

Textile defect detection using YOLOv5 on AITEX Dataset

Rodolfo Seidel¹, Hilário Seibel Júnior¹, Karin Satie Komati¹

¹Programa de Pós-graduação em Computação Aplicada (PPComp)

Instituto Federal do Espírito Santo (IFES)

Av. dos Sabiás, 330 – 29.166-630 – Serra – ES – Brazil

rodsei@gmail.com, {hsjunior, kkomati}@ifes.edu.br

Abstract. *Due to the manual identification of textile defects still nowadays, it is necessary to find methods to detect defects in an automated and efficient way. Hence, this work applies YOLOv5 model in the AITEX¹ dataset, using an object detection approach to locate and identify defects, evaluating different object annotation and image data augmentation techniques. From the results, it is concluded that YOLOv5 handled very well another context with distinct objects from the pre-training, annotations using Bounding Boxes allowed greater learning and recognition of defects, even with different shapes and sizes, and at last, the combination of augmentations boosted its performance.*

Resumo. *Devido à identificação manual de defeitos têxteis ainda nos dias atuais, é necessário encontrar meios de detectar defeitos de forma automatizada e eficiente. Para isso, este trabalho se propõe a aplicar o modelo YOLOv5 na base de dados AITEX, usando a abordagem de detecção de objetos para localizar e identificar defeitos, avaliando diferentes técnicas de anotação de objetos e data augmentation. Com os resultados obtidos, concluiu-se que o YOLOv5 adaptou-se muito bem a outro contexto com objetos distintos do pré-treinamento, as anotações com Bounding Boxes permitiram maior aprendizado e reconhecimento dos defeitos, mesmo com diferentes formas e tamanhos, e por fim, a combinação de data augmentation potencializam seu desempenho.*

1. Introdução

A indústria em geral está em meio à revolução da Indústria 4.0. Sua principal característica é a possibilidade da automação completa do sistema produtivo através da Inteligência Artificial, eliminando erros causados pelo fator humano [Popkova et al. 2019]. Na indústria têxtil, suas particularidades dificultam a adoção de certas tecnologias em alguns processos, que continuam sendo feitos em sua maioria de forma manual e demandam um tempo significativo [Petronas 2020].

Um desses processos é a revisão de tecidos. Por possuir uma subjetividade na definição de um defeito, que é praticamente qualquer anomalia presente na superfície, é feito de forma manual desde as primeiras iniciativas de automação. Somado a esse aspecto, a capacidade humana de identificação e análise variam ao longo do tempo de trabalho [Conceição 1998]. Além da fadiga, existem defeitos quase imperceptíveis que passaram a ser mais comuns com a evolução dos maquinários. Esse avanço desafia até mesmo

¹[Silvestre-Blanes et al. 2019] *AITEX Dataset*: <https://www.aitex.es/afid/>

as pesquisas que buscam desenvolver um modelo robusto e eficiente para a detecção de defeitos, o que traz grande relevância para as pesquisas nesse campo [Jun et al. 2021].

Relatos recentes mostram que as inspeções de defeitos continuam sendo realizados em sua maioria por humanos. Apesar dos profissionais qualificados, registra-se uma acurácia manual próxima de 70%. Nesse cenário, os sistemas de inspeção visual podem otimizar o processo de produção, mantendo a detecção de defeitos têxteis em destaque no contexto de visão computacional [Jing et al. 2022].

Em uma das pesquisas, foram coletadas imagens em uma fábrica através de um sistema real. A partir daí, criaram a base de dados AITEX para possibilitar a comparação adequada entre métodos de detecção de defeitos [Silvestre-Blanes et al. 2019]. Nesse período, já existiam algumas versões da abordagem proposta no modelo de detecção de objetos YOLO [Redmon et al. 2015]. No ano seguinte, foram lançadas a YOLOv4 [Bochkovskiy et al. 2020] e a YOLOv5 [Jocher et al. 2020]. Após a publicação das novas versões, até onde é de conhecimento, poucos trabalhos foram realizados com a versão mais recente no contexto de detecção de defeitos têxteis com a base AITEX. Os artigos de [Zheng et al. 2021] e [Jin and Niu 2021] utilizaram outras bases de imagens públicas.

Este estudo se propõe a detectar defeitos têxteis aplicando a versão mais recente do YOLOv5 [Jocher et al. 2020] na base AITEX [Silvestre-Blanes et al. 2019]. Serão utilizadas as devidas técnicas e adaptações para adequar as imagens ao formato de entrada do modelo sem comprometer as características originais das imagens. Assim será possível comparar com os resultados dos demais trabalhos que utilizaram esta mesma base.

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, estão mais detalhes das características da base de dados AITEX, do modelo pré-treinado do YOLOv5, das ferramentas utilizadas na preparação da base das métricas de treinamento e validação. Já a seção 4 demonstra os experimentos e avanços obtidos, seguidos da conclusão e proposta de trabalhos futuros na seção 5.

