Unsupervised Specialization of Visual Subclasses Using K-Means in YOLO-Based Detection Pipelines

Pedro Henrique Campos Moreira [1], Bianca Panacho Ferreira [2], Marcus Vinicius Diniz dos Reis [3]

[1]Instituto de Ciências Exatas e Técnologicas
Universidade Federal de Viçosa, Campus Rio Paranaíba (UFV-CRP)

pedro.henrique.moreira@ufv.br [1], bianca.p.ferreira@ufv.br [2], marcus.v.reis@ufv.br [3]

Abstract. *This work presents a pipeline for enriching object detection datasets through automatic visual subclass labeling. Using generic annotations, the methodology uses a YOLO detector to extract object instances and then applies image processing to extract color vectors. The unsupervised K-Means algorithm is used to cluster these vectors, autonomously defining new subclasses. The YOLOv8n model, retrained with the refined dataset, achieved 94.68% accuracy in distinguishing sports teams, validating the approach as an effective solution for overcoming the need for manual annotation.*

Resumo. *Este trabalho apresenta um pipeline para enriquecer datasets de detecção de objetos através da rotulagem automática de subclasses visuais. A partir de anotações genéricas, a metodologia utiliza um detector YOLO para extrair instâncias de objetos e, em seguida, aplica processamento de imagem para extrair vetores de cor. O algoritmo não supervisionado K-Means é usado para agrupar estes vetores, definindo novas subclasses de forma autônoma. O modelo YOLOv8n, re-treinado com o dataset refinado, alcançou 94.68% de precisão na tarefa de distinguir equipes esportivas, validando a abordagem como uma solução eficaz para contornar a necessidade de anotação manual.*

1. Introdução

A deteção de objetos, impulsionada por modelos de aprendizagem profunda (*deep learning*), tornou-se uma tecnologia central em inúmeras aplicações, desde a condução autônoma até ao diagnóstico médico. Dentro deste ecossistema, a família de modelos YOLO (*You Only Look Once*) [Redmon and Farhadi 2017] destacou-se pela sua arquitetura inovadora, que reformulou a deteção, permitindo um processamento em tempo real sem sacrificar significativamente a precisão. Esta combinação de velocidade e eficácia tornou o YOLO uma escolha padrão para a primeira etapa de muitos sistemas de análise visual, fornecendo a capacidade fundamental de localizar e categorizar objetos de interesse em imagens e vídeos com um desempenho robusto.

A sua aplicabilidade estende-se por uma vasta gama de domínios, desempenhando um papel transformador em cada um deles. Em sistemas de condução autônoma, a deteção de objetos é a tecnologia que permite a um veículo ”ver”e reagir a peões, outros carros e sinais de trânsito. Na área da saúde, auxilia radiologistas na identificação de anomalias em imagens médicas, como tumores ou lesões, com uma precisão que pode superar a do olho humano. No retalho, otimiza a gestão de inventário através da contagem automática de produtos em prateleiras, enquanto em sistemas de segurança, permite

a monitorização e o alerta em tempo real para atividades suspeitas. Esta ubiquidade demonstra que a deteção de objetos não é apenas uma ferramenta acadêmica, mas uma tecnologia fundamental que sustenta muitas das inovações.

No entanto, o sucesso dos modelos de deteção como o YOLO depende intrinsecamente da disponibilidade de grandes conjuntos de dados, meticulosamente anotados com rótulos de classe precisos. Este processo de rotulagem manual é um conhecido gargalo no desenvolvimento de sistemas de inteligência artificial, sendo moroso e dispendioso.

Paralelamente ao avanço dos modelos supervisionados, técnicas clássicas de Machine Learning não supervisionado continuam a ser ferramentas indispensáveis para a análise de dados. Entre elas, o algoritmo K-Means destaca-se pela sua simplicidade e eficiência na tarefa de clustering. Esta capacidade de encontrar uma estrutura inerente nos dados sem qualquer tipo de anotação prévia torna-o uma ferramenta poderosa para a descoberta de padrões, especialmente quando confrontado com um grande volume de dados onde a categorização manual seria impraticável.

