificando padrões, objetos e movimentos em tempo real (AWS, 2024).

Com o avanço das técnicas de aprendizado profundo, especialmente as (), a () passou a atingir níveis de precisão melhores na execução de tarefas complexas. Modelos de deep learning são utilizados em tarefas como detecção de objetos, segmentação semântica, reconhecimento facial e análise de movimento (GETGURU, 2024).

Hoje, a () está presente em diversos setores. Na indústria, é usada para inspeção automatizada de peças; na medicina, auxilia na interpretação de exames de imagem; em cidades inteligentes, contribui para a vigilância e controle de tráfego; e na robótica, permite a navegação autônoma de dispositivos móveis (INGRAM MICRO, 2024).

2.4.1 Visão computacional no basquete

No basquetebol, a () tem sido empregada em diversas frentes. A NBA, por exemplo, utiliza câmeras de rastreamento que monitoram, com alta precisão, a movimentação dos jogadores e da bola durante os jogos. Isso permite a geração de estatísticas avançadas que auxiliam tanto na análise tática quanto na preparação física dos atletas (ULTRALYTICS, 2024).

Além disso, pesquisadores têm desenvolvido sistemas que combinam () com realidade aumentada para melhorar a qualidade dos treinos. Um dos focos é o arremesso livre: utilizando sensores e câmeras, sistemas consolidados no mercado como o **HomeCourt App** e o **Noah Basketball System** fornecem feedback em tempo real sobre a postura corporal, ângulo do braço e trajetória da bola. Esses dados ajudam o atleta a ajustar sua mecânica de arremesso e a melhorar o desempenho ao longo do tempo (LIN et al., 2021).

O HomeCourt é um aplicativo para dispositivos móveis (iOS e Android) que utiliza a câmera do smartphone e () para funcionar como um treinador de basquete pessoal. Ele rastreia uma vasta gama de métricas de arremesso, como tempo e ângulo de lançamento, velocidade da bola e altura do salto. O sistema oferece feedback auditivo durante a prática

e relatórios detalhados com vídeos e estatísticas após o treino, sendo uma ferramenta acessível para jogadores de todos os níveis (HomeCourt, 2025).

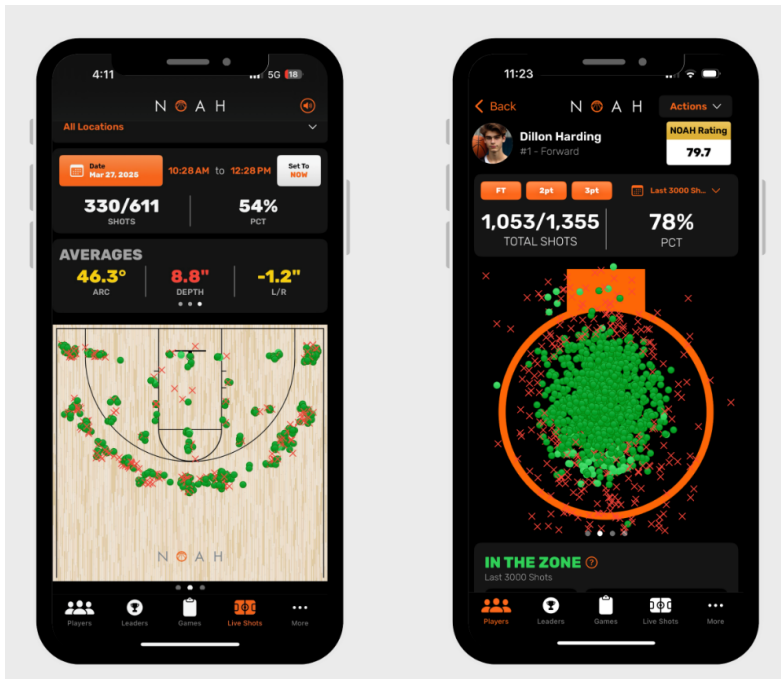
O Noah Basketball System é uma solução baseada em hardware, composta por um sensor de alta precisão instalado acima da cesta. Seu foco principal é otimizar o arco do arremesso, fornecendo feedback instantâneo e audível sobre se o atleta atingiu o arco ideal de 45 graus, além de medir a profundidade e a consistência da entrada da bola no aro. Os dados são enviados para um aplicativo, permitindo análises detalhadas do desempenho. É uma ferramenta amplamente utilizada em centros de treinamento e por equipes profissionais que buscam o mais alto nível de precisão, sendo utilizada inclusive por 28 das 30 franquias da National Basketball League (NBA), e por várias faculdades que competem nacionalmente no esporte. (Noah Basketball, 2025).

Figura 3 – Hardware usado para a captação de arremessos.



Fonte: Noah Basketball (2025).

Figura 4 – Aplicativo do Noah Basketball.



Fonte: Noah Basketball (2025).

As ilustrações acima, em sequência, mostram o Hardware utilizado acima da tabela de basquete, que faz a captura dos arremessos, conseguindo assim por meio de () mapear o local exato do arremesso no aro. Com a captura dos arremessos, é possível fazer a análise dos mesmos pelo aplicativo, podendo assim ver dados como porcentagem de acurácia e angulação média do arremesso.

2.4.2 Processamento de Imagens

O processamento de imagens é um conjunto de técnicas aplicadas a imagens digitais com o objetivo de realçar características visuais, reduzir ruídos ou ajustar propriedades para facilitar a interpretação por sistemas computacionais. As operações típicas incluem nitidez, suavização, filtragem, binarização, realce de contraste e transformações geométricas.

Segundo a NVIDIA (2023), “o processamento de imagens usa algoritmos para alterar imagens, incluindo nitidez, suavização, filtragem ou aprimoramento”. Já a (), por outro lado, “não altera uma imagem, mas dá sentido ao que vê e realiza tarefas, como a rotulagem”. A empresa ainda destaca que o processamento de imagem pode ser usado como etapa preliminar, preparando a imagem para ser interpretada por modelos de ().

Em muitos sistemas modernos, especialmente os que utilizam (), o pré-processamento

da imagem é essencial para garantir que os dados visuais estejam padronizados e livres de ruídos que possam prejudicar o desempenho dos algoritmos. Ferramentas como () e MATLAB são amplamente utilizadas nesse processo.

Dentre as principais técnicas aplicadas no processamento de imagens, destaca-se a **equalização de histograma**, que visa melhorar o contraste das imagens, redistribuindo os níveis de intensidade dos pixels e tornando as características visuais mais perceptíveis para os algoritmos de detecção (OpenCV, 2025). Essa técnica é especialmente útil em ambientes com iluminação desigual, como ginásios esportivos, que é o caso deste trabalho.

Outra abordagem adotada é a **suavização da imagem**, geralmente feita com filtros como o *Gaussian Blur*, que reduz ruídos de alta frequência e variações bruscas de cor que podem interferir na segmentação do corpo do atleta (OpenCV, 2025). Essa técnica atua como um pré-filtro, garantindo que a entrada da rede neural (()) esteja mais limpa e menos sujeita a falsos positivos.

Além disso, é aplicada a **normalização de brilho e contraste**, por meio de transformações lineares na intensidade dos pixels, com o objetivo de uniformizar a exposição da imagem, garantindo consistência entre diferentes vídeos analisados (OpenCV, 2025). Isso contribui para a padronização das amostras de entrada.

Figura 5 – Correção de contraste com OpenCV.



Fonte: OpenCV (2025).

Esses procedimentos são executados em sequência como parte do **pipeline de pré-processamento**, visando fornecer imagens ideais para os modelos de () e posterior extração de características biomecânicas. Ferramentas como a biblioteca () são utilizadas amplamente para essas finalidades, dada sua robustez e variedade de algoritmos implementados (OpenCV, 2025).

2.5 MODELOS DE DETECÇÃO DE POSES E ESTRUTURA CORPORAL

A detecção de poses humanas é uma área da () voltada à identificação automática das articulações corporais em imagens ou vídeos. Através dessa tecnologia, é possível gerar uma estrutura esquelética baseada em pontos-chave (*keypoints*), que permite a análise precisa de postura, movimentação e biomecânica do corpo (VISO.AI, 2023).

Segundo Boesch (2023), a tarefa de estimativa de pose pode ser classificada em dois grupos principais: detecção em 2D, que localiza as articulações no plano da imagem, e detecção em 3D, que busca reconstruir a estrutura corporal tridimensional. Embora a detecção em 3D ofereça maior contexto espacial, a abordagem 2D permanece amplamente utilizada devido à sua eficiência e maior disponibilidade de *datasets* anotados.

Entre os principais modelos atuais, destacam-se:

- **OpenPose:** Um dos primeiros frameworks de código aberto capazes de detectar múltiplas pessoas em tempo real, mapeando pontos da face, mãos e corpo de forma simultânea. Utiliza uma abordagem *bottom-up*, identificando os pontos-chave primeiro para depois agrupá-los por indivíduo (Boesch, 2023; PAPERSWITHCODE, 2024).
- **BlazePose:** Criado pelo Google, é um modelo otimizado para dispositivos móveis. Utiliza uma malha corporal com 33 pontos e apresenta excelente desempenho em aplicações que requerem baixa latência, como jogos e aplicativos de condicionamento físico (AIP PUBLISHING, 2023).
- **PoseNet:** Também desenvolvido pelo Google, é amplamente usado por sua leveza e por funcionar diretamente em navegadores com imagens RGB, o que o torna prático para aplicações web e mobile (CODERADE.IO, 2024).