2. Trabalhos Relacionados

Em [Ouyang et al. 2019], foi proposta a rede neural convolucional PPAL-CNN avaliando diferentes funções de ativação, comparando-as com o SegNet, outro modelo de segmentação de imagens. Para realizar as avaliações do modelo, todas as imagens da base foram inspecionadas manualmente para realizar as anotações dos defeitos. Para a avaliação do modelo, foram utilizadas duas bases, uma própria dos autores contendo 1160 iamgens e a base pública TILDA², ambas com resultados similares e bem superiores ao SegNet.

Visando criar uma ferramenta de detecção eficiente, [Jing et al. 2022] propuseram a Mobile-Unet com um foco no desempenho otimizado para tarefas em tempo real, apresentando uma convolução separável em profundidade com o MobileNetV2. Outra característica de destaque foi a utilização da função de perda CE-MFB (*cross-entropy loss with median frequency balancing*) para lidar com o desbalanceamento dos dados conforme demonstrado na Figura 1, aplicando pesos ponderados de acordo com a participação de cada

²TILDA: base pública desenvolvida pelo grupo de análise têxtil DFG (*Deutsche Forschungsgemeinschaft*) disponível em <https://lmb.informatik.uni-freiburg.de/resources/datasets/tilda.en.html>

classe em toda a base. Com isso, alcançaram resultados superiores ao modelo U-Net com menos recursos computacionais e menos parâmetros, e consequentemente menor tempo.

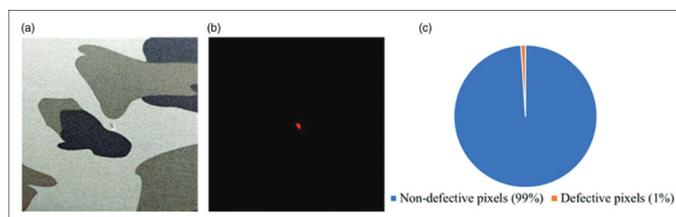


Figura 1. Quantidade desbalanceada de *pixels* defeituosos e sem defeitos. (a) Imagem original; (b) máscara com o defeito; (c) comparativo dos *pixels* presentes na imagem. [Jin and Niu 2021].

Na mesma linha de pesquisa, [Rong-qiang et al. 2021] construíram uma versão aprimorada da rede neural U-Net reduzindo o número de canais e adicionando um mecanismo de atenção com quatro vezes mais características que a original para aumentar a precisão da localização. Além disso a detecção de defeitos foi tratado como um problema de classificação binária, identificando se os *pixels* pertencem ou não à classe defeituosa. O modelo foi validado com a base AITEX e ao fim da execução, uma imagem segmentada é gerada e nela os defeitos detectados são marcados com um retângulo para facilitar a visualização dos resultados.

No estudo de [Zhou et al. 2021], foi criado um modelo híbrido com o modelo *Variational Autoencoder* (VAE) extraíndo as características e reconstruindo as imagens sem defeitos, tornando os defeitos identificáveis ao comparar a imagem original com a reconstruída. Depois, o *Gaussian mixture model* (GMM), utilizado também para detecção de anomalias, realiza a estimativa de densidade considerando como *outliers* os objetos que apresentarem baixa densidade, o que ocorre com os defeitos de forma natural uma vez que o modelo é treinado apenas com as imagens sem defeitos. Os experimentos foram validados com as bases AITEX e DAGM 2007³, fragmentando as imagens originais em cortes menores de 16×16 *pixels* para suprir a insuficiência de imagens no treinamento das reconstruções.

Com as possibilidades da Indústria 4.0 em mente, [Seçkin and Seçkin 2022] apresentaram o método *Intertwined Frame Vector* (IFV) com a intenção de detectar defeitos em aparelhos com baixo nível de processamento. Para isso são extraídas características de fragmentos aleatórios a partir de janelas reduzidas das imagens originais e posteriormente, são selecionadas as cinco características mais relevantes em conjunto com seus respectivos rótulos identificando o tipo de defeito. Por fim, esses dados são processados nos modelos tradicionais *K-Nearest Neighbors* (KNN) e *Random Forest* (RF).

Em [Wang et al. 2021], os autores destacam duas tarefas dos modelos de detecção de objetos, a localização e classificação das áreas, e ressaltam suas complexidades na presença de formas irregulares e pequenos objetos. Para lidar com isso, adicionaram uma nova estrutura na rede neural do YOLOv5 para filtrar características de outros níveis da rede, mantendo informações relevantes ao longo da cadeia. Essas características fo-

³DAGM 2007 Dataset [Wieler et al. 2007]. Disponível em <https://hci.iwr.uni-heidelberg.de/content/weakly-supervised-learning-industrial-optical-inspection>

ram tratadas em uma camada de transformação para potencializar as mais relevantes. A proposta foi construída na arquitetura do YOLOv5s e validada com uma base de dados contendo 3946 imagens no tamanho de 1000×1000 *pixels* com quatro tipos de defeitos, em sua maioria pequenos.