Para contornar as limitações de rotulagem de modelode de detecção, este trabalho propõe uma abordagem que se afasta da necessidade de anotação manual, explorando o potencial de algoritmos clássicos de Machine Learning para refinar e enriquecer anotações existentes. Neste projeto foi feito um *pipeline* em duas fases, concebido para a descoberta e rotulagem automática de sub-classes visuais. Na primeira fase, um detetor de objetos pré-treinado é utilizado para isolar as instâncias com os rótulos genéricos já disponíveis. Na segunda fase, em vez de recorrer a redes neurais complexas para a re-classificação, empregamos uma combinação de técnicas de pré-processamento de imagem e o algoritmo de clustering não supervisionado K-Means.

2. Referencial Teórico

Este trabalho se fundamenta na interseção de três domínios da ciência da computação: a detecção de objetos baseada em aprendizagem profunda, os algoritmos de aprendizagem não supervisionada para agrupamento de dados e as técnicas de processamento digital de imagens para extração de características. A seguir, detalhamos os conceitos teóricos de cada um desses pilares que sustentam a metodologia proposta.

2.1. Detecção de Objetos com Redes Neurais Convolucionais

A detecção de objetos é uma tarefa central em visão computacional que visa identificar e localizar instâncias de objetos pertencentes a determinadas classes em uma imagem ou vídeo. O advento das Redes Neurais Convolucionais (CNNs) revolucionou esta área, superando significativamente os métodos clássicos [LeCun et al. 2002]. As CNNs são arquiteturas de *deep learning* projetadas para processar dados com uma topologia de grade, como as imagens, através da aplicação de filtros (ou kernels) que aprendem a extrair características de forma hierárquica, desde bordas e texturas simples em camadas iniciais até conceitos complexos e abstratos em camadas mais profundas.

Dentro deste contexto, a família de modelos YOLO (*You Only Look Once*) [Redmon and Farhadi 2017] representa um marco. Diferentemente de abordagens anteriores de duas etapas (que primeiro propunham regiões e depois as classificavam), o YOLO trata a detecção de objetos como um único problema de regressão. A imagem de entrada é dividida em uma grade (grid), e para cada célula dessa grade, o modelo prevê

simultaneamente as coordenadas das *bounding boxes*, um índice de confiança (*confidence score*) para a presença de um objeto e as probabilidades de classe. Este paradigma de *single-shot* permite um processamento extremamente rápido, tornando o YOLO e suas variantes, como o YOLOv8n utilizado neste trabalho, ideais para aplicações em tempo real e como primeira etapa em pipelines de análise de vídeo mais complexos.

2.2. Aprendizagem Não Supervisionada e o Algoritmo K-Means

Em contraste com a aprendizagem supervisionada, que depende de datasets extensivamente rotulados, a aprendizagem não supervisionada explora a estrutura intrínseca dos dados na ausência de rótulos prévios. O seu objetivo é encontrar padrões, grupos ou anomalias de forma autônoma. A técnica mais proeminente dentro deste paradigma é o agrupamento (*clustering*).

O algoritmo K-Means é um dos métodos de agrupamento mais populares e amplamente utilizados devido à sua simplicidade e eficiência computacional [MacQueen 1967]. O seu objetivo é particionar um conjunto de n observações em k clusters, nos quais cada observação pertence ao cluster com a média (centróide) mais próxima. O algoritmo opera de forma iterativa:

1. **Inicialização:** k centróides são inicializados aleatoriamente no espaço de características.
2. **Atribuição:** Cada ponto de dado é atribuído ao cluster cujo centróide é o mais próximo, geralmente com base na distância euclidiana.
3. **Atualização:** Os centróides de cada cluster são recalculados como a média de todos os pontos de dado atribuídos a ele.