Esses modelos têm sido empregados na realidade aumentada, fisioterapia digital e na análise esportiva, permitindo monitoramento técnico em múltiplos esportes, tanto coletivos quanto individuais com uso de câmeras comuns. Abaixo, por meio de uma tabela, serão abordados os principais aspectos desses modelos.

2.6 REDES NEURAIAS CONVOLUCIONAIS

As redes neurais convolucionais (CNNs) surgiram como uma poderosa arquitetura

Tabela 1 – Comparação entre modelos de detecção de pose corporal.

Critério	OpenPose	BlazePose	PoseNet
Desenvolvedor	Carnegie Mellon University	Google	Google
Número de pontos detectados	Variável (face, mãos e corpo completo)	33 pontos corporais	17 pontos corporais
Abordagem técnica	Bottom-up	Top-down	Top-down
Aplicações principais	Detecção múltipla simultânea, análise detalhada de movimentos	Jogos, aplicativos móveis e condicionamento físico	Aplicações leves em navegadores e dispositivos móveis
Precisão e desempenho	Alta precisão, maior consumo computacional	Alta precisão, otimizado para baixa latência	Precisão moderada, desempenho otimizado
Ambientes de uso	Ambientes controlados e pesquisa avançada	Aplicações móveis e interativas	Aplicações web e mobile leves
Disponibilidade e popularidade	Alta (open source)	Alta (amplamente usado)	Alta (amplamente usado)

Fonte: O Autor, adaptado de Boesch (2023), PapersWithCode (2024), AI Publishing (2023) e Coderade.io (2024).

dentro do campo de aprendizado profundo, especialmente eficazes para o reconhecimento de padrões em imagens, vídeos e dados com estrutura espacial. Ao contrário das redes totalmente conectadas (MLPs), as () exploram o princípio da localidade espacial e da invariância translacional, o que permite reduzir drasticamente a quantidade de parâmetros e melhorar a generalização dos modelos. Isso é possível graças à aplicação de filtros convolucionais que percorrem localmente a entrada, extraindo características relevantes em diferentes escalas e posições (GOODFELLOW; BENGIO; COURVILLE, 2016).

Essas redes são compostas, em sua forma mais básica, por camadas convolucionais, funções de ativação (como ReLU), camadas de pooling (que reduzem a dimensionalidade), e camadas totalmente conectadas ao final do modelo para classificação ou regressão. Em contextos visuais, por exemplo, as primeiras camadas de uma () aprendem a detectar bordas e texturas simples, enquanto camadas mais profundas passam a identificar formas e padrões mais complexos, como olhos ou rostos em imagens humanas (ARORA et al.,

2021).

A capacidade hierárquica de representação é uma das grandes forças das (). Como mencionado por Arora et al. (2021), os filtros nas primeiras camadas capturam características de baixo nível (e.g., bordas), enquanto as camadas subsequentes combinam essas informações para formar representações mais abstratas. O processo de aprendizado ocorre por meio da retropropagação (backpropagation), otimizando os pesos dos filtros conforme os erros na saída do modelo (KARPATHY; LI, 2023).

Além disso, a natureza parametrizada dos filtros permite que uma () reconheça um objeto independentemente da sua posição na imagem, promovendo o conceito de invariância espacial, essencial em aplicações como reconhecimento facial, classificação de imagens médicas e detecção de objetos.

2.6.1 YOLO (You Only Look Once)

YOLO (*You Only Look Once*) é uma arquitetura de rede neural para detecção de objetos em tempo real, proposta por Redmon et al. (2016). Ao contrário de abordagens tradicionais como R-CNN, que realizam múltiplos estágios no processo de detecção, () reformula o problema como uma única tarefa de regressão, mapeando diretamente a imagem para caixas delimitadoras e probabilidades de classe. Segundo os autores, “reformulamos a detecção de objetos como um problema de regressão única, direto dos pixels da imagem para as coordenadas das caixas e as probabilidades das classes” (REDMON et al., 2016).

A estrutura da rede divide a imagem de entrada em uma grade (geralmente 7×7). Cada célula da grade é responsável por prever um número fixo de *bounding boxes* e suas respectivas pontuações de confiança, além da classificação dos objetos presentes. A arquitetura original da () é composta por 24 camadas convolucionais e duas totalmente conectadas, utilizando a rede *Darknet* como base (CARVALHO et al., 2022).

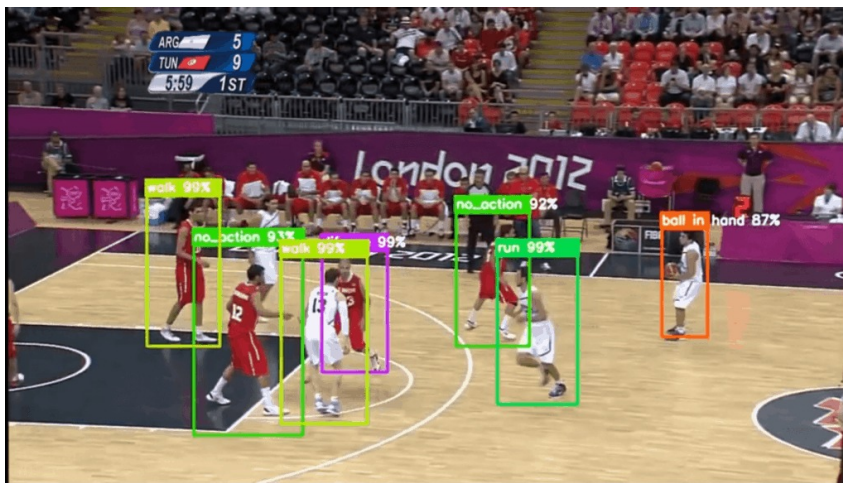
O guia técnico da Encord (2024) destaca que “a () processa a imagem inteira de uma vez, permitindo que a rede aprenda representações contextuais mais globais e reduza erros comuns, como falsas detecções em regiões de fundo”. Essa abordagem global é uma das principais vantagens da () em relação a métodos que se concentram apenas em regiões localizadas da imagem.

Nos testes realizados pelos autores originais, o modelo atingiu desempenho superior

a 45 () com latência inferior a 25 milissegundos, possibilitando aplicações em vídeo em tempo real (REDMON et al., 2016).

Além disso, Carvalho et al. (2022) explicam que a () combina os dois processos fundamentais da detecção de objetos — classificação e localização — em uma única etapa, tornando o processo mais eficiente. Eles afirmam que “ () é capaz de responder duas perguntas: quais objetos estão na imagem? E onde eles estão posicionados?” (CARVALHO et al., 2022).

Figura 6 – Uso da YOLO em partidas da NBA.



Fonte: FRANCI, Simone, (2024).

2.6.1.1 Evolução das Versões do YOLO

Desde sua criação, o framework () evoluiu significativamente, com novas versões que aprimoraram a velocidade, a precisão e a versatilidade. As versões desenvolvidas pela Ultralytics mais conhecidas são **YOLOv5**, **YOLOv8**, e a iteração estável mais recente, o **YOLOv11**. O YOLOv8 expandiu o framework para incluir nativamente múltiplas tarefas, como a estimacão de pose, enquanto o YOLOv11 focou em otimizar a eficiência, buscando maior precisão com um menor número de parâmetros que seu antecessor (Ultralytics, 2024).

A tabela 2 a seguir compara as três principais versões do (), com informações retiradas do site do desenvolvedor:

A arquitetura do YOLOv8, ilustrada de forma simplificada na figura, é a base para

Tabela 2 – Comparativo entre as Principais Versões do YOLO (Ultralytics)..

Característica	YOLOv5	YOLOv8	YOLOv11
Arquitetura (Head)	Baseada em Âncoras (Anchor-Based)	Livre de Âncoras (Anchor-Free)	Mantém a abordagem Livre de Âncoras com backbone e neck aprimorados.
Tarefas Suportadas	Foco principal em Detecção de Objetos.	Suporte nativo a Detecção, Segmentação, Estimação de Pose, Rastreamento e Classificação.	Mantém e aprimora a gama de tarefas do YOLOv8, incluindo Detecção Orientada (OBB).
Principal Avanço	Popularização do balanço entre alta velocidade e boa precisão.	Introdução de um framework multitarefa com API unificada e modelos pré-treinados.	Foco em eficiência computacional: alcançar maior precisão com menos parâmetros que o YOLOv8.
Modelo para Pose	Não possui um modelo oficial na versão original.	Inclui modelos YOLOv8-Pose pré-treinados e prontos para uso.	Inclui modelos YOLOv11-Pose pré-treinados, continuando o suporte nativo à tarefa.

Fonte: O Autor, adaptado de Ultralytics (2024).

