

Defect detection in the textile industry using the YOLOv8 model

Cláudio Alberto Waiandt *, Hilário Seibel Júnior

* Programa de Pós-graduação em Computação Aplicada (PPComp)

Instituto Federal do Espírito Santo (IFES) Campus Serra

waiandt26@yahoo.com.br *, hsjunior@ifes.edu.br

Resumo—The textile industry plays a fundamental role in the global economy, being responsible for the production of a wide variety of products. Defect detection in this context is a critical aspect as it presents a series of defects that include objects of reduced dimensions and inspection is often carried out manually. This article improves quality control processes in the textile industry by applying the YOLOv8 model to the public AITEX dataset for automated defect detection, representing a technological advance when compared to the YOLOv5 model, both developed by ultralytics. Furthermore, issues such as the architecture and variants of the YOLOv8 model, limitations, parameter optimization and training strategies, as well as model evaluation are discussed. Promising results are presented, reaching an mAP of 85.11%, indicating that YOLOv8 can effectively be used in the context of textile defect detection.

Index Terms—YOLOv8. AITEX. Object detection. Textile defects. Hyper parameters. Textile industry.

I. INTRODUÇÃO

A indústria têxtil contribui significativamente para a economia ao redor do mundo. Além de empregar milhões de pessoas, tem participação significativa no PIB de vários países. Essa indústria passou por importantes mudanças ao longo dos tempos, com o desenvolvimento de novas tecnologias e métodos modernos de produção. Apesar desses avanços, a indústria têxtil ainda encontra desafios relacionados à qualidade de produção do tecido. Os defeitos têxteis podem afetar significativamente a qualidade do tecido, levando a vendas reduzidas e menor capacidade de manutenção [1].

No Brasil, conforme dados do Programa de Internacionalização da Indústria Têxtil e de Moda Brasileira [2], o faturamento da cadeia têxtil e de confecção chegou a R\$ 190 bilhões em 2022. O setor foi o segundo maior empregador da indústria de transformação, com 19,5% dos empregos, demonstrando sua relevância.

Os principais problemas enfrentados pela indústria têxtil são a grande quantidade de defeitos e a baixa qualidade da produção por conta da inspeção ser realizada de forma manual através do olhar humano [3].

A visão computacional é uma subárea da inteligência artificial que permite que às máquinas entendam o mundo

Os autores agradecem ao Ifes, apoio da FAPES e CAPES (proc 2021-2S6CD, nº FAPES 132/2021) do PDPG (Programa de Desenvolvimento da Pós-Graduação, Parcerias Estratégica nos Estados).

visual de maneira similar aos seres humanos. Essa tecnologia está transformando diversas indústrias e pode ser utilizada para garantir a qualidade na fabricação têxtil por meio da automação.

O YOLOv8 é o modelo mais recente da série YOLO (*You Only Look Once*) para detecção de objetos e segmentação de imagens em tempo real. Baseado em seus predecessores, introduz novos recursos e melhorias para aprimorar o desempenho, flexibilidade e a eficiência, oferecendo suporte a várias tarefas de IA na visão computacional [4].

O presente estudo apresenta a aplicação da tecnologia YOLOv8 para a detecção de defeitos em tecidos utilizando o conjunto de dados AITEX¹. Esta abordagem representa um avanço em relação a modelos anteriores, como o YOLOv5, ambos desenvolvidos pela empresa Ultralytics. O conjunto de dados AITEX é notoriamente desafiador, abrangendo uma ampla variedade de defeitos têxteis, incluindo objetos de pequenas dimensões e parcialmente ocultos. A comparação dos resultados obtidos com estudos correlatos demonstra melhorias substanciais em termos de precisão e eficiência na identificação de defeitos, destacando a evolução e a eficácia da nova tecnologia.

O artigo está organizado conforme detalhes a seguir. A seção 2 apresenta alguns estudos recentes que abordam os tópicos relacionados a este trabalho. Na seção 3 serão apresentados os materiais e métodos. Na seção 4 estão os experimentos e resultados e por fim a seção 5 traz as considerações finais e trabalhos futuros.

II. TRABALHOS RELACIONADOS

Ao longo dos últimos anos, diversos pesquisadores propuseram melhorias e alterações no modelo original do YOLO. Criado inicialmente por [6] na Universidade de Washington, a primeira versão YOLOv1 foi lançada em 2016 e foi o primeiro modelo de detecção de objetos em tempo real a alcançar alta precisão.