Os passos 2 e 3 são repetidos até que a posição dos centróides convirja, ou seja, até que as atribuições dos pontos aos clusters não mudem mais. A sua eficácia em agrupar dados com base em vetores de características torna-o ideal para a tarefa de diferenciar subclasses visuais, como times em um esporte, a partir de atributos como a cor.

2.3. Extração de Características Visuais por Processamento de Imagem

A ponte entre a detecção de objetos e o agrupamento não supervisionado é construída por meio de técnicas de processamento digital de imagens, que permitem a extração de uma representação vetorial significativa de cada objeto detectado. Para que o K-Means possa operar, a informação visual de uma região da imagem (a *bounding box*) precisa ser convertida em um vetor numérico.

A cor é uma das características mais salientes para a diferenciação de objetos. As cores em uma imagem digital são comumente representadas em espaços de cor, como o RGB (Red, Green, Blue). A extração da cor média de uma região de interesse (ROI) é uma forma eficaz de engenharia de características (*feature engineering*), que resume a informação cromática de milhares de pixels em um único vetor tridimensional.

Para aumentar a robustez dessa extração, técnicas de pré-processamento como a segmentação de imagem são aplicadas. Ao isolar o objeto de interesse do fundo, garante-se que apenas os pixels pertencentes ao objeto contribuam para o cálculo do vetor de características, reduzindo o ruído e aumentando a capacidade de discriminação do algoritmo de agrupamento.

3. Trabalhos Relacionados

A crescente demanda por sistemas inteligentes capazes de operar com mínima intervenção humana tem impulsionado a busca por estratégias que permitam a especialização automática de classes genéricas. No campo da visão computacional, isso se reflete na identificação de subclasses dentro de categorias previamente rotuladas, tarefa particularmente desafiadora quando não se dispõe de dados anotados com granularidade suficiente. Nesse contexto, a combinação entre detecção de objetos baseada em redes neurais convolucionais e técnicas de agrupamento não supervisionado, como o algoritmo K-Means, tem se mostrado promissora. Essa abordagem híbrida tem sido explorada como alternativa eficiente para a geração de pseudo-rótulos, possibilitando a descoberta de estruturas latentes nos dados e a redução do custo associado à rotulagem manual em pipelines de visão computacional.

Qiu et al. propuseram o CLDA-YOLO, um modelo de detecção baseado em YOLO com aprendizagem contrastiva não supervisionada, que utiliza pseudo-rótulos gerados dinamicamente para adaptação entre domínios [Qiu et al. 2024]. A arquitetura demonstrou alta eficácia em ambientes com variações visuais complexas, sem exigir anotações manuais adicionais, sendo particularmente útil para subdividir categorias genéricas em subconjuntos visuais coerentes.

No domínio esportivo, Koshkina et al. exploraram a classificação não supervisionada de jogadores em vídeos utilizando aprendizado contrastivo. Apesar de não utilizarem YOLO diretamente, os autores demonstraram que é possível atingir acurácia superior a 94% na identificação de equipes baseando-se apenas em padrões visuais extraídos de frames não anotados, o que valida a eficácia do uso de atributos visuais para a formação de subclasses sem a necessidade de labels explícitos [Koshkina et al. 2021].

Grishin et al. desenvolveram o YOLO-CL, uma modificação do YOLO otimizada para detecção de aglomerados de galáxias [Grishin et al. 2023]. Embora voltado para astronomia, o modelo demonstrou que a detecção baseada em padrões visuais (como densidade e cor) pode ser extremamente eficaz, mesmo quando as instâncias pertencem a uma mesma classe genérica, reforçando a ideia de que subagrupamentos visuais podem ser extraídos sem rótulos explícitos.

Complementarmente, Kowsari e Alassaf propuseram um algoritmo de detecção baseado em clustering ponderado aplicado sobre dados RGB-D [Kowsari and Alassaf 2016]. Após extrair as posições e cores de objetos detectados em tempo real, aplicaram um algoritmo de clustering que segmentava automaticamente instâncias distintas com base em atributos como cor e localização espacial, reforçando a viabilidade do uso de características visuais como base para rotulagem automática.