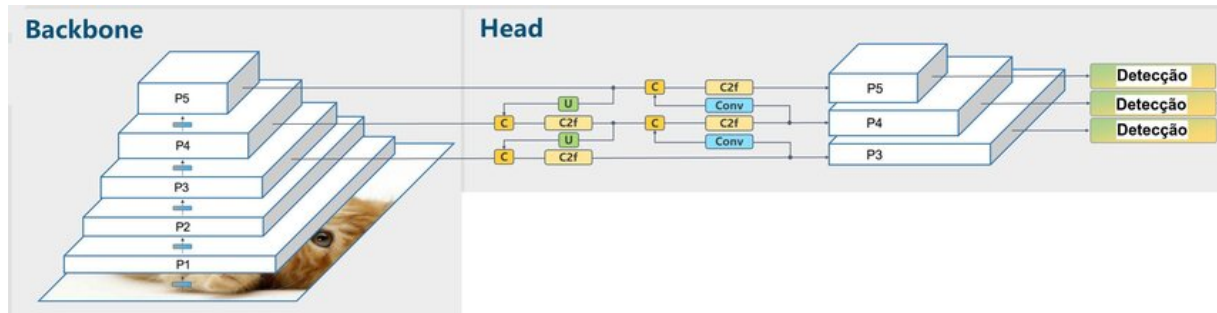
o seu alto desempenho e pode ser dividida em três componentes principais que segundo Ultralytics (2025) são:

Backbone (Coluna Vertebral): É a primeira parte da rede, responsável por extrair características importantes da imagem de entrada. Ele processa a imagem em várias camadas (P1 a P5), onde a resolução espacial diminui, mas a complexidade semântica da informação aumenta. Características de diferentes níveis (P3, P4, P5) são então enviadas para a próxima etapa.

Neck (PESCOÇO): Atua como uma ponte, mesclando e combinando as características que vêm do *Backbone*. Utilizando uma estrutura de Agregação de Caminho (Path Aggregation Network - PANet), ele funde a informação semântica das camadas profundas (que sabem “o que” é um objeto) com a informação espacial das camadas mais rasas (que sabem exatamente “onde” ele está). Isso é fundamental para detectar objetos de diferentes tamanhos.

Head (Cabeça): É a parte final, responsável por realizar as predições. Ela recebe os mapas de características refinados do *Neck* e utiliza múltiplas “cabeças de detecção”, cada uma especializada em uma escala (objetos pequenos, médios e grandes). A saída final são as caixas delimitadoras, a classe e a confiança de cada objeto detectado — ou, no caso do YOLO-Pose, as coordenadas dos keypoints.

Figura 7 – Arquitetura da YOLOv8 simplificada.



Fonte: Jocher et al. (2023).

2.6.1.2 YOLO-Pose para Análise de Keypoints

Para a análise biomecânica de arremessos, que depende da identificação precisa de articulações, as variantes de estimação de pose do () são as ferramentas mais adequadas. Tanto o **YOLOv8-Pose** quanto o **YOLOv11-Pose** são modelos construídos para, além de localizar uma pessoa na imagem, estimar a posição de seus **17 keypoints corporais** (Ultralytics, 2024).

Esses pontos-chave, treinados no dataset COCO-pose, incluem as principais articulações para a análise de movimento, como ombros, cotovelos, punhos, quadris, joelhos e tornozelos. A saída do modelo fornece as coordenadas (x, y) e um score de confiança para cada keypoint detectado, permitindo a construção de um esqueleto digital do atleta para posterior cálculo de ângulos e trajetórias.

A escolha entre os diferentes tamanhos de modelos disponíveis (nano, small, medium, etc.) é uma decisão fundamental do projeto, permitindo um balanço entre a velocidade de inferência e a precisão da detecção, um fator crucial a ser definido dependendo dos requisitos computacionais do hardware disponível, como uma () NVIDIA. (Ultralytics, 2024).

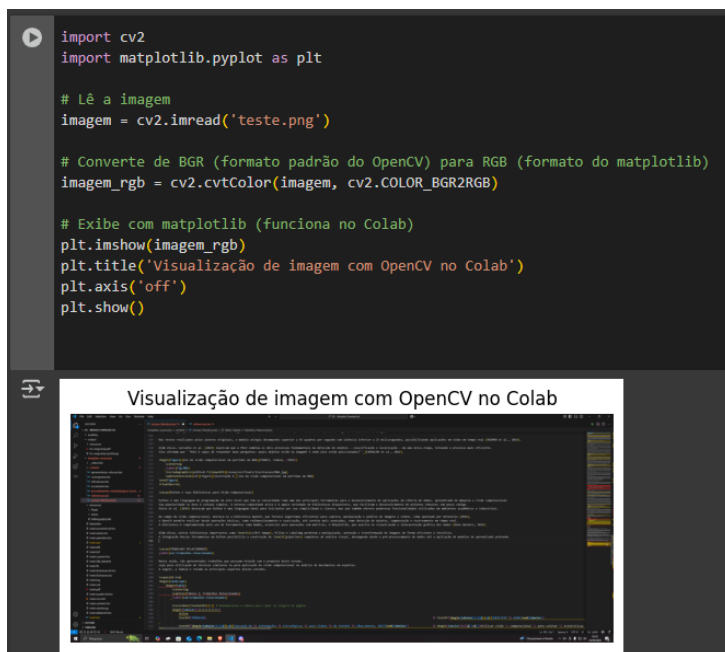
2.7 PYTHON E SUAS BIBLIOTECAS PARA VISÃO COMPUTACIONAL

Python é uma linguagem de programação de alto nível que tem se consolidado como uma das principais ferramentas para o desenvolvimento de aplicações em ciência de dados, () e (). Sua popularidade se deve à sintaxe simples, à extensa comunidade ativa e à ampla variedade de bibliotecas disponíveis, que facilitam o desenvolvimento de projetos robustos com pouco código. Paiva et al. (2019) destacam que Python é uma

linguagem ideal para iniciantes por sua simplicidade e clareza, mas que também oferece poderosas funcionalidades utilizadas em ambientes acadêmicos e industriais.

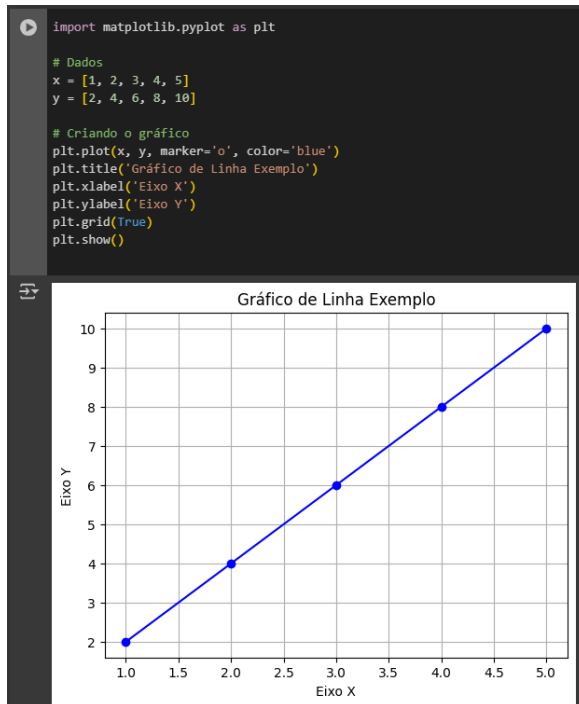
No campo da (), destaca-se a biblioteca (), que fornece algoritmos eficientes para captura, manipulação e análise de imagens e vídeos. Como apontado por Antonello (2016), o () permite realizar desde operações básicas, como redimensionamento e suavização, até tarefas mais avançadas, como detecção de objetos, segmentação e rastreamento em tempo real. A biblioteca é complementada pelo uso de ferramentas como NumPy, essencial para operações com matrizes, e Matplotlib, que auxilia na visualização e interpretação gráfica dos dados (Data Hackers, 2019).

Figura 8 – Uso da biblioteca OpenCV e Matplotlib no Google Colab.



Fonte: O Autor.

Figura 9 – Uso da biblioteca Matplotlib.



Fonte: O Autor.

Além disso, outras bibliotecas importantes como *scikit-image*, Pillow e LabelImg permitem a manipulação, anotação e transformação de imagens de forma eficiente e intuitiva. A integração dessas ferramentas em Python possibilita a construção de *pipelines* completos de análise visual, abrangendo desde o pré-processamento de dados até a aplicação de modelos de aprendizado profundo.

2.8 TRABALHOS RELACIONADOS

Nesta seção, são apresentados trabalhos que possuem relação com a proposta deste estudo, seja pela utilização de técnicas similares ou pela aplicação da () na análise de movimentos em esportes. A seguir, cada um dos trabalhos é abordado individualmente, seguido de uma tabela comparativa ao final.

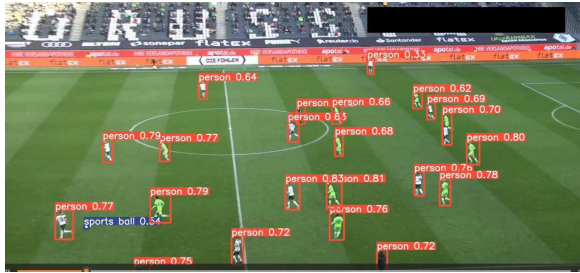
2.8.1 Geração de Informações Estratégicas para Clubes de Futebol (Nascimento, 2023)

O trabalho de Nascimento (2023) propõe um sistema de () que utiliza o algoritmo () para detectar jogadores e rastrear a trajetória da bola em partidas de futebol. O objetivo

principal é fornecer estatísticas estratégicas, como contagem de passes e ocupação espacial, auxiliando a comissão técnica durante análises táticas.