A detecção de defeitos têxteis baseado em um modelo aprimorado do YOLOv4 foi apresentado por [7]. Em sua proposta, uma camada convolucional deformável foi adicionada ao CSPDarknet53 da rede *backbone* do YOLOv4 e

¹ [5] Dataset: <https://www.aitex.es/afid/>

algumas camadas de convolução 3x3 foram substituídas por convolução deformável. Várias técnicas de *data augmentation* foram realizadas, enriquecendo a base de dados original Ali Tianchi². Um agrupamento K-means foi realizado no conjunto de dados para obter âncoras mais adequadas. As métricas de avaliação utilizadas na pesquisa foram o *precision* (P), *recall* (R), *F1 score*, *average precision* (AP), *mean average precision* (mAP). Os resultados demonstraram que o mAP cresceu 11,6%, atingindo 81,26%.

A Fig. 1 ilustra duas imagens aumentadas do conjunto de dados comparando o desempenho do algoritmo proposto. A primeira (superior) uma imagem original e a segunda (inferior) o resultado da detecção após a aplicação do algoritmo.

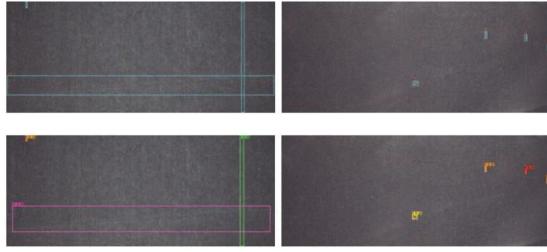


Figura 1. Resultado da detecção de defeitos com YOLOv4-DCN [7].

Otimizar o modelo YOLOv8 para detecção de objetos pequenos também foi uma proposta de estudo realizada por [9]. De acordo com a pesquisa, o YOLOv8 apresenta dificuldades em lidar com objetos pequenos menores que 8*8, pois sua cabeça de detecção é predefinida, insuficiente para detectar detalhes de objetos pequenos. Como proposta, os autores otimizaram a arquitetura da cabeça de detecção, diminuindo seu campo de percepção para identificar alvos acima de 4*4.

Assim, para melhorar o desempenho, é introduzida uma nova camada de detecção de objetos para identificar alvos menores. A Fig. 2 ilustra a camada adicionada a arquitetura.

```
- [-1, 1, nn.Upsample, [None, 2, "nearest"]]
- [[-1, 2], 1, Concat, [1]] # cat backbone P2
- [-1, 3, C2f, [128]] # 18 (P2/4-xsmall)
```

Figura 2. Camada adicionada a arquitetura do YOLOv8.

Ao comparar o modelo otimizado com o modelo original pode-se observar uma clara melhoria. Após adicionar uma nova camada de detecção, especificamente ao treinar a base de dados SOD, o mAP@0,5 aumentou de 74,2% para 78,4%.

A tecnologia permanece em constante evolução. Assim, os resultados apresentados por [10] foram aprimorados em [11], aplicando novas técnicas e versões do YOLOv5 na base de dados aumentada a partir da AITEX original. Desta forma foi possível melhorar os resultados obtidos e tracejar uma linha de pesquisa evolutiva em consonância com a evolução do YOLO.

Os modelos pré-treinados obtiveram resultados mais relevantes na detecção de defeitos têxteis, localizando e classificando os defeitos de forma eficiente atingindo um

mAP@0,5:0,95 de 60,12% e superando um resultado alcançado na pesquisa anterior de 56,06%.

A Tabela I apresenta as métricas e os resultados alcançados no estudo de [11] comparando com outros trabalhos correlatos. Observa-se melhores resultados alcançados na precisão, F1-score e mAP@0,5.

Tabela I
TRABALHOS CORRELATOS E RESULTADOS OBTIDOS - ADAPTADO DE [11]

Autores	Base	Modelo	Precision	F1-Score	mAP 0.5
Silvestre-Blanes (2019b)	AITEX	Gabor Filter	-	-	-
Seçkin and Seçkin (2022)	AITEX	IFV	82,68%	82,58%	-
Zheng et al. (2021)	Tianchi	SE-YOLOv5	85,33%	62,21%	-
Zheng et al. (2021)	Tianchi	YOLOv5	79,72%	60,40%	-
Seidel, Komati e Junior (2022)	AITEX	YOLOv5s	89,66%	84,36%	84,40%
Seidel, Waiandt e Junior (2023)	AITEX	YOLOv5x	91,81%	87,63%	86,90%