Diante dos trabalhos analisados, observa-se uma convergência metodológica em torno do uso de atributos visuais simples, como cor, forma ou localização, para a segmentação de subclasses a partir de anotações genéricas. A presente proposta se insere nesse cenário ao implementar um pipeline modular em duas fases, que alia a robustez da detecção com YOLOv8n à simplicidade e eficiência do agrupamento por K-Means, utilizando vetores cromáticos extraídos via pré-processamento visual. Ao evitar o uso de redes neurais adicionais, embeddings semânticos ou OCR, a abordagem destaca-se por sua leveza computacional e aplicabilidade em contextos com recursos limitados ou ausência

de rótulos refinados. Essa arquitetura flexível permite não apenas a especialização automática de classes genéricas, como também o reaproveitamento dos dados em ciclos supervisionados subsequentes, estendendo seu potencial a diferentes domínios de aplicação.

4. Método Proposto

Para superar o desafio da anotação manual de subclasses, propomos um pipeline que enriquece *datasets* existentes com rótulos específicos para sub-classes. A nossa abordagem, ilustrada na Figura 1, é executada em quatro fases sequenciais: (1) preparação e validação dos dados de entrada; (2) pré-processamento e extração de características visuais; (3) agrupamento e rotulagem automática com K-Means; e (4) geração do novo dataset para o re-treinamento do detector de objetos.

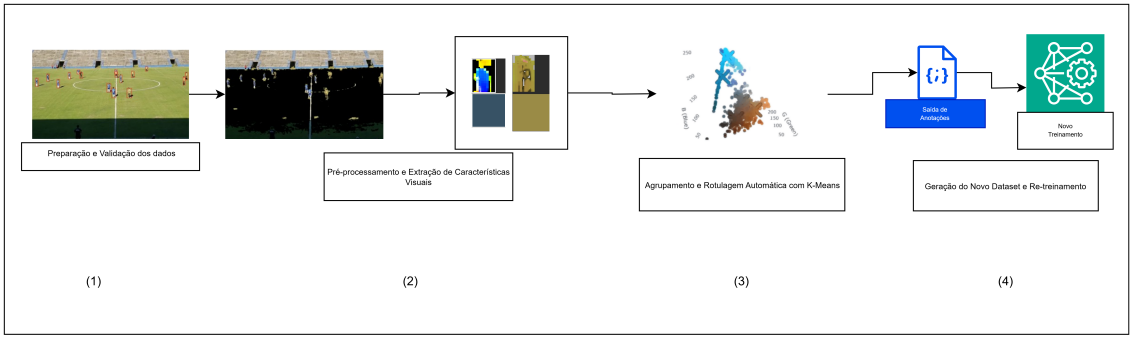

Figura 1. Diagrama Geral da Metodologia

4.1. Preparação e Validação dos Dados

A etapa inicial do pipeline consiste na ingestão dos dados brutos: os vídeos do evento esportivo e os correspondentes arquivos de anotação em formato JSON. Estes arquivos contêm as coordenadas das *bounding boxes* para cada instância da classe genérica (e.g., ”pessoa”).

Para assegurar a integridade das anotações, um processo de validação visual é executado. Frames são extraídos dos vídeos e as *bounding boxes* são sobrepostas, permitindo a verificação da correspondência entre as coordenadas e os objetos em cena. Esta etapa é fundamental para garantir a qualidade dos dados que alimentarão as fases subsequentes do processo.

4.2. Pré-processamento e Extração de Características Visuais

Uma vez validadas as anotações, o pipeline foca-se no isolamento dos objetos de interesse para extrair características visuais discriminativas. Para cada *bounding box*:

1. **Recorte e Segmentação:** A região do objeto é recortada do frame original. Em seguida, uma máscara de segmentação é aplicada ao fundo (*background*) para neutralizar informações visuais irrelevantes e isolar completamente o objeto de interesse.
2. **Realce Visual:** A imagem resultante do objeto isolado passa por ajustes de brilho, saturação e zoom. O objetivo desta etapa é realçar as características cromáticas e texturais dos uniformes e, ou, outros acessórios, que servirão como base para a diferenciação das subclasses.