A arquitetura do sistema inclui módulos de detecção, rastreamento e análise. Os testes realizados mostraram precisão variando entre 40% e 90% na contagem de passes, com bons níveis de eficiência na trajetória da bola.

Figura 10 – Detecção de jogadores em campo utilizando YOLO.



Fonte: Nascimento, 2023.

2.8.2 Classificação de Golpes de Karate (Teixeira, 2024)

Teixeira (2024) desenvolveu um sistema de reconhecimento de ações voltado à classificação de golpes de Karate. O sistema emprega a detecção de *keypoints* corporais com base em (), utilizando as bibliotecas OpenPose, YOLOv5 e algoritmos de classificação como SVM.

Os dados capturados durante os movimentos são processados em tempo real para identificar padrões específicos de golpes. Os testes apontaram precisão de 99,45% e latência de processamento adequada para aplicações em treinamento e arbitragem automatizada.

Figura 11 – Detecção de *keypoints* durante execução de golpes.



Fonte: Teixeira, 2024.

2.8.3 Biomechanical Analysis of Shooting Performance (Pan et al., 2021)

O estudo de Pan et al. (2021) investigou o desempenho de arremessos no basquete por meio de uma análise biomecânica baseada em (). O sistema faz uso de sensores Kinect Azure para capturar o movimento dos atletas e aplica modelos de esqueleto para medir postura, ângulo de lançamento e tempo de execução.

O objetivo da pesquisa foi entender como fatores como postura corporal e coordenação influenciam a precisão do arremesso. Os resultados mostraram que diferenças significativas entre gêneros foram observadas e que a postura está diretamente ligada ao desempenho.

Figura 12 – Modelo esquelético utilizado para análise biomecânica.



Fonte: Pan et al., 2022.

Ao final dessa seção, a Tabela 3 resume os principais aspectos desses estudos abordados.

Ambos os estudos demonstram que a () pode ser aplicada com sucesso na análise esportiva, reforçando a viabilidade do presente trabalho e oferecendo referências metodológicas para a implementação e validação do sistema de análise de arremesso no basquete.

As pesquisas bibliográficas a seguir foram feitas dentro do Google Acadêmico em Março de 2025, a plataforma Sci-Hub também foi utilizada para procurar artigos que são pagos na plataforma do Google, porém não foram encontradas amostras na plataforma. Na tabela 4, podemos ver detalhes de resultados obtidos nas pesquisas avançadas de trabalhos relacionados, todos os termos foram pesquisados utilizando como pesquisa palavras utilizadas no título de artigos, trabalhos que utilizam a visão computacional tiveram um maior número de resultados, já trabalhos com o tema () em específico foram mais difíceis de se encontrar.

Tabela 3 – Trabalhos Relacionados.

TRABALHO	OBJETIVO GERAL	TECNOLOGIAS	MÉTRICAS E VALIDAÇÃO	QUESTÃO DE PESQUISA	RESULTADOS ALCANÇADOS
Geração de Informações Estratégicas para Clubes de Futebol (Nascimento, 2023)	Utilizar visão computacional para coletar estatísticas em partidas de futebol, como contagem de passes e trajetória da bola.	YOLO para detecção de jogadores e rastreamento da bola; Redes Neurais Convolucionais (CNNs).	Precisão variando entre 40% e 90% para contagem de passes; Eficiência na trajetória da bola.	Como aplicar visão computacional para coletar estatísticas de futebol de forma precisa?	Implementação bem-sucedida da tecnologia, com impacto na análise tática.
Classificação de Golpes de Karate (Teixeira, 2024)	Utilizar visão computacional para classificar golpes de Karate e auxiliar na arbitragem e treinamento.	YOLO para detecção de keypoints; Redes Neurais Convolucionais (CNNs); Algoritmo SVM.	FPS e latência por quadro como métricas; Precisão de 99,45% no recebimento de golpes.	Como aplicar aprendizado de máquina para reconhecimento e classificação de golpes no Karate?	Modelo eficiente para análise de golpes, útil para arbitragem automatizada.
Biomechanical Analysis of Shooting Performance for Basketball Players (Pan et al., 2021)	Analisar biomecanicamente o desempenho de arremesso no basquetebol com visão computacional.	Kinect Azure para captura de movimento; Processamento de dados cinemáticos.	Análise estatística das diferenças entre gêneros no arremesso; Correlação entre postura e eficiência.	Como a visão computacional pode contribuir para a análise biomecânica do arremesso?	Resultados mostraram que a coordenação e a postura influenciam a precisão do arremesso.

Fonte: O Autor.

Tabela 4 – Resultados das pesquisas por termos no título de artigos.

Termos Pesquisados	Período de Tempo	Resultados Obtidos
Visão computacional, esporte, yolo	2005 - 2025	50 resultados
Basquete, visão computacional	2005 - 2025	Nenhum resultado
Yolo, esporte	2005 - 2025	Nenhum resultado
Yolo, sports	2005 - 2025	7 resultados
Computer vision, basketball	2005 - 2025	22 resultados

Fonte: O Autor.

3 PROCEDIMENTOS METODOLÓGICOS E TÉCNICOS

Neste capítulo do trabalho será abordada a caracterização da metodologia de pesquisa, questões de pesquisa, como será a aplicação da metodologia e o desenvolvimento do sistema computacional, além de mostrar como será a análise de resultados e a população e amostra do trabalho.

3.1 CARACTERIZAÇÃO DA METODOLOGIA DE PESQUISA

O presente trabalho se caracteriza como uma metodologia aplicada, tendo uma abordagem quantitativa, buscando atingir seus objetivos de forma descritiva e exploratória através de procedimentos exploratórios e de um estudo de caso.

Segundo Gerhardt e Silveira (2009, p. 35), a pesquisa aplicada visa gerar conhecimentos para aplicação prática, dirigidos à solução de problemas específicos, buscando responder às necessidades concretas de organizações, instituições ou grupos sociais. Em complemento, Fonseca (2002) destaca que a pesquisa aplicada está orientada para transformar a realidade de forma direta, aplicando conhecimentos científicos já estabelecidos.

No que se refere à abordagem quantitativa, Gerhardt e Silveira (2009, p. 33) afirmam que esta traduz em números as informações e opiniões, classificando-as e analisando-as de forma objetiva, buscando generalizações estatísticas. Por explorar dados numéricos como a angulação e rotatividade do braço no movimento do arremesso, esse trabalho se caracteriza como quantitativo em sua abordagem.

A pesquisa exploratória, de acordo com Gil (2008, p. 27), tem como finalidade proporcionar maior familiaridade com o problema, visando torná-lo mais explícito ou construir hipóteses, sendo especialmente útil quando o fenômeno é pouco conhecido. Já a pesquisa descritiva, conforme Gil (2008, p. 28), objetiva descrever as características de determinada população ou fenômeno, bem como estabelecer relações entre variáveis. Esse trabalho se caracteriza como exploratório e descritivo, pois o mesmo interpreta dados biomecânicos para executar o estudo, além de abordar um ramo que hoje está em desenvolvimento no Brasil que é a Análise de Desempenho Esportiva.

No que tange aos procedimentos, Gerhardt e Silveira (2009, p. 36) indicam que a pesquisa experimental é caracterizada pela manipulação de variáveis para observar os efeitos provocados, sendo amplamente empregada em estudos controlados. A manipulação

de variáveis dentro de um arremesso fazem com que este trabalho se caracterize como uma pesquisa experimental. O estudo de caso, por sua vez, segundo Gil (2008, p. 57), busca uma análise profunda e exaustiva de um ou poucos objetos, proporcionando um conhecimento abrangente e detalhado da realidade investigada. O fator determinante para este trabalho ser um estudo de caso, se dá pelo fato de ser executado em uma equipe específica, sendo estudado de início somente com esse número pequeno de jogadores, para caso haja êxito, ser posteriormente evoluído a um estudo para outras equipes também.

3.2 QUESTÕES DE PESQUISA

O protótipo proposto é capaz de capturar, processar e analisar biomecanicamente, com base em visão computacional, os movimentos do arremesso no basquetebol, fornecendo diagnósticos para auxiliar atletas e treinadores na correção do gesto de arremesso?

3.3 APLICAÇÃO DA METODOLOGIA

Nesta subseção, são descritas as etapas adotadas para o desenvolvimento da metodologia deste trabalho. As atividades englobam desde a preparação do ambiente de desenvolvimento e coleta de dados até a implementação dos algoritmos de visão computacional e a validação dos resultados obtidos. Cada fase foi conduzida com o objetivo de garantir a confiabilidade do sistema final e a coerência com os objetivos traçados neste trabalho.