Não menos importante, a otimização de hiperparâmetros para melhorar o desempenho do YOLO na detecção de objetos foi reproduzida por [12]. O estudo propôs a detecção de lesões tumorais cerebrais em imagens de ressonância magnética utilizando o YOLOv7 no conjunto de dados de imagens extraído do BraTS2020 no plano axial da sequência T1CE. Com o intuito de melhorar o desempenho do modelo, buscouse encontrar o melhor conjunto de hiperparâmetros através de três técnicas: busca aleatória (RS), algoritmo genético (GA) e otimização bayesiana (BO). Os resultados apresentados demonstraram que a otimização bayesiana foi o método mais eficiente para encontrar a combinação ideal de hiperparâmetros, trazendo um ganho de 8% quando comparado ao modelo original e alcançando um mAP de 91,60%. A pesquisa indicou que a otimização de hiperparâmetros pode sim melhorar o desempenho do modelo.

III. MATERIAIS E MÉTODOS

A. Base de dados

A base de dados utilizada neste estudo foi cedida por [11]. Trata-se de uma nova base de dados gerada a partir da base original AITEX, que passou por processos de *data augmentation*. O objetivo foi gerar novos dados a partir dos existentes para treinar um novo modelo de ML, permitindo assim a continuidade da pesquisa, avaliando a evolução tecnológica do YOLO através da comparação de resultados alcançados.

A base de dados AITEX criada por [13] contém 245 imagens de 7 tipos de tecidos diferentes. São 140 imagens sem defeitos, 20 para cada tipo de tecido e 105 imagens com diferentes tipos de defeitos. A nova base gerou um total de 802 amostras com defeitos após processos de *data augmentation*, mantendo a mesma proporção para treino (65%), validação (20%) e teste (15%) e está disponível em Roboflow Aitex Dataset³.

³ [14] Roboflow Dataset: <https://app.roboflow.com/aitex-data-set-waiandt/>

² [8] Dataset: <https://tianchi.aliyun.com/dataset/>

Na Fig. 3 são representados, conforme base de dados AITEX, exemplo de defeito 19 no tecido 02 e sua máscara (é mostrada apenas uma área de 256x256).

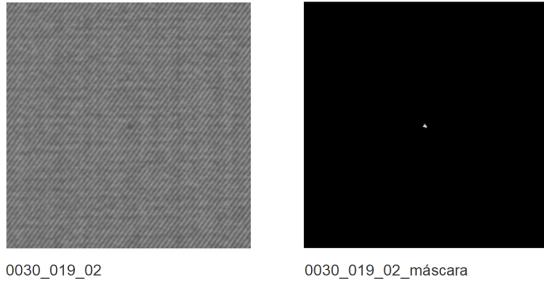


Figura 3. Defeito 19 no tecido 02 e sua máscara base de dados AITEX.

B. YOLOv8 - Um algoritmo de detecção de objetos

You Only Look Once (YOLO) é um modelo amplamente reconhecido para detecção de objetos e segmentação de imagens. Desenvolvido por [6] na Universidade de Washington, o YOLO foi lançado em 2015 e rapidamente ganhou popularidade devido à sua velocidade e precisão. Desde então, várias versões foram desenvolvidas.

O modelo utilizado neste trabalho é o YOLOv8, um dos mais recentes lançado em janeiro de 2023 [4]. Como um modelo de última geração da família YOLO, introduz novas funcionalidades e melhorias para alcançar um desempenho ainda melhor, maior flexibilidade e eficiência.

O YOLOv8 utiliza uma estrutura de rede aprimorada no *backbone*. O módulo CSPDarknet substituiu o módulo C3 original pelo C2f, proporcionando ainda mais leveza. Na seção *neck* é empregada uma arquitetura de rede de pirâmide de recursos PAN-FPN. O módulo principal *head* foi alterado de uma estrutura de acoplamento original utilizada no YOLOv5 para uma estrutura de desacoplamento (*decoupled head*). As funções de perda utilizadas são o *binary cross entropy* (BCE) para classificação e a combinação de *complete intersection over union* (CIOU) loss e *distribution focal loss* (DFL) para regressão. Essas escolhas aumentam a probabilidade da rede se concentrar rapidamente no local próximo ao alvo de detecção.

Na Fig. 4 observa-se a arquitetura de rede básica do YOLOv8, conforme estruturas de *backbone*, *neck* e *head* descritas anteriormente.