Já [Zheng et al. 2021] adicionaram no *backbone* do YOLOv5 um módulo SE (*Squeeze-and-Excitation*), construído sobre um operador convolucional que mapeia uma entrada para um mapa de características atuando como um mecanismo de atenção. Mecanismo este descrito como um viés de alocação dos recursos disponíveis para os componentes mais relevantes de um sinal. Além disso, usaram a função de ativação *ActivateOrNot* (ACON), extraído de características do mapa gerado e enriquecendo as informações obtidas da imagem. O modelo foi comparado com o YOLOv5 original em uma base de dados contendo 3622 imagens.

Também baseado no YOLOv5, [Jin and Niu 2021] apresentaram uma arquitetura professor-estudante com o objetivo de lidar com a baixa quantidade de imagens de defeitos disponíveis. Para a avaliação do modelo foram utilizadas as bases TILDA e outra disponibilizada pelo *Xuelang Tianchi AI Challenge*, com imagens rotuladas por *Bounding Boxes*. Nesse estudo, os autores destacam os problemas do desbalanceamento da base e da quantidade de tipos de defeitos com características muito diferentes entre si, necessitando de um mecanismo de atenção.

Na área de *data augmentation*, [Sakkos et al. 2019] destacam suas vantagens devido à grande quantidade de dados necessários para os algoritmos de *deep learning* e ao alto custo de obtenção de dados rotulados, somados à baixa disponibilidade de bases públicas. Eles propõem a criação de bases relevantes que aumentem o poder de generalização dos modelos com técnicas semânticas que façam sentido no contexto do problema. Os exemplos podem ser vistos na Figura 2, como alterações de iluminação locais e globais simulando variações plausíveis como um poste de luz durante a noite ou um clarão durante uma tempestade, combinando todas as variações possíveis.

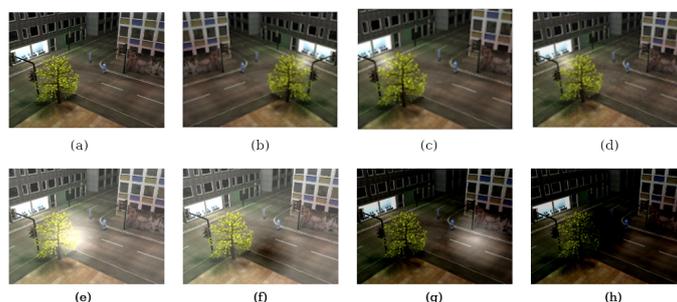


Figura 2. Combinação de efeitos semânticos em *data augmentation*. (a) Imagem original; (b) Inversão; (c) Centralização; (d) Adição de ruídos; (e) Clareamento global com brilho localizado; (f) Clareamento global com sombra localizada; (g) Sombreamento global com brilho localizado; (h) Sombreamento global com sombra localizada. Adaptada de [Sakkos et al. 2019]

3. Materiais e Métodos

3.1. Base de Dados

Neste trabalho foi utilizada a base de imagens AITEX, composta por 245 imagens de 7 tipos distintos de tecidos não identificados em um tamanho de 4.096×256 pixels, sendo 140 imagens sem defeitos e 105 com defeitos, distribuídos em 12 classes conforme a Tabela 1. Cada imagem contém uma ou mais máscaras com a localização e extensão de cada defeito (Figura 4).

Tabela 1. Distribuição de imagens na base geral e somente com defeitos).

Código	Classe	Tradução	Total Amostras	Distribuição Geral	Distribuição entre Defeitos
000	Defect-Free	Sem Defeito	140	56,91%	-
002	Broken End	Finalização quebrada	9	3,66%	8,49%
006	Broken Yarn	Fio quebrado	8	3,25%	7,55%
010	Broken Pick	Seleção quebrada	10	4,07%	9,43%
016	Weft Curling	Enroscos de trama	3	1,22%	2,83%
019	Fuzzyball	Bola felpuda	39	15,85%	36,79%
022	Cut Selvage	Corte de orela	9	3,66%	8,49%
023	Crease	Vinco	5	2,03%	4,72%
025	Warp Ball	Bola de urdidura	6	2,44%	5,66%
027	Knots	Nós	1	0,41%	0,94%
029	Contamination	Contaminação	1	0,41%	0,94%
030	Nep	Sujeira no fio	14	5,69%	13,21%
036	Weft Crack	Rachadura de trama	1	0,41%	0,94%

Ao mapear os defeitos como objetos, estes são descritos com sua classe, posição central no eixo X, posição central no eixo Y, largura e altura (Figura 5), com um objeto por linha e valores normalizados entre 0 e 1. Assim, o modelo estará apto a localizar e identificar as características de cada objeto dentro das áreas delimitadas [Jocher et al. 2020]. As anotações foram feitas manualmente a partir das máscaras da base original com *Bounding Boxes* e Polígonos através de uma ferramenta específica⁴ que calcula e gera os descritivos de todas as anotações visuais.