3. **Extração de Cor:** As informações de cor da região de interesse são extraídas e armazenadas. A cor média da região é calculada e utilizada como vetor de características (*feature vector*) para a fase de agrupamento.

4.3. Agrupamento e Rotulagem Automática com K-Means

A classificação das subclasses é realizada por meio do algoritmo de agrupamento não supervisionado K-Means. Para a definição dos clusters, optou-se por um valor de k deliberadamente superior ao número de subclasses esperadas (e.g., time A, time B, juiz). Esta abordagem, conhecida como *over-clustering*, permite uma separação mais fina das nuances de cor, isolando eficazmente não só as classes principais, mas também variações de iluminação e instâncias ruidosas.

Após a execução do K-Means, foi realizada uma análise visual dos centróides de cada cluster. Clusters com cores semanticamente semelhantes (e.g., diferentes tons de azul do uniforme do Time A) foram então agrupados para formar a subclasse final. Este passo intermédio garante uma rotulagem mais robusta e precisa. Ao final do processo, os arquivos de anotação JSON são programaticamente atualizados: o rótulo genérico é substituído pelo identificador da nova subclasse.

4.4. Geração do Novo Dataset e Re-treinamento

Com os arquivos JSON agora enriquecidos com os rótulos das subclasses, a fase final do pipeline prepara os dados para o treinamento de um modelo de deteção especializado. Os arquivos JSON são convertidos para o formato `.txt`, seguindo o padrão exigido pela arquitetura YOLO.

Este processo culmina na criação de um novo conjunto de dados, agora com rótulos de classes específicas e refinadas. Este *dataset* é, então, utilizado para treinar um novo modelo de deteção, como o YOLOv8n. O resultado é um detector capaz não apenas de localizar os objetos, mas também de classificá-los diretamente em suas respectivas subclasses visuais, concluindo o ciclo de especialização de classes sem a necessidade de intervenção manual no processo de rotulagem.

4.5. Avaliação das Métricas

Para uma análise quantitativa e robusta da performance do classificador, utilizaremos um conjunto de métricas de avaliação padrão. Cada métrica oferece uma perspectiva única sobre a capacidade do modelo, permitindo-nos medir desde sua taxa de acerto geral até sua eficiência em cenários específicos de classificação.

- Precisão: Foca na qualidade das classificações positivas. Ela mede, dentre todas as previsões positivas feitas pelo modelo, quantas foram de fato corretas. É uma métrica fundamental quando o custo de um falso positivo é elevado. (Equação 1).

$$Precision = \frac{TP}{TP + FP} \tag{1}$$

- Recall: Mede a capacidade do modelo de identificar todas as instâncias que são realmente positivas. O seu foco é garantir que o mínimo possível de falsos negativos ocorra. (Equação 2).

$$Recall = \frac{TP}{TP + FN} \tag{2}$$

5. Resultados

A eficácia da metodologia proposta foi avaliada através de uma análise quantitativa e qualitativa. O desempenho do modelo YOLOv8n, re-treinado com o nosso dataset enriquecido, foi medido em um conjunto de teste dedicado, extraído de vídeos não utilizados durante o treino.