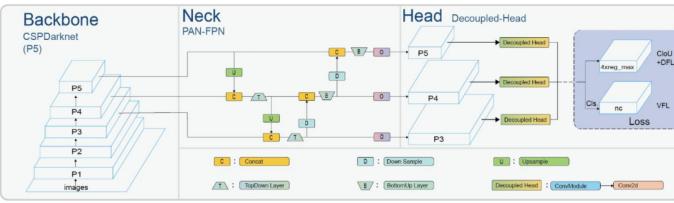


Figura 4. Arquitetura de rede básica do modelo YOLOv8.

Outra mudança importante quando comparado a modelos anteriores, o YOLOv8 abandona o método baseado em âncoras (*anchor-base*) e adota o método *anchor-free* para localização

da posição do alvo, melhorando a velocidade de detecção. Os métodos baseados em âncoras, como por exemplo o modelo YOLOv5, possuem alto mAP. A nova arquitetura do YOLOv8 trouxe diversas melhorias, mas também algumas limitações na detecção de objetos pequenos, impactando na precisão do modelo e prejudicando o mAP quando comparada a versões anteriores que utilizam âncoras em sua arquitetura [4].

De forma a contornar essa limitação, especificamente na cabeça de detecção, a arquitetura do YOLOv8 foi otimizada gerando uma variante chamada YOLOv8-P2 para detecção de objetos pequenos. O modelo padrão possui três cabeças de detecção, para alvos pequenos (8x8), médios (16x16) e grandes (32x32). Conforme demonstrado na Fig. 2, a variante P2 adiciona dois novos mapas de recursos de saída para identificar alvos acima de 4x4.

C. YOLOv5 versus YOLOv8

O YOLOv8 melhorou significativamente a precisão quando comparado ao YOLOv5. A Tabela II demonstra os resultados oficiais do mAP, parâmetros e flops testados no conjunto de dados COCO Val 2017, que é base utilizada pelo YOLO para treinamento dos modelos. O número de parâmetros e flops dos modelos n, s e m aumentaram significativamente. Pode-se observar também que a velocidade de inferência do YOLOv8 é mais lenta em comparação a maioria dos modelos YOLOv5, porém a quantidade de parâmetros utilizados nos modelos considerados mais pesados como l e x houve uma redução. Isso significa que o modelo consegue executar a mesma tarefa com uma quantidade menor de recursos.

Tabela II
RESULTADOS OFICIAIS DO MAP PARA O YOLOv5 E YOLOv8 [15]

model	size (pixels)	YOLOv5			YOLOv8		
		mAP 0.5-0.95	Params (M)	Flops (B)	mAP 0.5-0.95	Params (M)	Flops (B)
n	640	28.0	1.9	4.5	37.3	3.2	8.7
s	640	37.4	7.2	16.5	44.9	11.2	28.6
m	640	45.4	21.2	49.0	50.2	25.9	78.9
l	640	49.0	46.5	109.1	52.9	43.7	165.2
x	640	50.7	86.7	205.7	53.9	68.2	257.8

D. Ambiente de treinamento

O ambiente experimental e as configurações específicas adotados para este estudo são apresentados abaixo:

- Plataforma: Google Colab.
- CUDA: Versão 12.2.
- GPU: Tesla T4-15GB, NVIDIA L4-22.5GB e NVIDIA A100-SXM4-40GB.
- Software: Python, via linha de comando CLI.

E. Otimização de Hiperparâmetros

Em diferentes casos, as configurações apropriadas de hiperparâmetros têm um impacto significativo no desempenho dos modelos de aprendizagem profunda. O correto dimensionamento desses parâmetros influencia diretamente no desempenho, velocidade e precisão do modelo [4].

A otimização bayesiana surge como o método mais rápido para encontrar o conjunto de dados ideal de hiperparâmetros por lidar com dados contínuos [12], sendo a técnica empregada nesta pesquisa para otimizar os hiperparâmetros durante o treinamento do modelo.

As principais configurações de treinamento incluem: *batch*, taxa de aprendizagem (*lr0*), redução de peso (*weight decay*), imagem (*imgsz*), otimizador (*optimizer*) e aumento (*augment*). Além disso, a função de perda (*loss function*) e a composição do conjunto de dados de treinamento também têm impacto significativo no processo de treinamento.

Na Tabela III são apresentados os hiperparâmetros e valores sugeridos para realização do treinamento do modelo.