Durante os experimentos foram utilizadas técnicas de *data augmentation*. Baseado no estudo de [Sakkos et al. 2019], foram incluídas variações plausíveis em um ambiente industrial como ruídos de até 3% dos pixels, alterações de brilho entre -25% e +25% e inversões horizontais e verticais na captura das imagens (Figura 3). Cada uma das técnicas foi isolada em bases distintas e uma nova foi criada aplicando todas as variações simultaneamente, sendo quatro novas bases. Outras duas foram criadas contendo as amostras anteriores, uma nomeada Mista com as variações de Inversões, Brilhos e Ruídos e outra nomeada Mista Completa incluindo as amostras com Variações Simultâneas. Em todas elas, as imagens originais estavam presentes somente nas etapas de validação e teste.

3.2. Modelo de Aprendizagem

O YOLOv5 é uma família de arquiteturas e modelos pré-treinados de detecção de objetos apresentado por [Jocher et al. 2020], que utiliza o *framework* PyTorch e a base de dados COCO⁵, que por sua vez, reúne 123 mil imagens de objetos em seu contexto natural. Cada

⁴The Best Way to Annotate Images for Object Detection (<https://docs.roboflow.com/annotate/best-practices>)

⁵COCO dataset: Common Objects in Context (<https://cocodataset.org/#download>) [Lin et al. 2014]

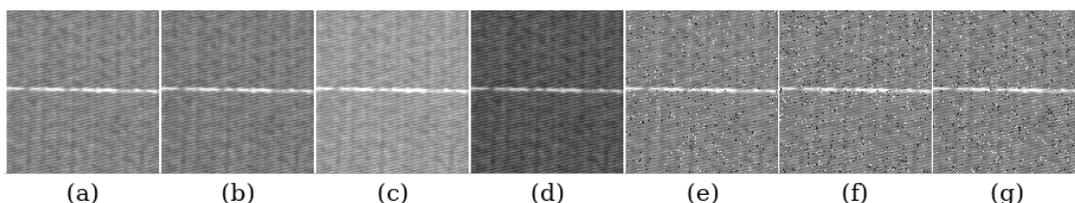


Figura 3. Amostras das técnicas de data augmentation.

(a) Imagem original; (b,c,d): Variações de brilho; (e,f,g) Adição de Ruídos.



Figura 4. Amostras de máscaras dos defeitos do tipo 010 (*Broken Pick*): diferentes formatos e variações.

modelo possui capacidades de execução e desempenho baseados em sua arquitetura e total de parâmetros utilizados, sendo o **YOLOv5s** o que apresenta o melhor custo-benefício utilizando uma GPU V100-16GB e *batch-size* 32.

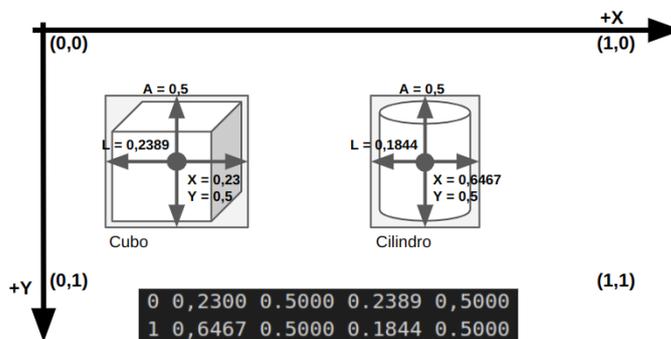


Figura 5. Anotação de objetos e arquivo descritivo. Detalhes: classe, eixo X, eixo Y, largura e altura; Classes: cubo (0) e cilindro(1). [Jocher et al. 2020]

Durante o treinamento, a IoU (*Intersection over Union*) e suas variações podem ser utilizadas como *loss function*, comparando a sobreposição existente entre a marcação predita e a marcação original. Ela está representada pela Equação 1, onde C_{PRED} e C_{REAL} são as características de cada marcação, ou seja, os eixos X e Y, largura e altura [Zheng et al. 2021]. Cada variação resolve um problema específico, aprimorando a velocidade de convergência e a precisão da localização. O GIoU (*Generalized IoU*) corrigiu o cálculo do gradiente quando não há sobreposição entre as marcações [Rezatofighi et al. 2019] e o DIoU (*Distance IoU*) adicionou uma penalidade para minimizar a distância entre os pontos centrais das marcações [Zheng et al. 2020]. As

diferenças de cálculo entre as funções podem ser observadas na Figura 6.

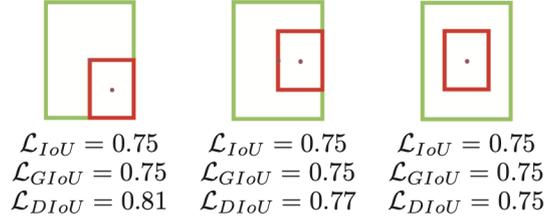


Figura 6. Comparativo entre as métricas IoU, GloU e DIoU. Em verde, a marcação real e em vermelho a predição.