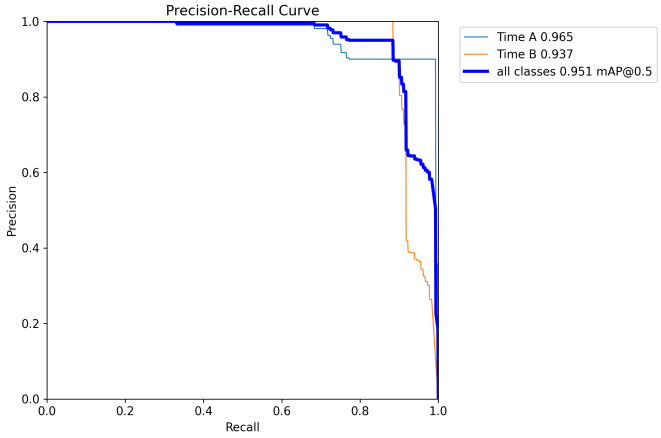

Figura 2. Gráfico de Precision-Recall

Os resultados quantitativos demonstram um desempenho robusto do modelo. Alcançou-se uma precisão (*precision*) de 94.68% e um recall de 93.16%. A curva de precisão-recall, apresentada na Figura 2, ilustra o excelente balanço que o modelo atinge entre estas duas métricas. Adicionalmente, os gráficos de convergência das funções de perda, compilados na Figura 3, confirmam a estabilidade do processo de treino.

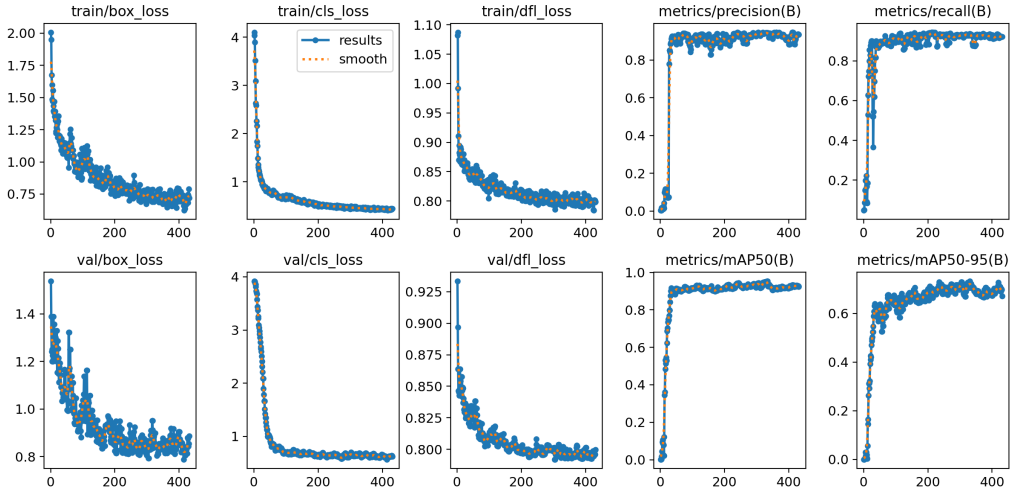

Figura 3. Gráficos de métricas e perdas gerados pelo YOLOv8 durante o treino e validação.

A observação qualitativa, apresentada na Figura 4, complementa os dados numéricos. As imagens ilustram o desempenho do modelo em cenários reais, evidenciando a correta diferenciação entre as subclasses “Time A” e “Time B” com alta confiabilidade, mesmo em condições visuais desafiadoras, como sobreposição de jogadores e

variações de iluminação. Estes resultados visuais validam a capacidade do modelo em generalizar o conhecimento adquirido para além dos dados de treino, reforçando o potencial da nossa abordagem para especializar classes genéricas de forma automática e eficaz.