Tabela III
HIPERPARÂMETROS OTIMIZADOS PARA TREINAMENTO DO MODELO

Otimizados	Valor	Outros	Valor
augment	true	task	detect
batch	32	mode	train
imgsz	1280	model	yolov8(n,s,m,l,x)
lr0	0.10	epochs	100 200 300
optimizer	auto	pretrained	true

F. Métricas de Avaliação

Para avaliar o desempenho da evolução do YOLOv8 quando comparado a outros modelos, as métricas utilizadas neste artigo foram o mAP (*Mean Average Precision*), *recall*, *precision* e F1-score calculado a partir do *precision* e *recall*, conforme representado em (1).

$$F1 = \frac{2 \times Precision \times Recall}{(Precision + Recall)} \quad (1)$$

O *precision* e *recall* são representados em (2) e (3).

$$Precision = \frac{TP}{(TP + FP)} \quad (2)$$

$$Recall = \frac{TP}{(TP + FN)} \quad (3)$$

sendo:

- Número de detecções verdadeiras positivas (TP): Indica o número de objetos que o modelo detectou de forma correta.
- Número de detecções falsas positivas (FP): Indica o número de objetos que o modelo detectou incorretamente.
- Número de detecções falsas negativas (FN): Refere-se ao número de objetos reais que o modelo não detectou.

O mAP é uma métrica de avaliação usada para detecção de objetos (localização e classificação). Calcula a área sob a curva de precisão - recuperação, fornecendo um valor único. É calculada com um limiar de intersecção sobre união (IoU) de 0,50, representada em (4).

$$mAP = \frac{1}{N} \sum_{i=1}^N AP_i \quad (4)$$

Para fornecer uma avaliação mais abrangente, o mAP@0,5:0,95 estende a avaliação para uma faixa de limites de IoU de 0,5 a 0,95. É a principal métrica para avaliar o desempenho de modelos de detecção de objetos como o YOLO. Fornece uma avaliação abrangente da capacidade do modelo de identificar e localizar objetos em imagens calculando a média *precision* em diferentes níveis de *recall*, representada em (5).

$$mAP@0,5:0,95 = \frac{1}{10} \times \sum_{IoU=0,5;i=0,05}^{0,95} \left(\sum_{c=1}^9 Precision(Recall_c)_{@IoU} \right) \quad (5)$$

- mAP = 0,95 indica que o modelo está detectando e localizando objetos com alta precisão.
- mAP = 0,50 indica que o modelo precisa de mais treinamento ou ajustes para ter um desempenho satisfatório.

IV. EXPERIMENTOS E RESULTADOS

O modelo YOLOv8 foi aplicado à base de dados AITEX para detecção de defeitos têxteis. Seu desempenho foi comparado com outros modelos e principalmente o YOLOv5, empregado nos estudos realizados por [11], utilizando o mesmo conjunto de dados e técnicas para reproduzir um ambiente semelhante de forma a avaliar a evolução tecnológica do modelo para a visão computacional.

Foram realizados diversos experimentos com todas as variantes do YOLOv8 para avaliar a aplicabilidade do modelo. Por apresentar um contexto de defeitos pequenos a variante P2 do YOLOv8, otimizada para esse propósito, apresentou os melhores resultados. Como pode ser observado na Fig. 2, foi introduzida uma nova camada de detecção de objetos a sua arquitetura para identificar alvos ainda menores.

Os experimentos foram conduzidos em *epochs* de 100, 200 e 300. A otimização de hiperparâmetros utilizando a técnica bayesiana proporcionou ao modelo a adequação às melhores configurações de treinamento. A Tabela IV demonstra uma comparação dos hiperparâmetros selecionados.

Tabela IV
COMPARAÇÃO DE HIPERPARÂMETROS PARA TREINAMENTO

YOLOv8 - este estudo		YOLOv5 [11]	
Hiperparâmetro	Valor	Hiperparâmetro	Valor
lr0	0.01	lr0	0.01
lrf	0.01	lrf	0.01
momentum	0.937	momentum	0.937
batch	16	batch	16
weight decay	0.001	weight decay	0.0005
optimizer	auto	optimizer	SGD
augment	true	augment	true
mosaic	1.0	mosaic	1.0
imgsz	1024 e 1280	imgsz	1024 e 1280
anchor_t	null	anchor_t	4.0

Comparando-os com os hiperparâmetros aplicados no estudo de [11] com o modelo YOLOv5, destacam-se alguns

hiperparâmetros que diferenciam o treinamento de ambos. Dentre eles, “*anchor*”, que foi abandonado no modelo YOLOv8 e passou a usar o conceito de *anchor free*, “*optimizer*”, que no caso do YOLOv8 se adaptou melhor à opção auto, além do hiperparâmetro “*weight decay*”.