[Zheng et al. 2020]

Além da distância na DIoU (Equação 2), onde p é a distância euclidiana entre os centros das formas A_{PRED} e A_{REAL} e c é o comprimento diagonal da menor forma que une as duas marcações, fatores geométricos das imagens durante a regressão foram tratados com dois novos parâmetros, sendo eles: v (Equação 3) como uma medida de consistência da proporção entre largura (w) e altura (h), e α (Equação 4) como um parâmetro de balanceamento. Essa é a CIoU (*Complete IoU*), utilizada como *loss function* padrão do YOLOv5 e representada na Equação 5 [Zheng et al. 2020].

$$IoU = \frac{|C_{PRED} \cap C_{REAL}|}{|C_{PRED} \cup C_{REAL}|} \quad (1)$$

$$DIoU = IoU + \frac{p^2(A_{PRED}, A_{REAL})}{c^2} \quad (2)$$

$$v = \frac{4}{\pi^2} * \left(\arctan \frac{w_{REAL}}{h_{REAL}} - \arctan \frac{w_{PRED}}{h_{PRED}} \right)^2 \quad (3)$$

$$\alpha = \frac{v}{IoU + v'} \quad (4)$$

$$CIoU = DIoU + \alpha v \quad (5)$$

3.3. Métricas de Avaliação

Para avaliar os resultados no aspecto classificação, foram utilizadas a precisão e o *recall*, calculadas a partir das detecções corretas de objetos existentes na imagem: **TP** (*True Positives*), das detecções incorretas de objetos que não existem: **FP** (*False Positives*) e das detecções não identificadas de objetos presentes na imagem: **FN** (*False Negatives*) [Zheng et al. 2021]. Já no contexto de detecção de objetos utiliza-se a métrica IoU, determinando o acerto ou erro através de um *threshold* e permitindo calcular a precisão e *recall*. Dessa forma, quanto maior a área sobreposta melhor [Jing et al. 2020].

Como métrica final foi utilizado o mAP (mean Average Precision), representada pela média das áreas abaixo da curva *Precision-Recall* (AP) de todas as classes e o mAP@0,5:0,95, combinando os resultados de vários *thresholds* da IoU, de 0,5

até 0,95, com intervalos de 0,05. Na detecção de objetos, o AP indica a margem de confiança da classificação e o *threshold* do IoU indica uma localização correta ou incorreta. Nesse cenário, o uso do mAP@0,5:0,95 visa reconhecer resultados que possuam uma localização mais apurada dos objetos [Arlen 2018].

4. Experimentos e Resultados

Os experimentos foram realizados em três fases. Na primeira utilizou-se as imagens originais da base para compreender seu comportamento orgânico e definir um *baseline*. Devido às diferentes formas entre defeitos de mesma classe, inicialmente foram utilizadas anotações por polígonos. Utilizando somente imagens com defeitos no treinamento os resultados alcançaram mAP@0,5 quase 6 vezes maior que a base completa (Figura 7), e mAP@0,5 e mAP@0,5:0,95 de 3 a 4 vezes maiores com imagens escaladas em 1024px e 1280px (Figura 8). O descarte das imagens sem defeitos diminuiu o desequilíbrio entre as classes.

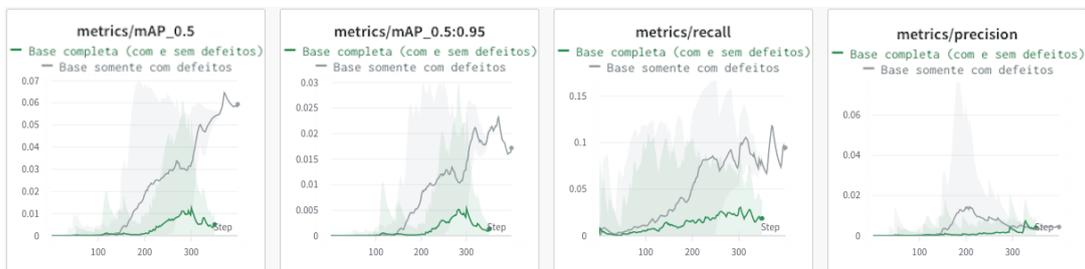


Figura 7. Comparativo das bases originais: completa e somente com defeitos.



Figura 8. Comparativo das escalas nas bases originais.

Na segunda fase, a partir dos comportamentos observados, foram utilizadas apenas as imagens com defeitos, descartando aqueles com apenas uma amostra cada e redistribuindo as imagens entre treino (65%), validação (20%) e teste (15%). Além disso, com os resultados dos polígonos abaixo do esperado, foram utilizadas as anotações em *Bounding Boxes*, repetindo o comportamento das imagens escaladas e alcançando melhorias significativas (Figura 9). Entende-se neste ponto que, apesar do algoritmo e funções de regressão estarem preparados para lidar com as anotações por polígonos, as várias formas existentes em uma mesma classe e sua irregularidade (Figura 4) fizeram com que o modelo tivesse uma menor capacidade convergência dentre as características identificadas, consequentemente, diminuindo o potencial de localização e identificação dos objetos.