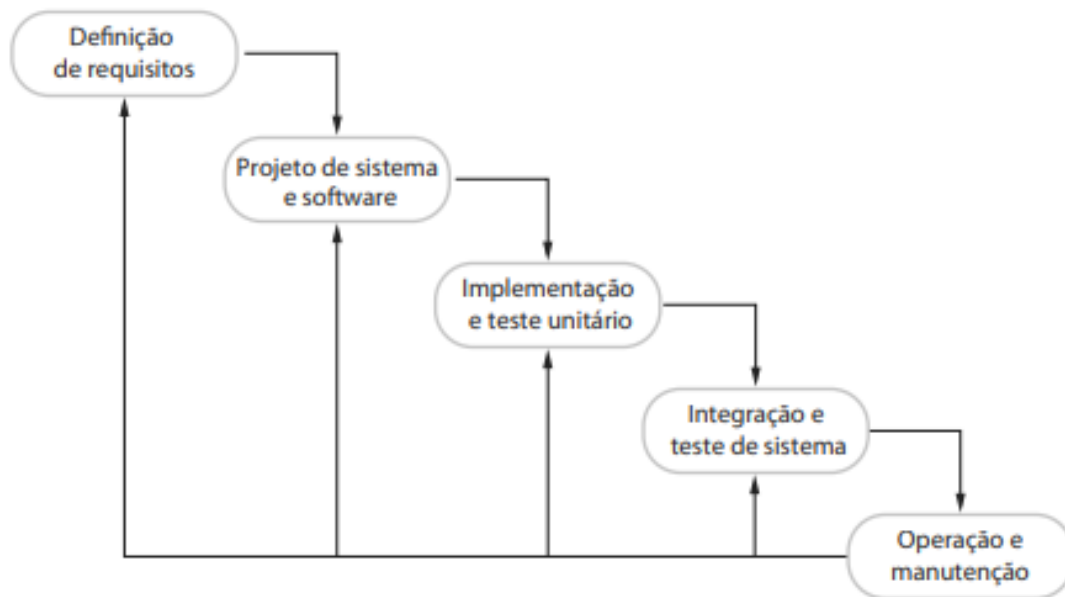
3.3.1 Desenvolvimento da solução proposta na prática

O modelo em cascata é uma das abordagens clássicas para o desenvolvimento de software e foi originalmente descrito por Winston Royce em 1970. Este modelo foi posteriormente estruturado por Ian Sommerville em sua obra clássica “Engenharia de Software” como um processo sequencial em que cada fase depende da entrega da anterior e gera um produto definido.

Conforme ilustrado na Figura 13 (Modelo cascata de Sommerville), o modelo é dividido em cinco fases principais: definição de requisitos, projeto de sistema e software, implementação e teste unitário, integração e teste de sistema, e operação e manutenção.

Figura 13 – Modelo Cascata.

O modelo em cascata



Fonte: Sommerville (2011).

Cada fase possui entregáveis específicos e critérios de entrada e saída bem definidos.

3.3.1.1 Definição de Requisitos

A definição de requisitos é a etapa de desenvolvimento de software que descreve detalhadamente o que o sistema deve fazer e quais restrições ele deve obedecer. De acordo com Sommerville (2011), um requisito é “uma descrição de algo que o sistema deve fazer ou uma propriedade que ele deve ter”. Essa definição engloba tanto funcionalidades esperadas quanto limitações técnicas ou organizacionais que devem ser consideradas na construção do projeto.

Ainda segundo Sommerville (2011), os requisitos funcionais especificam os serviços que o sistema deve oferecer e como ele deve reagir a entradas específicas. Por outro lado, os requisitos não funcionais dizem respeito a restrições sobre os serviços ou funções oferecidas, como requisitos de desempenho, padrões de qualidade, requisitos organizacionais ou de segurança. Para a obtenção desses requisitos, será adotada a técnica de entrevistas semiestruturadas com os usuários finais que serão de início os atletas participantes do estudo de caso. Essa abordagem permitirá captar informações sobre as necessidades práticas dos usuários, possibilitando uma definição mais precisa e contextualizada dos requisitos

do sistema.

3.3.1.2 Projeto de Sistema e Software

A fase de projeto de sistema e software visa transformar os requisitos previamente definidos em uma arquitetura viável para implementação. De acordo com Sommerville (2011), o projeto de software é o processo de definir a arquitetura, os componentes e suas interfaces, garantindo que o sistema satisfaça os requisitos especificados. Neste projeto, essa etapa se desdobra em três frentes principais: a coleta das imagens, o pré-processamento com técnicas de normalização e data augmentation, e a definição do algoritmo baseado em ().

A coleta de imagens será feita por meio da captura de vídeos durante treinos e jogos da equipe, com posterior extração de quadros (frames) relevantes utilizando ferramentas de segmentação automática. As filmagens serão feitas com um celular POCO X6 PRO, tendo em sua câmera principal um sensor de 64MP, que irá gravar em 1080p a 30 (), a câmera será posicionada de forma estratégica para capturar os arremessos com clareza, priorizando ângulos laterais e frontais. A curadoria dessas imagens será feita manualmente para garantir que apenas imagens úteis ao treinamento do modelo sejam utilizadas.

O processamento das imagens incluirá ajustes de tamanho (redimensionamento), conversão para escala de cinza quando apropriado, normalização dos valores de pixel (entre 0 e 1) e remoção de ruídos. Para ampliar a base de dados, serão aplicadas técnicas de data augmentation, como rotações, mudanças de brilho e reflexões horizontais, conforme sugerido por Sommerville (2011) ao destacar que a reutilização de componentes e técnicas pode melhorar a confiabilidade do software e reduzir custos de desenvolvimento.

A definição do algoritmo será baseada na arquitetura da YOLOv8s-pose, que é uma abordagem de detecção de objetos em tempo real, focada principalmente na detecção de pontos de referência no corpo humano em computadores com restrições de recursos. A YOLOv8s-pose utiliza como backbone uma versão aprimorada do CSPDarknet, com um neck baseado no PANet e um head adaptado para múltiplas detecções em diferentes escalas. Essa arquitetura é ideal para aplicações como a deste projeto, que exige velocidade e precisão ao identificar arremessos de basquete em imagens. A YOLOv8s-pose apresenta desempenho competitivo mesmo em hardwares com restrições de recursos, e por esse motivo foi escolhida para este projeto, levando em consideração que será utilizada uma (

) GeForce GTX 1650 para o processamento das imagens.

3.3.1.3 Fase de Implementação

A fase de implementação representa o momento em que as decisões definidas nas etapas de projeto e análise se concretizam na construção prática do sistema. Segundo Sommerville (2011), essa etapa está diretamente relacionada à transformação dos modelos em código executável, respeitando os padrões estabelecidos e validando o funcionamento do sistema como um todo.

O treinamento do modelo será realizado a partir das imagens que forem previamente coletadas e anotadas com o auxílio da ferramenta LabelImg. Trata-se de uma aplicação gráfica de código aberto, escrita em Python, que permite a marcação manual de regiões de interesse por meio de caixas delimitadoras. Essas anotações são fundamentais para treinar modelos de () supervisionados, como o YOLOv8s-pose, pois indicam a localização e a classe dos objetos presentes nas imagens. O LabelImg possibilita salvar as anotações em formatos compatíveis com diversos frameworks, incluindo o formato .txt utilizado pelo (). Por meio dessa ferramenta, será possível rotular elementos importantes para o trabalho, como a bola de basquete e os segmentos corporais do atleta.

Serão definidos hiperparâmetros iniciais com base em experimentação e na capacidade do hardware disponível, composto por um processador Ryzen 3 3100 e uma () NVIDIA GTX 1650. Espera-se utilizar uma taxa de aprendizado de 0,001, com 100 épocas de treinamento e tamanho de lote de 16 imagens. Para evitar o sobreajuste e promover uma boa generalização do modelo, será aplicada validação cruzada no formato hold-out, com 80% dos dados destinados ao treinamento e 20% à validação.

Durante o processo, também serão aplicadas técnicas de aumento de dados, como espelhamento horizontal, ajustes de brilho e rotação. Isso permitirá expandir a variabilidade do conjunto de dados sem a necessidade de novas coletas, o que é particularmente importante em projetos com volume limitado de amostras. A avaliação do modelo será conduzida utilizando métricas como precisão, revocação, F1-score e média da precisão média (mean Average Precision - mAP), fornecendo uma visão da eficácia da solução e orientando eventuais ajustes futuros.

A linguagem de programação escolhida para o desenvolvimento é Python. Serão utilizadas as bibliotecas () e NumPy para manipulação e processamento das imagens,

além do framework PyTorch para o uso da YOLOv8s-pose. A programação será conduzida na IDE VS Code, que permite integração com ferramentas de versionamento e ambientes virtuais.

O repositório da aplicação será controlado via Git e hospedado no GitHub, garantindo controle de versão, rastreabilidade de mudanças e backup do progresso. Isso também permitirá a integração com pipelines de entrega contínua futuramente, caso a aplicação seja expandida.

3.3.1.4 Fase de integração e testes

A fase de testes é responsável por garantir a qualidade do software e a conformidade com os requisitos especificados. Segundo Pressman (2011), os testes de software devem ser planejados de forma sistemática e aplicados com o propósito de revelar o maior número possível de defeitos, contribuindo para a qualidade e confiabilidade do produto final.

Os testes serão divididos entre a verificação funcional da aplicação desenvolvida em Python e a avaliação do desempenho do modelo de detecção de imagens. O roteiro de testes considera a execução da aplicação com vídeos reais, gravados durante os treinos, que serão analisados quadro a quadro pelo modelo treinado. O objetivo é validar se as detecções ocorrem corretamente, se os dados gerados estão de acordo com os critérios esperados e se a experiência de uso da aplicação permanece fluida. As verificações funcionais incluem a leitura adequada dos arquivos de vídeo, a ativação correta do modelo, a identificação precisa de elementos como a bola de basquete e os membros superiores, além da geração de análises coerentes dos arremessos e do bom desempenho geral do sistema. Parte dos testes será também automatizada com o uso do framework PyTest, focando em funções auxiliares como o pré-processamento das imagens e a validação da integração com arquivos simulados, o que permite assegurar a estabilidade do sistema mesmo após possíveis futuras alterações no código.