Inicialmente foi construído o modelo YOLOv8s-P2 com um treinamento em 100 *epochs* adotando no argumento *pretrained* o modelo YOLOv8x para transferência de aprendizagem carregando os pesos já treinados. Os modelos mais pesados do YOLOv8, como l e x, não se adaptaram bem ao conjunto de dados apresentando resultados baixos, alto custo e tempo de treinamento. Desta forma, somente foram utilizados para transferência de aprendizado.

A Fig. 5 apresenta as configurações de parâmetros da escala s do modelo P2 realizada em um arquivo “.yaml”. Os modelos mais leves n e s foram os que melhor se adaptaram ao conjunto de dados durante o treinamento.

```
# Parameters
nc: 09 # number of classes
#scales: model compound scaling constants
# [depth, width, max_channels]
# n: [0.33, 0.25, 1024]
# s: [0.33, 0.50, 1024]
# m: [0.67, 0.75, 768]
# l: [1.00, 1.00, 512]
# x: [1.00, 1.25, 512]

# Selected scale architecture = "s"
depth_multiple: 0.33
width_multiple: 0.50
max_channels: 1024
```

Figura 5. Escala selecionada para a arquitetura YOLOv8s-P2.yaml.

O modelo gerado deu sequência aos treinamentos em 200 e 300 *epochs*. Os resultados alcançados foram satisfatórios, demonstrando que o modelo específico P2 do YOLOv8 se adaptou muito bem ao conjunto de dados. Os resultados alcançados para cada métrica após a finalização dos treinamentos em 300 *epochs* foram um *recall* de 81,50%, *precision* de 93,50%, mAP@0.5 de 85,11% e um mAP@0.5:0.95 de 57,43%. A Fig. 6 apresenta a evolução dos resultados alcançados das principais métricas durante os treinamentos realizados.

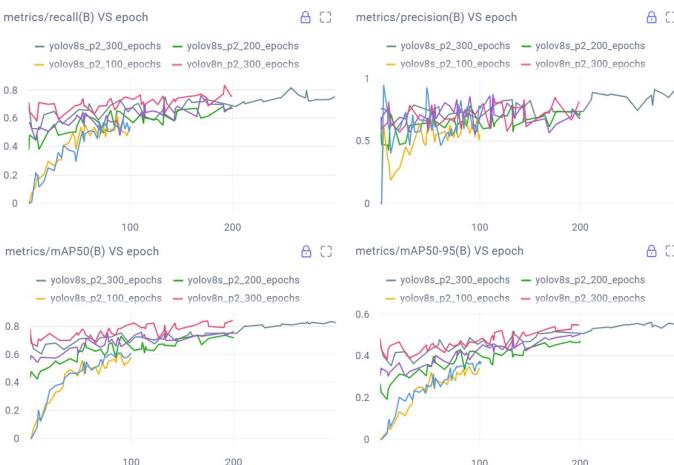


Figura 6. Resultados métricas de desempenho.

A curva *precision-recall* é uma ferramenta importante para avaliar o desempenho de um modelo de detecção de objetos. O *precision* pode ser uma métrica enganosa quando analisada isoladamente para conjuntos de dados desequilibrados, por isso geralmente é discutida em conjunto com o *recall*.

A Fig. 7 representa o gráfico da curva *precision-recall* para o treinamento realizado com o YOLOv8s-P2 em 300 *epochs*. A análise sugere que o modelo convergiu significativamente bem com essa versão do YOLOv8. O mAP@0.5 subiu de 58,01% nos treinamentos iniciais com as configurações padrões do YOLOv8 para um mAP@0.5 de 85,11% após avanço nos estudos e otimização do desempenho, representando aproximadamente um ganho de 47% no desempenho do modelo.

Analizando o gráfico têm-se que a maior parte das classes de defeitos melhoraram as métricas de desempenho durante a evolução do treinamento. A classe ‘025 - Warp Ball’ foi uma das que não apresentou evolução quando comparada ao treinamento inicial. Esta classe de defeito representa muito bem o contexto de aplicação deste estudo que são os objetos pequenos de difícil detecção. As classes ‘016 – Weft Curling’, ‘002 – Broken End’ e ‘023 - Crease’ obtiveram os melhores resultados seguidas das classes ‘006 – Broken Yarm’ e ‘022 – Cut Selvage’, contribuindo assim para um melhor desempenho do mAP.