A terceira fase se inicia com o *baseline* definido pela base otimizada e anotada com *Bounding Boxes*. Foram aplicadas as técnicas de *data augmentation* apresentadas

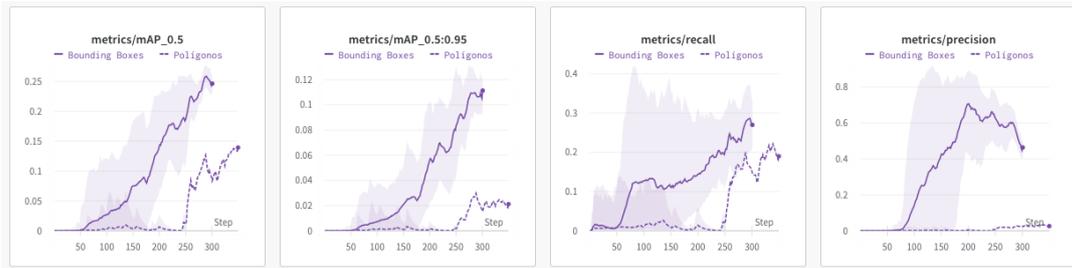


Figura 9. Comparativo dos tipos de anotação: *Bounding Boxes* e *Polígonos*.

anteriormente e os experimentos realizados nas novas bases com 300, 350 e 400 épocas nas resoluções de imagens de 1024px e 1280px. Os melhores resultados isolados foram com as bases de Inversões e Variação de Brilho, alcançando resultados de mAP@0,5 e mAP@0,5:0,95 quase 3 vezes acima do *Baseline* e quase o dobro de *precision* e *recall*. As bases com Adição de Ruídos e Variações Simultâneas obtiveram resultados ligeiramente abaixo do *Baseline* (Figura 10).

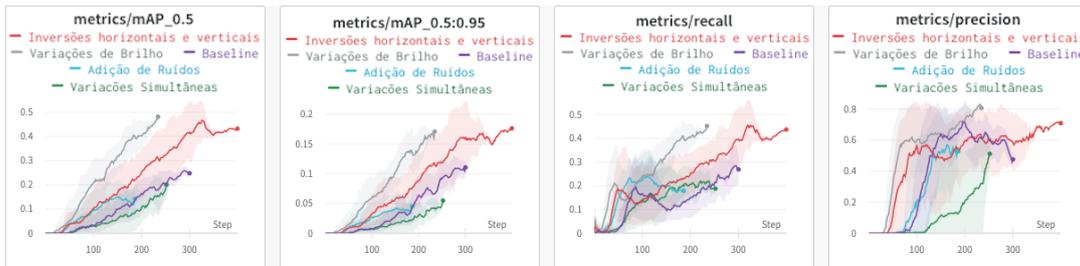


Figura 10. Comparativo entre as técnicas isoladas de *data augmentation*.

O desempenho das bases com as amostras combinadas alcançou um mAP@0,5:0,95 de 56,06%, muito superior aos 8,78% do *Baseline* além de evoluir também os resultados de *precision* e *recall* em duas vezes e meia e mais de quatro vezes, respectivamente, comparados ao *baseline*. Já os resultados inferiores ao *Baseline* observados de forma isolada com a base de Variações Simultâneas não prejudicaram o treinamento, mas sim, aumentou ligeiramente os resultados alcançados pela base Mista Completa em três das quatro métricas (Figura 11).

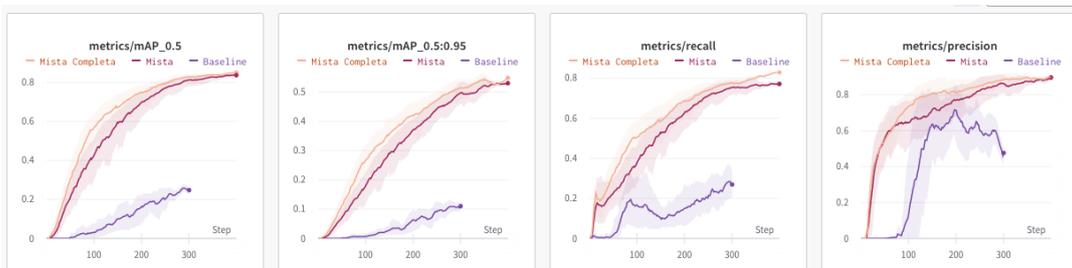


Figura 11. Comparativo das técnicas de *data augmentation* combinadas.