3.3.1.5 Operação e manutenção

A fase de operação e manutenção corresponde à última etapa do ciclo de vida no modelo em cascata, sendo responsável por garantir a continuidade do funcionamento do sistema após sua entrega, além de contemplar ajustes, correções de falhas e melhorias que

possam surgir com o uso contínuo da aplicação. De acordo com Pressman (2016), esta etapa não apenas envolve a correção de erros não detectados anteriormente, mas também a adaptação do software às mudanças de ambiente e requisitos, bem como o aprimoramento de sua funcionalidade.

Neste trabalho, serão feitas manutenções para o bom funcionamento do sistema, buscando sempre atualizações que melhorem a eficiência do mesmo. Possivelmente melhorias como a implementação de análises de outros movimentos, como passes ou bloqueios poderão ser feitas com manutenções posteriores a entrega do projeto inicial.

3.3.2 População e Amostra

A população desse estudo é composta por uma equipe de basquete, hoje competindo em âmbito amador, tendo um foco maior na modalidade de basquete 3x3. Atualmente a equipe conta com aproximadamente 15 atletas, onde serão selecionados alguns para a realização dos testes deste trabalho.

A seleção não considerará variáveis como idade, posição em quadra ou nível técnico individual, visto que o objetivo da pesquisa é avaliar a aplicabilidade da ferramenta em um contexto coletivo e real de treinamento.

4 APRESENTAÇÃO DA SOLUÇÃO

Neste capítulo, será abordada a modelagem do sistema proposto, contendo os diagramas e também os requisitos que fazem parte do trabalho.

4.1 REQUISITOS DO SISTEMA

Logo abaixo, será apresentada a tabela 3 contendo os requisitos funcionais do sistema computacional, seguido da tabela 4 com os requisitos não funcionais do mesmo.

Tabela 5 – Requisitos funcionais do sistema computacional.

Número Requisito	Requisito Funcional
01	O sistema deve permitir a importação de vídeos contendo arremessos de basquete para análise biomecânica.
02	O sistema deve realizar a detecção da estrutura corporal do atleta nos quadros do vídeo utilizando visão computacional.
03	O sistema deve extrair dados cinemáticos dos movimentos, como ângulos articulares, posição dos membros e trajetória.
04	O sistema deve calcular a velocidade do movimento do atleta a partir da variação das posições ao longo do tempo.
05	O sistema deve apresentar graficamente os dados obtidos em tempo de execução, facilitando a análise.
06	O sistema deve identificar automaticamente o momento do arremesso para fins de segmentação do movimento.

Fonte: O Autor.

Tabela 6 – Requisitos não funcionais do sistema computacional.

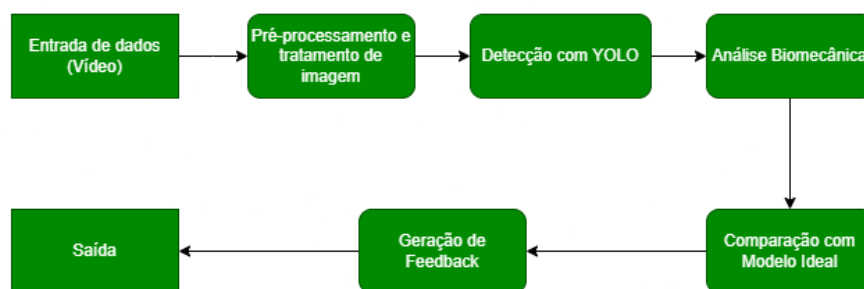
Número Requisito	Requisito Não Funcional
01	O sistema computacional deve ser desenvolvido utilizando a linguagem de programação Python.
02	O sistema deve utilizar bibliotecas como OpenCV, MediaPipe e Matplotlib.
03	O sistema deve ser compatível com o sistema operacionais Windows.
04	A interface do sistema deve ser simples e acessível a usuários sem experiência técnica.
05	O repositório do projeto deve estar no Github.
06	O sistema deve utilizar o YOLOv8 como modelo.

Fonte: O Autor.

4.2 FLUXOGRAMA DA ARQUITETURA DO PROJETO

Na ilustração 14 a seguir, temos um fluxograma que representa a arquitetura do projeto na prática, mostrando os passos que serão executados para que o projeto tenha um resultado satisfatório.

Figura 14 – Fluxograma da arquitetura do projeto.



Fonte: O Autor.

O fluxograma é composto pelas seguintes ações:

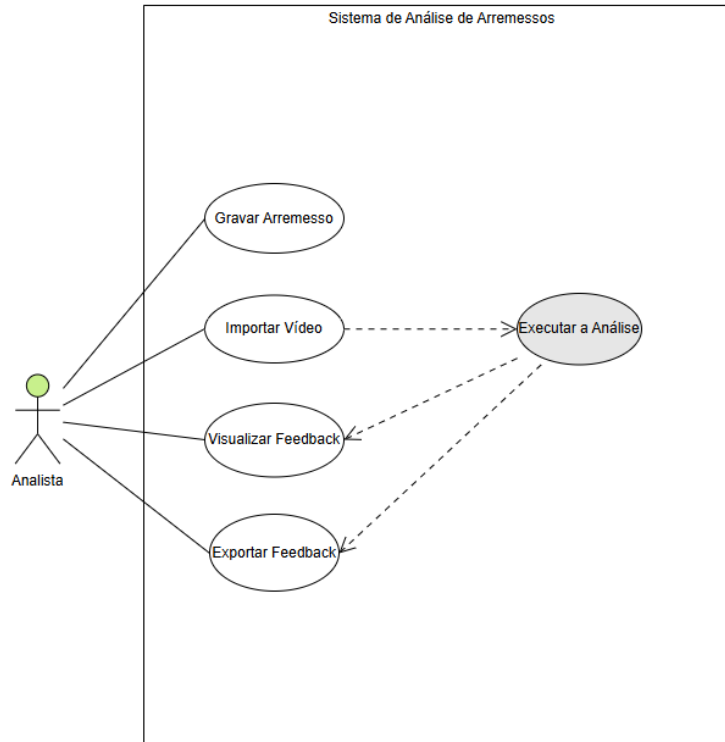
- **Entrada de dados (Vídeo):** O sistema recebe como entrada vídeos contendo arremessos no basquetebol, que serão utilizados como base para análise biomecânica.
- **Pré-processamento e tratamento de imagem:** Os quadros extraídos dos vídeos passam por processos de redimensionamento, filtragem e correção de contraste, preparando as imagens para a detecção com maior precisão.
- **Detecção com YOLO:** Utiliza-se o modelo YOLO para identificar automaticamente partes do corpo do atleta, como cabeça, ombros, cotovelos e mãos, em cada quadro analisado.
- **Análise Biomecânica:** A partir dos pontos detectados, são calculados ângulos articulares e características do movimento que representam a execução do arremesso.
- **Comparação com Modelo Ideal:** Os dados obtidos são comparados com uma referência ideal de arremesso, permitindo identificar falhas na técnica ou na postura do atleta.
- **Geração de Feedback:** Com base na comparação, o sistema fornece orientações visuais e textuais destacando o que pode ser aprimorado na execução do movimento.
- **Saída:** Os resultados são disponibilizados na forma de visualizações gráficas, imagens anotadas ou arquivos exportáveis, auxiliando treinadores e atletas no processo de correção e desempenho.

4.3 DIAGRAMA DE CASO DE USO

O diagrama de caso de uso apresentado na Ilustração 15 a seguir mostra as principais funcionalidades do Sistema de Análise de Arremessos sob a perspectiva do utilizador

principal e as interações que este pode ter para atingir os seus objetivos.

Figura 15 – Diagrama de Caso de Uso.



Fonte: O Autor.

Como demonstrado no diagrama, o O **Analista**, que pode ser o treinador ou o próprio jogador que fez o arremesso, possui as seguintes ações dentro do sistema:

- **Gravar Arremesso:** Embora tecnicamente a gravação de vídeo possa ocorrer fora do sistema (por meio de câmeras ou dispositivos móveis), este caso de uso está presente como uma etapa inicial no fluxo de trabalho do Analista. Ele representa a captura do conteúdo necessário para alimentar o sistema de análise.
- **Importar Vídeo:** Após a gravação, o Analista realiza o upload do vídeo para o sistema. O vídeo deve conter um arremesso completo e estar em um formato compatível com os algoritmos de análise.
- **Executar a Análise:** Com o vídeo já carregado, o Analista pode solicitar que o sistema realize a análise biomecânica. Esse processo envolve múltiplas operações internas, como a detecção de pontos articulares via modelo YOLO, cálculo dos ângulos de movimento e comparação dos dados obtidos com um modelo ideal pré-estabelecido.
- **Visualizar Feedback:** Após a conclusão da análise, o Analista pode acessar os resul-

tados gerados. O feedback inclui melhorias que o atleta pode buscar para melhorar a eficiência de seus arremessos.