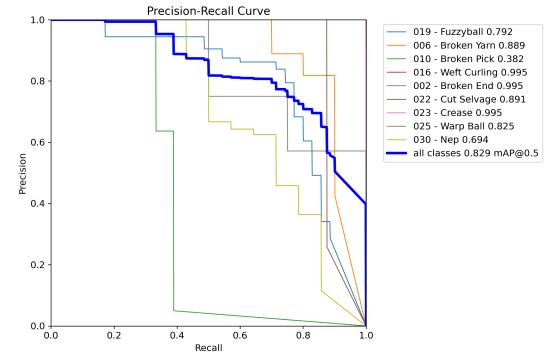


Figura 7. Curva *precision-recall* treinamento YOLOv8s-P2, 300 *epochs*.

A Tabela V apresenta as métricas de desempenho do modelo YOLOv8 aplicado no conjunto de dados AITEX para detecção de objetos têxteis após a realização de todo treinamento. Os resultados alcançados evoluíram à medida que os estudos foram avançando e técnicas e métodos foram aplicados.

Tabela V
RESULTADOS TREINAMENTO YOLOV8

Fase	Modelo	Base	mAP 0.5	mAP 0.5:0.95	Precision	Recall	F1-Score
Início	YOLOV8n	AITEX	58,01%	41,11%	81,90%	59,34%	68,82%
Otimização	YOLOV8n	AITEX	65,33%	50,38%	86,21%	63,59%	73,19%
Final	YOLOV8s-p2	AITEX	85,11%	57,43%	93,50%	81,50%	87,09%

Quando comparado a outros estudos relevantes conforme mencionados na Tabela I, o modelo YOLOv8s-P2 alcançou resultados significativos. Para os estudos que utilizaram a

mesma base de dados AITEX, o F1-Score de 87,09% superou os estudos de [16] com 81%, [17] 82,58% e [10] com 84,36%. O F1-Score também foi melhor quando comparado aos estudos que utilizaram outras bases de dados como [18]. O *precision* de 93,50% superou todos os resultados apresentados na Tabela I. O mAP@0.5, uma das principais métricas de detecção, com um valor de 85,11% superou quase todos os estudos apresentados, ficando pouco abaixo somente do estudo de [11] com o YOLOv5x, não representando diferença significativa na prática.

A Fig. 8 representa um exemplo dos resultados da detecção e classificação de defeitos em imagens da base de dados AITEX após realização do treinamento e validação. Classes de defeito: 2 “010 - Broken pick” - 0 “019 Fuzzyball” - 1 “006 - Broken Yarn” - 3 “016 - Weft Curling” - 4 “002 - Broken End”.

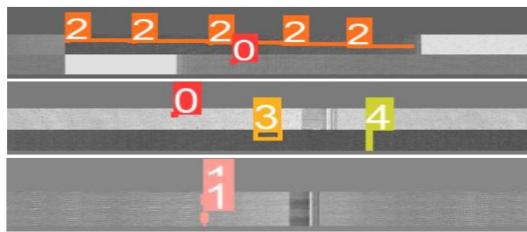


Figura 8. Resultados da detecção de defeitos.

V. CONCLUSÃO E TRABALHOS FUTUROS

Inspirado no sucesso de seus predecessores o YOLOv8 introduziu novas funcionalidades expandindo os limites da detecção de objetos. Com o avanço dos estudos, percebeu-se que o principal objetivo ao desenvolver a arquitetura do YOLOv8 foi deixar a inferência mais rápida, tornando-o um excelente modelo para detecção de objetos em tempo real. Outra mudança impactante foi a implementação de uma nova cabeça para detecção de objetos, que eliminou a necessidade de caixas de âncora predefinida. Essas modificações, em conjunto com as mudanças arquitetônicas, equilibraram a velocidade e precisão de uma forma geral deixando-o mais rápido. Porém, quando aplicado em contextos específicos, como objetos pequenos, o modelo, agora sem ancoras, diminuiu sua capacidade de detecção de objetos.

Contudo, analisando a métrica mAP@0.5, de acordo com a Tabela V, os resultados alcançados nos treinamentos desde o início até a versão final evoluíram mais de 46,70%. Comparado ao modelo YOLOv5, aplicado nos estudos realizados por [11] e outros modelos de trabalhos correlatos, os resultados alcançados foram promissores. Destacam-se a métrica *precision*, que alcançou o maior valor entre os modelos e o F1-Score.

De uma forma geral o modelo YOLOv8 atende bem ao contexto de detecção de objetos pequenos na indústria têxtil. Durante os treinamentos, as análises realizadas através das métricas alcançadas demonstraram que o modelo é evolutivo e que novos experimentos poderão ser realizados em *epochs*

de 400 e 500. Trabalhos futuros poderão utilizar o conjunto de dados deste estudo disponível no roboflow como fonte de dados para novas pesquisas, além de treinar o modelo em novas versões do YOLO e em outros algoritmos de detecção de objetos de forma a evoluir o modelo com novas tecnologias.