Na Tabela 2, vemos os resultados obtidos com cada base. Nas bases originais, ainda com poucas amostras de cada defeito, o modelo não conseguiu localizá-los e classificá-los corretamente, porém vemos a diferença entre os *Bounding Boxes* e *Polígonos*,

com o $mAP@0,5:0,95$ quatro vezes maior e *precision* vinte vezes superior. As bases mistas demonstraram um grande poder de generalização, mesmo com apenas três novas amostras de cada técnica mais suas aplicações simultâneas, em especial a Mista Completa destacada na tabela, alcançando um $mAP@0,5:0,95$ de 56,06%, *precision* de 89,66% e *recall* de 79,66%, sendo estes os melhores resultados obtidos. Esses resultados demonstram o poder do *data augmentation* e a capacidade de aprendizado do modelo YOLOv5, que com poucas variantes e técnicas de pré-processamento alcançou uma evolução significativa.

Nessa nova base, o YOLOv5s apresentou resultados superiores aos obtidos na base COCO, alcançando um desempenho maior até que o YOLOv5x6 em sua base original⁶. A partir desse comportamento, vemos que o modelo adaptou-se muito bem ao contexto de detecção de defeitos, mesmo com os objetos sendo em sua maioria pequenos e com proporções muito diferentes dos objetos presentes na base COCO.

Tabela 2. Resultados dos experimentos.

Base de Imagens	Resoluções	mAP 0.5	mAP 0.5:0.95	Precision	Recall	F1-Score
Mista Completa	1280 e 1024	84,4%	56,06%	89,66%	79,66%	84,36%
Mista	1280 e 1024	84,5%	55,00%	86,22%	78,54%	82,20%
Inversões	1280 e 1024	45,5%	21,40%	65,95%	40,15%	49,91%
Brilho	1280 e 1024	35,4%	18,45%	59,18%	33,43%	42,73%
Ruídos	1280 e 1024	20,8%	10,06%	41,26%	22,17%	28,84%
Original (Bounding Boxes)	1280 e 1024	15,9%	8,78%	34,30%	18,44%	23,99%
Simultânea	1280 e 1024	13,1%	6,54%	10,74%	20,09%	14,00%
Original (Polígonos)	1280 e 1024	7,44%	2,66%	1,68%	12,35%	2,96%
YOLOv5s (COCO)	640	56,8%	37,4%	-	-	-
YOLOv5s6 (COCO)	1280	63,7%	44,8%	-	-	-
YOLOv5x (COCO)	640	68,9%	50,7%	-	-	-
YOLOv5x6 (COCO)	1280	72,7%	55,0%	-	-	-

5. CONCLUSÃO E TRABALHOS FUTUROS

Este trabalho avaliou os comportamentos do YOLOv5 na base de defeitos têxteis AITEX, aplicando técnicas de anotação de objetos e *data augmentation*. A base original com 103 amostras de 12 tipos distintos de defeitos, contava com apenas 66 imagens para a etapa de treinamento. Com *data augmentation*, a base chegou a 802 amostras de treino, o suficiente para alcançar resultados significativos, mesmo comparados com os resultados dos modelos pré-treinados na base original COCO, contendo mais de 117 mil amostras.

A anotação de objetos com *Bounding Boxes* obteve resultados consideravelmente melhores, mesmo sem otimizações na base original. Além disso, com a utilização de técnicas isoladas de *data augmentation* o modelo obteve resultados individuais superiores ao *baseline*, sendo potencializados após a junção de todas as amostras obtidas com cada técnica, incluindo as que obtiveram resultados isolados inferiores. As novas amostras aumentaram o poder de generalização do modelo, destacando as amostras com todas as técnicas combinadas, que incrementaram um pouco mais os resultados. Esses comportamentos comprovam que o YOLOv5 pode se adaptar em qualquer contexto, mesmo com objetos distintos da sua base original e permite que o modelo seja executado direta-

⁶Para os modelos YOLOv5, as métricas *precision* e *recall* não foram fornecidas na página oficial.

mente em imagens sem pré-processamento como cortes e fragmentos o que seria vantajoso quando aplicado por exemplo, em um ambiente com um sistema de vídeo monitoramento que captura as imagens em tempo real.

Em trabalhos futuros serão avaliadas novas abordagens de função de regressão de objetos neste contexto, como a SIOU, proposta recentemente por [Gevorgyan 2022] e ainda não disponível na versão atual do YOLOv5. Além disso, assim como as versões pré-treinadas com a base COCO, a partir da base de dados AITEX serão criados novos pontos iniciais (âncoras) do modelo, especializados no contexto de detecção de defeitos têxteis. O novo modelo pré-treinado para os objetos desse contexto poderá ser utilizado e validado em outras bases de dados públicas ou privadas.

6. Agradecimentos

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 Estratégicas nos Estados). A profª Komati agradece ao CNPq pela Bolsa de Produtividade DT-2 (308432/2020-7) e à FAPES pelo Auxílio Taxa de Pesquisa (nº 293/2021).