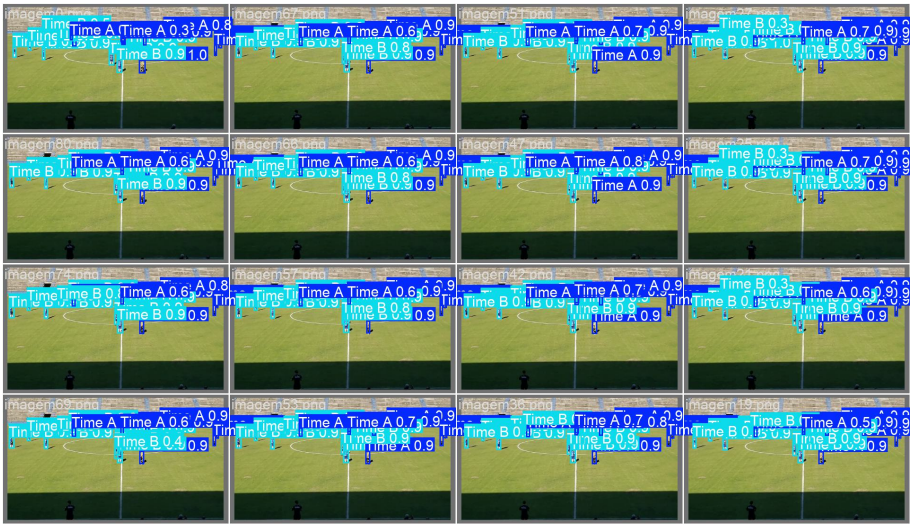

Figura 4. Predições do modelo YOLOv8 re-treinado com as subclasses visuais geradas automaticamente (“Time A” e “Time B”).

6. Conclusão

Este trabalho demonstrou que técnicas clássicas de Machine Learning, como o algoritmo K-Means, ainda desempenham um papel estratégico no ecossistema de visão computacional moderna. Em um cenário amplamente dominado por redes neurais profundas e modelos supervisionados, mostramos que abordagens não supervisionadas continuam sendo ferramentas valiosas, especialmente quando associadas a estratégias inteligentes de pré-processamento e integração com pipelines de anotação e re-treinamento supervisionado.

A proposta apresentada foi capaz de refinar anotações genéricas e gerar subclasses visuais coerentes sem qualquer intervenção humana no processo de rotulagem. O pipeline modular, desde a segmentação visual até a reanotação automática em formato YOLO, demonstrou versatilidade e aplicabilidade prática. Os resultados obtidos, tanto em métricas objetivas quanto nas predições visuais, evidenciaram que os clusters formados foram suficientes para treinar um detector robusto e preciso.

Esta pesquisa reforça que, mesmo diante da sofisticação crescente dos modelos de Deep Learning, técnicas de agrupamento como o K-Means ainda têm espaço relevante, sobretudo em contextos com limitação de recursos ou escassez de rótulos especializados. Como perspectivas futuras, propomos a avaliação de outros métodos de clusterização e a extração de descritores visuais mais elaborados, além da aplicação do pipeline em domínios como varejo, biometria ou diagnóstico por imagem.

Por fim, nossa abordagem destaca um caminho promissor para especializar classes genéricas sem o custo da rotulagem manual, promovendo o uso eficiente e acessível da inteligência artificial em cenários do mundo real.

Referências

Grishin, K., Chupin, S., Vasylenko, A., Barkhatova, T., and Burenin, R. (2023). Yolo-cl: Galaxy cluster detection in the sdss with deep machine learning. *Astronomy & Astrophysics*, 677:A101.

Koshkina, A., Nauata, N., Tighe, J., and Felsen, P. (2021). Unsupervised classification of players in team sports. *arXiv preprint arXiv:2104.10068*.

Kowsari, K. and Alassaf, M. H. (2016). Weighted unsupervised learning for 3d object detection. *arXiv preprint arXiv:1602.05920*.

LeCun, Y., Bottou, L., Bengio, Y., and Haffner, P. (2002). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324.

MacQueen, J. (1967). Multivariate observations. In *Proceedings ofthe 5th Berkeley Symposium on Mathematical Statisticsand Probability*, volume 1, pages 281–297.

Qiu, T., Yang, L., Zhang, L., Tang, J., and Xu, X. (2024). Clda-yolo: Visual contrastive learning based domain adaptive yolo detector. *arXiv preprint arXiv:2412.11812*.

Redmon, J. and Farhadi, A. (2017). Yolo9000: better, faster, stronger. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7263–7271.