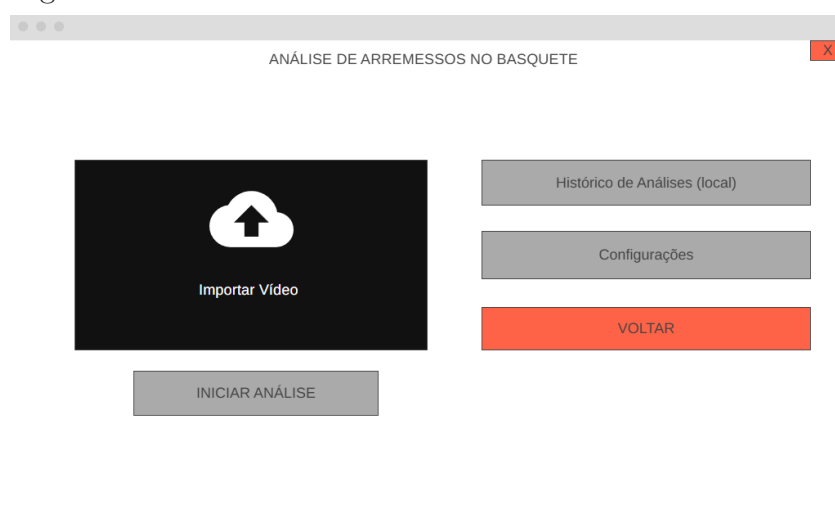
- **Exportar Feedback:** Por fim, o Analista tem a possibilidade de exportar os dados gerados pelo sistema, para assim poder comparar os arremessos ao longo do tempo.

4.4 WIREFRAMES

Os Wireframes a seguir mostram uma projeção do protótipo do sistema de análise.

4.4.1 Wireframe 1

Figura 16 – Wireframe 1.

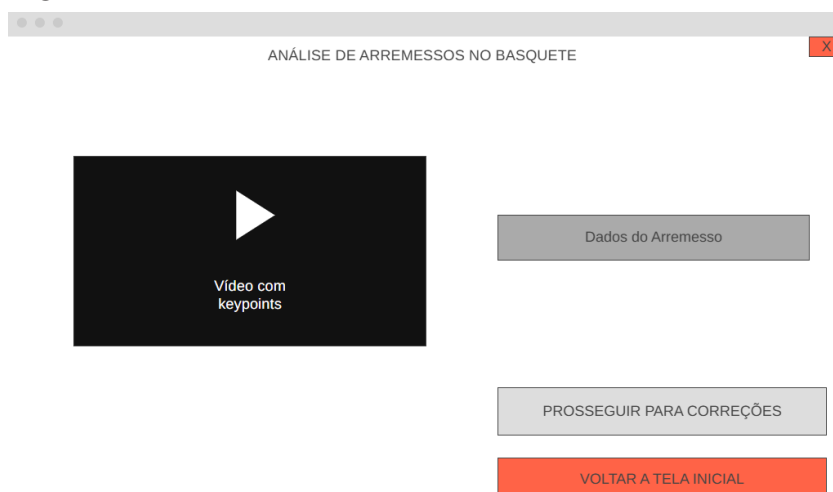


Fonte: O Autor.

O Wireframe mostrado na figura acima, descreve a tela inicial do protótipo, tendo como função principal importar o vídeo do arremesso, após a importação ao clicar no botão de análise, o modelo do YOLO entra em ação, fazendo o processamento do vídeo. Além disso, temos a aba de configurações e também um botão para ver o histórico de análises, utilizando os registros que foram salvos localmente pelo usuário, por fim, um botão de fechar o sistema está disponível.

4.4.2 Wireframe 2

Figura 17 – Wireframe 2.



Fonte: O Autor.

Já o Wireframe 2 mostrado na figura acima, descreve a tela em que o vídeo processado aparece, com as métricas do arremesso aparecendo ao lado, além disso existe a opção de voltar ao menu principal e a opção de avançar para o feedback de melhorias do arremesso.

4.4.3 Wireframe 3

Figura 18 – Wireframe 3.



Fonte: O Autor.

Nesta parte, conseguimos novamente ver o vídeo, porém agora com o feedback de pontos a melhorar ao lado, caso o usuário ache necessário, existe a opção de exportar o vídeo e também o feedback em pdf, um botão de voltar ao menu principal após a conclusão

da análise está disponível, todas as abas tem um botão para fechar o programa também no canto superior direito da tela.

REFERÊNCIAS

- AIP PUBLISHING. **Human Pose Estimation Using BlazePose**. 2023. Disponível em: <https://pubs.aip.org/aip/acp/article/2971/1/040049/3296267/> Human-pose-estimation-using-blaze-pose. Acesso em: 21 maio 2025.
- AMADIO, A. C.; SERRÃO, J. C. A biomecânica em educação física e esporte. **Revista Brasileira de Educação Física e Esporte**, v. 25, n. esp., p. 15-24, 2011. Disponível em: <https://www.scielo.br/j/rbefe/a/6LRgqXLHGhgyrFMsFG5Vyd/>. Acesso em: 19 maio 2025.
- ANALYTICS VIDHYA. **A Comprehensive Guide on Human Pose Estimation**. 2022. Disponível em: <https://www.analyticsvidhya.com/blog/2022/01/a-comprehensive-guide-on-human-pose-estimation/>. Acesso em: 21 maio 2025.
- ANTONELLO, R. **Introdução à Visão Computacional com Python e OpenCV 3**. 2016. Disponível em: <http://www.antonello.com.br>. Acesso em: 21 maio 2025.
- ARORA, S. et al. **Theoretical Physics of Deep Learning**. arXiv:2106.10165, 2021. Disponível em: <https://arxiv.org/abs/2106.10165>
- AWS. **O que é visão computacional?** 2024. Disponível em: <https://aws.amazon.com/pt/what-is/computer-vision/>. Acesso em: 19 maio 2025.
- BIOMECHANICS of the Basketball Jump Shot**. In: PHYSIO-PEDIA. [S. l.], [s. d.]. Disponível em: https://www.physio-pedia.com/Biomechanics_of_the_Basketball_Jump_Shot. Acesso em: 11 jun. 2025.
- BOESCH, G. **Human Pose Estimation - Everything You Need to Know**. Viso.ai, 2023. Disponível em: <https://viso.ai/deep-learning/pose-estimation-ultimate-overview/>. Acesso em: 21 maio 2025.
- BRANDINA, K. **Biomecânica aplicada ao esporte**. São Paulo: Editora Sol, 2018.
- CABRAL, Adriel dos Santos Araújo. **Sistema de Visão Computacional para Análise de Jogadas em Goalball**. 2024. Trabalho de Conclusão de Curso (Bacharelado em Ciência da Computação) - Universidade Federal da Paraíba, João Pessoa, 2024. Disponível em: <https://repositorio.ufpb.br/jspui/handle/123456789/32677>. Acesso em: 02 abr. 2025.
- CANAN, F.; MENDES, J. C.; SILVA, R. V. Análise estatística no basquetebol de base: perfil do Campeonato Paranaense de Basquetebol masculino Sub-17. **Revista Brasileira de Educação Física e Esporte**, v. 29, n. 2, p. 289-302, 2015. Disponível em: https://www.researchgate.net/publication/307763304_Analise_

estatistica_no_basquetebol_de_base_perfil_do_Campeonato_Paranaense_de_Basquetebol_masculino_Sub-17. Acesso em: 17 maio 2025.

CARVALHO, A. S. A. de et al. Análise comparativa entre os principais algoritmos de detecção facial: Haar Cascade, HOG, CNN, YOLO e DeepFace. **Open Science Research V**, 2022. Disponível em: <https://doi.org/10.37885/220709383>. Acesso em: 21 maio 2025.

CASEIRO, A. **Análise cinemática do lançamento em suspensão do basquetebol**. Dissertação (Mestrado em Ciências do Desporto) - Universidade de Coimbra, 2015. Disponível em: <https://estudogeral.uc.pt/handle/10316/96639>. Acesso em: 19 maio 2025.

CASEIRO, A.; COSTA, M. J.; OLIVEIRA, D.; VAZ, J. R.; CASTRO, M. A. Biomechanical parameters of the basketball jump shot: comparison between distances and experience level. **Acta of Bioengineering and Biomechanics**, v. 25, n. 1, p. 25-33, 2023. DOI: 10.37190/abb-02205-2023-01.

CBB - CONFEDERAÇÃO BRASILEIRA DE BASKETBALL. **Regras Oficiais de Basketball – FIBA 2020 (versão em português)**. São Paulo: CBB, 2020. Disponível em: <https://www.cbb.com.br/wp-content/uploads/Regras-Oficiais-de-Basketball-FIBA-2020-Traduzida-para-Portugues.pdf>. Acesso em: 15 maio 2025.

COB - COMITÊ OLÍMPICO DO BRASIL. **Basquete**. Disponível em: <https://www.cob.org.br/time-brasil/esportes/1-basquete>. Acesso em: 15 maio 2025.

CODERADE.IO. **Pose Estimation in Computer Vision: Everything You Need to Know**. 2024. Disponível em: <https://www.codetrade.io/blog/pose-estimation-in-computer-vision-everything-you-need-to-know/>. Acesso em: 21 maio 2025.