Referências

- Arlen, T. C. (2018). Understanding the map evaluation metric for object detection. <https://medium.com/@timothycarlen/understanding-the-map-evaluation-metric-for-object-detection-a07fe6962cf3>.
- Bochkovskiy, A., Wang, C.-Y., and Liao, H.-Y. M. (2020). Yolov4: Optimal speed and accuracy of object detection. *arXiv preprint arXiv:2004.10934*.
- Conceição, A. M. G. (1998). Critérios de classificação de tecidos quanto à qualidade. http://www.textilia.net/materias/ler/textil/maquina-textil/computacao_visual_no_gerenciaimento_da_qualidade.
- Gevorgyan, Z. (2022). Siou loss: More powerful learning for bounding box regression. *arXiv preprint arXiv:2205.12740*.
- Jin, R. and Niu, Q. (2021). Automatic fabric defect detection based on an improved yolov5. *Mathematical Problems in Engineering*, 2021.
- Jing, J., Wang, Z., Rättsch, M., and Zhang, H. (2022). Mobile-unet: An efficient convolutional neural network for fabric defect detection. *Textile Research Journal*, 92(1-2):30–42.
- Jing, J., Zhuo, D., Zhang, H., Liang, Y., and Zheng, M. (2020). Fabric defect detection using the improved yolov3 model. *Journal of engineered fibers and fabrics*, 15:1558925020908268.
- Jocher, G., Nishimura, K., Mineeva, T., and Vilariño, R. (2020). Yolov5 (2020). *GitHub repository: <https://github.com/ultralytics/yolov5>*.
- Jun, X., Wang, J., Zhou, J., Meng, S., Pan, R., and Gao, W. (2021). Fabric defect detection based on a deep convolutional neural network using a two-stage strategy. *Textile Research Journal*, 91(1-2):130–142.

- Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., and Zitnick, C. L. (2014). Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer.
- Ouyang, W., Xu, B., Hou, J., and Yuan, X. (2019). Fabric defect detection using activation layer embedded convolutional neural network. *IEEE Access*, 7:70130–70140.
- Petronas, I. I. (2020). Indústria têxtil 4.0: quais são as novidades para este setor? <https://inovacaoindustrial.com.br/industria-textil-40/>.
- Popkova, E. G., Ragulina, Y. V., and Bogoviz, A. V. (2019). Fundamental differences of transition to industry 4.0 from previous industrial revolutions. In *Industry 4.0: Industrial Revolution of the 21st Century*, pages 21–29. Springer.
- Redmon, J., Divvala, S., Girshick, R., and Farhadi, A. (2015). You look only once: unified real-time object detection. *arXiv preprint arXiv:1506.02640*.
- Rezatofighi, H., Tsoi, N., Gwak, J., Sadeghian, A., Reid, I., and Savarese, S. (2019). Generalized intersection over union: A metric and a loss for bounding box regression. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 658–666.
- Rong-qiang, L., Ming-hui, L., Jia-chen, S., and Yi-bin, L. (2021). Fabric defect detection method based on improved u-net. In *Journal of Physics: Conference Series*, volume 1948, page 012160. IOP Publishing.
- Sakkos, D., Shum, H. P., and Ho, E. S. (2019). Illumination-based data augmentation for robust background subtraction. In *2019 13th International Conference on Software, Knowledge, Information Management and Applications (SKIMA)*, pages 1–8. IEEE.
- Seçkin, A. Ç. and Seçkin, M. (2022). Detection of fabric defects with intertwined frame vector feature extraction. *Alexandria Engineering Journal*, 61(4):2887–2898.
- Silvestre-Blanes, J., Albero Albero, T., Miralles, I., Pérez-Llorens, R., and Moreno, J. (2019). A public fabric database for defect detection methods and results. *Autex Research Journal*, 19(4):363–374.
- Wang, Y., Hao, Z., Zuo, F., and Pan, S. (2021). A fabric defect detection system based improved yolov5 detector. In *Journal of Physics: Conference Series*, volume 2010, page 012191. IOP Publishing.
- Wieler, M., Hahn, T., and Hamprecht, F. A. (2007). Weakly supervised learning for industrial optical inspection. [dataset]. <https://hci.iwr.uni-heidelberg.de/content/weakly-supervised-learning-industrial-optical-inspection>.
- Zheng, L., Wang, X., Wang, Q., Wang, S., and Liu, X. (2021). A fabric defect detection method based on improved yolov5. In *2021 7th International Conference on Computer and Communications (ICCC)*, pages 620–624. IEEE.
- Zheng, Z., Wang, P., Liu, W., Li, J., Ye, R., and Ren, D. (2020). Distance-iou loss: Faster and better learning for bounding box regression. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 12993–13000.
- Zhou, Q., Mei, J., Zhang, Q., Wang, S., and Chen, G. (2021). Semi-supervised fabric defect detection based on image reconstruction and density estimation. *Textile Research Journal*, 91(9-10):962–972.