REFERÊNCIAS

- [1] A. Mehta and R. Jain, “An analysis of fabric defect detection techniques for textile industry quality control,” in *2023 World Conference on Communication & Computing (WCONF)*, 2023, pp. 1–5.
- [2] P. d. I. d. I. T. e. d. M. B. TEXBRASIL, “Dados da indústria têxtil,” 2023. [Online]. Available: <https://texbrasil.com.br/pt/impressa/dados-da-industria-textil-e-de-confeccao/>
- [3] K. Ramakrishnan, P. G. Jayakumar, P. Saravanan, and P. Sivakumar, “A novel fabric defect detection network in textile fabrics based on dl,” in *2023 International Conference on Artificial Intelligence and Knowledge Discovery in Concurrent Engineering (ICECONF)*, 2023, pp. 1–8.
- [4] G. Jocher, A. Chaurasia, and J. Qiu, “Ultralytics yolov8,” 2023. [Online]. Available: <https://github.com/ultralytics/ultralytics>
- [5] F. I. D. AITEX, “Fabric image database aitex,” 2019. [Online]. Available: <https://www.aitex.es/afid/>
- [6] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, “You only look once: Unified, real-time object detection,” in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 779–788.
- [7] T. Liu and S. Chen, “Yolov4-dcn-based fabric defect detection algorithm,” in *2022 37th Youth Academic Annual Conference of Chinese Association of Automation (YAC)*, 2022, pp. 710–715.
- [8] D. Ali Tianchi, “Database ali tianchi,” 2022. [Online]. Available: <https://tianchi.aliyun.com/dataset/>
- [9] H. Huang, B. Wang, J. Xiao, and T. Zhu, “Improved small-object detection using yolov8: A comparative study,” *Applied and Computational Engineering*, vol. 41, pp. 80–88, 02 2024.
- [10] R. Seidel, H. S. Junior, and K. Komati, “Textile defect detection using yolov5 on aitex dataset,” in *Anais do XIX Encontro Nacional de Inteligência Artificial e Computacional*. Porto Alegre, RS, Brasil: SBC, 2022, pp. 763–774. [Online]. Available: <https://sol.sbc.org.br/index.php/niac/article/view/22830>
- [11] R. Seidel, C. Waiandt, and H. S. Junior, “Defect detection in textile manufacturing: Improving findings using yolov5 versions,” in *Anais do XVIII Workshop de Visão Computacional*. Porto Alegre, RS, Brasil: SBC, 2023, pp. 66–71. [Online]. Available: <https://sol.sbc.org.br/index.php/wvc/article/view/27534>
- [12] D. Wahyudi, I. Soesanti, and H. A. Nugroho, “Optimizing hyperparameters of yolo to improve performance of brain tumor detection in mri images,” in *2023 6th International Conference on Information and Communications Technology (ICOIACT)*, 2023, pp. 413–418.
- [13] J. S. Blanes, T. Albero, I. Miralles, R. Pérez-Llorens, and J. Moreno, “A public fabric database for defect detection methods and results,” *Autex Research Journal*, vol. 19, no. 4, pp. 363–374, 2019. [Online]. Available: <https://doi.org/10.2478/aut-2019-0035>
- [14] P. Roboflow, “Roboflow,” 2024. [Online]. Available: https://app.roboflow.com/aitex-data-set-waiandt/aitatex-double_mixed_full_daset/4
- [15] H. Ultralytics, “Ultralytics yolo docs,” mai 2024. [Online]. Available: <https://docs.ultralytics.com/>
- [16] Q. Zhou, J. Mei, Q. Zhang, S. Wang, and G. Chen, “Semi-supervised fabric defect detection based on image reconstruction and density estimation,” *Textile Research Journal*, vol. 91, no. 9-10, pp. 962–972, 2021.
- [17] A. Ç. Seçkin and M. Seçkin, “Detection of fabric defects with intertwined frame vector feature extraction,” *Alexandria Engineering Journal*, vol. 61, no. 4, pp. 2887–2898, 2022.
- [18] L. Zheng, X. Wang, Q. Wang, S. Wang, and X. Liu, “A fabric defect detection method based on improved yolov5,” in *2021 7th International Conference on Computer and Communications (ICCC)*. IEEE, 2021, pp. 620–624.