COLETTA, Luiz F. S.; AMARAL, A. C. S. **Segmentação de Pólipos em Imagens de Colonoscopia utilizando YOLOv8**. 2024. Trabalho de Conclusão de Curso (Graduação em Engenharia de Computação) - Universidade Tecnológica Federal do Paraná, Cornélio Procópio, 2024. Figura 2. Disponível em: https://www.researchgate.net/figure/Figura-2-Arquitetura-simplificada-do-YOLOv8-Jocher-et-al-2023_fig1_381750528. Acesso em: 12 jun. 2025.

DATA HACKERS. **Matplotlib e Storytelling com Dados - Parte I**. Medium, 2019. Disponível em: <https://medium.com/data-hackers/matplotlib-e-storytelling-com-dados-pt-i-48c289943d60>. Acesso em: 21 maio 2025.

DE ROSE JÚNIOR, D.; GASPAR, M. F.; ASSUMPÇÃO, M. V. Construção e validação preliminar de instrumento de avaliação do desempenho técnico-tático individual no basquetebol. **Revista de Educação Física/UEM**, v. 16, n. 2, p. 201-208, 2005.

Disponível em: <https://www.scielo.br/j/refuem/a/FNrcSGnjgGgQ5bHNXMfSqmM/>. Acesso em: 16 maio 2025.

ENCORD. **YOLO Object Detection Explained: A Beginner's Guide**. 2024. Disponível em: <https://encord.com/blog/yolo-object-detection-guide/>. Acesso em: 21 maio 2025.

FONSECA, João José Saraiva da. **Metodologia da pesquisa científica**. Universidade Estadual do Ceará, 2002.

GERHARDT, Tatiana Engel; SILVEIRA, Denise Tolfo. **Métodos de pesquisa**. Porto Alegre: Editora da UFRGS, 2009.

GETGURU. **Computer Vision**. 2024. Disponível em: <https://www.getguru.com/pt/reference/computer-vision>. Acesso em: 19 maio 2025.

GIL, Antonio Carlos. **Métodos e técnicas de pesquisa social**. 6. ed. São Paulo: Atlas, 2008.

GOODFELLOW, I.; BENGIO, Y.; COURVILLE, A. **Deep Learning**. MIT Press, 2016. Disponível em: <https://www.deeplearningbook.org/>

HEMOCOURT. **The future of basketball training is here**. HomeCourt AI, 2025. Disponível em: <https://www.homecourt.ai>. Acesso em: 12 jun. 2025.

IMPULSIONA. **Fundamentos do basquete: conheça os 3 principais**. Disponível em: <https://impulsiona.org.br/3-fundamentos-do-basquete/>. Acesso em: 15 maio 2025.

INGRAM MICRO. **O que é visão computacional e como funciona?** 2024. Disponível em: <https://blog.ingrammicro.com.br/inovacao-e-tendencias/visao-computacional>. Acesso em: 19 maio 2025.

JOCHER, Glenn; QIU, Jing. **Ultralytics YOLOv11**. Versão 11.0.0. [S. l.]: Ultralytics, 2024. Disponível em: <https://github.com/ultralytics/ultralytics>. Acesso em: 12 jun. 2025.

KARPATHY, A.; LI, F. F. **CS231n: Convolutional Neural Networks for Visual Recognition**. Stanford University. Disponível em: <https://cs231n.github.io/convolutional-networks/>

Knudson, D. V. (2007). **Fundamentals of biomechanics**. Springer Science and Business Media. Link: <https://link.springer.com/book/10.1007/978-1-4757-2518-2>

LAMAS, L.; MORALES, J. C. P. Integração entre a análise do desempenho e o ensino-aprendizagem nos esportes coletivos. **Revista Brasileira de Ciências do**

Esporte, v. 44, 2022. Disponível em:

<https://www.scielo.br/j/rbce/a/qxXX8bn463Y8PPMLbQtXFVG/>. Acesso em: 16 maio 2025.

LEES, A. Technique analysis in sports: a critical review. **Journal of Sports Sciences**, v. 20, p. 813-828, 2002.

LIN, T. et al. **Towards an Understanding of Situated AR Visualization for Basketball Free-Throw Training**. arXiv preprint arXiv:2104.04118, 2021. Disponível em: <https://arxiv.org/abs/2104.04118>. Acesso em: 19 maio 2025.

NASCIMENTO, Jorge Mateus. **Geração de informações estratégicas para clubes de futebol utilizando visão computacional**. 2023. Trabalho de Conclusão de Curso (Graduação em Análise e Desenvolvimento de Sistemas) - Instituto Federal de Educação, Ciência e Tecnologia Baiano, Teixeira de Freitas, 2023. Disponível em: <https://saberaberto.uneb.br/items/f9bd1d98-a62d-4b76-9ba4-bebd26f256b7>. Acesso em: 16 abril 2025.

NOAH BASKETBALL. **Data-Driven Development**. Noah Basketball, 2025. Disponível em: <https://noahbasketball.com>. Acesso em: 12 jun. 2025.

NVIDIA. **Image Processing vs. Computer Vision: What's the Difference?** 2023. Disponível em: <https://resources.nvidia.com/en-us/edge-computing/image-processing-vs-computer-vision>. Acesso em: 21 maio 2025.

OKAZAKI, V. H. A. et al. A review on basketball jump shot. **Journal of Sports Science and Medicine**, v. 14, n. 3, p. 638-643, 2015. Disponível em: https://www.researchgate.net/publication/279180866_A_review_on_basketball_jump_shot. Acesso em: 19 maio 2025.

OPENCV. **OpenCV-Python Tutorials**. [S. l.]: OpenCV, 2025. Disponível em: https://docs.opencv.org/4.x/d6/d00/tutorial_py_root.html. Acesso em: 12 jun. 2025.

PAIVA, F. A. P. et al. **Introdução a Python com Aplicações de Sistemas Operacionais**. Natal: Editora IFRN, 2019. ISBN 978-65-86293-38-8.

PAN, W. J. et al. Biomechanical Analysis of Shooting Performance for Basketball Players of Different Genders Based on Computer Vision. **Journal of Physics: Conference Series**, v. 2024, n. 1, p. 012016, 2021. DOI: 10.1088/1742-6596/2024/1/012016. Disponível em: <https://iopscience.iop.org/article/10.1088/1742-6596/2024/1/012016>. Acesso em: 16 abril 2025.

PAPERSWITHCODE. **Pose Estimation**. 2024. Disponível em: <https://paperswithcode.com/task/pose-estimation>. Acesso em: 21 maio 2025.

Pressman, R. S. (2016). **Engenharia de software: uma abordagem profissional** (7. ed.). AMGH Editora.

REDMON, J. et al. **You Only Look Once: Unified, Real-Time Object Detection**. arXiv:1506.02640, 2016. Disponível em: <https://arxiv.org/abs/1506.02640>. Acesso em: 21 maio 2025.

SOMMERVILLE, Ian. **Engenharia de Software**. 9ª ed. São Paulo: Pearson Prentice Hall, 2011. Capítulo 8 - Teste de Software.

SOUZA, Werbert de Matos. **O esporte mediado pelas tecnologias digitais: análise de estudos brasileiros**. 2023. 32 f. Trabalho de Conclusão de Curso (Bacharelado em Educação Física) - Universidade Federal de Pernambuco, Centro Acadêmico de Vitória, Vitória de Santo Antão-PE, 2023.

STRUZIK, A.; PIETRASZEWSKI, B.; ZAWADZKI, J. Biomechanical analysis of the jump shot in basketball. **Journal of Human Kinetics**, v. 42, p. 73-79, 2014. DOI: 10.2478/hukin-2014-0062.

TEFERI, Getu; ENDALEW, Dessalew. Methods of Biomechanical Performance Analyses in Sport: Systematic Review. **American Journal of Sports Science and Medicine**, v. 8, n. 2, p. 47-52, 2020. DOI: 10.12691/ajssm-8-2-2.

TEIXEIRA, Jayor. **Classificação de Golpes de Karatê utilizando Redes Neurais e Visão Computacional**. 2024. Trabalho de Conclusão de Curso (Graduação em Sistemas de Informação) – Universidade Federal de Santa Catarina, Araranguá, 2024. Disponível em: <https://repositorio.ufsc.br/handle/123456789/262114>. Acesso em: 16 abril 2025.

ULTRALYTICS. **A aplicação da visão computacional no esporte**. 2024. Disponível em: <https://www.ultralytics.com/pt/blog/exploring-the-applications-of-computer-vision-in-sports>. Acesso em: 19 maio 2025.

ULTRALYTICS. Models. In: ULTRALYTICS. **Ultralytics YOLOv8 Documentation**. [S. l.], 2025. Disponível em: <https://docs.ultralytics.com/pt/models/>. Acesso em: 12 jun. 2025.

UNIVERSIDADE FEDERAL DE SANTA MARIA. **Métodos de Medição: Cinemetria**. Santa Maria: UFSM, 2019. Disponível em: https://www.ufsm.br/app/uploads/sites/644/2019/07/metodos_de_medicao_cinemetria.pdf. Acesso em: 12 abr. 2025.

YASEEN, Muhammad. **What is YOLOv8: An In-Depth Exploration of the Internal Features of the Next-Generation Object Detector**. arXiv preprint arXiv:2408.15857, 2024. Disponível em: <https://arxiv.org/abs/2408.15857>. Acesso em: 14 abr. 2025.