UNIVERSIDADE FEDERAL DE SANTA CATARINA
CENTRO TECNOLÓGICO
CURSO DE GRADUAÇÃO EM ENGENHARIA ELÉTRICA

Vítor Dias Javornik

Aprendizado Profundo na Viticultura: Técnicas de Segmentação Semântica para Imagens de Uvas

Florianópolis
2024

Ficha catalográfica gerada por meio de sistema automatizado gerenciado pela BU/UFSC.
Dados inseridos pelo próprio autor.

Javornik, Vitor Dias
Aprendizado Profundo na Viticultura: Técnicas de Segmentação Semântica para Imagens de Uvas / Vitor Dias Javornik ; orientador, Jônata Tyska Carvalho, coorientador, Thiago Teixeira Santos, 2024.
77 p.

Trabalho de Conclusão de Curso (graduação) - Universidade Federal de Santa Catarina, Centro Tecnológico, Graduação em Engenharia Elétrica, Florianópolis, 2024.

Inclui referências.

1. Engenharia Elétrica. 2. Visão Computacional. 3. Aprendizado Profundo. 4. Segmentação de Uvas. 5. Anotação de Imagens. I. Carvalho, Jônata Tyska. II. Santos, Thiago Teixeira. III. Universidade Federal de Santa Catarina. Graduação em Engenharia Elétrica. IV. Título.

Vítor Dias Javornik

Aprendizado Profundo na Viticultura: Técnicas de Segmentação Semântica para Imagens de Uvas

Trabalho de Conclusão de Curso submetido à Universidade Federal de Santa Catarina, como requisito necessário para obtenção do grau de Bacharel em Engenharia Elétrica

Florianópolis, julho de 2024

Vítor Dias Javornik

Aprendizado Profundo na Viticultura: Técnicas de Segmentação Semântica para Imagens de uvas

Este Trabalho Conclusão de Curso foi julgado adequado para obtenção do Título de "Bacharel em Engenharia Elétrica" e aceito, em sua forma final, pelo Curso de Graduação em Engenharia Elétrica.

Florianópolis, 09 de julho de 2024.

Documento assinado digitalmente
Miguel Moreto
Data: 15/07/2024 11:03:55-0300
CPF: ***.850.100-**
Verifique as assinaturas em https://v.ufsc.br

Prof. Miguel Moreto, Dr.
Coordenador do Curso de Graduação em Engenharia Elétrica

Banca Examinadora:

Documento assinado digitalmente
Jonata Tyska Carvalho
Data: 09/07/2024 16:21:39-0300
CPF: ***.301.250-**
Verifique as assinaturas em https://v.ufsc.br

Prof. Jônata Tyska Carvalho, PhD.
Orientador
Universidade Federal de Santa Catarina

Documento assinado digitalmente
Cesar Ramos Rodrigues
Data: 09/07/2024 16:25:58-0300
CPF: ***.557.990-**
Verifique as assinaturas em https://v.ufsc.br

Cesar Ramos Rodrigues, Dr.
Universidade Federal de Santa Catarina

Documento assinado digitalmente
Danilo Silva
Data: 09/07/2024 17:52:55-0300
CPF: ***.557.924-**
Verifique as assinaturas em https://v.ufsc.br

Prof. Danilo Silva, PhD.
Universidade Federal de Santa Catarina

Documento assinado digitalmente
JOCELI MAYER
Data: 09/07/2024 16:27:54-0300
CPF: ***.833.519-**
Verifique as assinaturas em https://v.ufsc.br

Prof. Joceli Mayer, PhD.
Universidade Federal de Santa Catarina

Este trabalho é dedicado a Mariluci, Volmir e Rodrigo pelo seu apoio incondicional.

AGRADECIMENTOS

Agradeço a Jônata e Thiago, por sua orientação e por tudo que ensinaram durante o desenvolvimento deste trabalho.

Sou grato a Ana, pela paciência e pela ajuda na anotação das imagens.

Agradeço a todos os colegas do Uruguai que participaram do projeto, sem seu trabalho em campo este trabalho seria impossível.

Também agredeço a todos os colaboradores da UFSC, que apesar das adversidades, continuam lutando pela educação pública e de qualidade.

RESUMO

Grandes avanços foram realizados na área de visão computacional nos últimos anos, especialmente na aplicação de modelos baseados em *Transformers* para tarefas de segmentação de imagens. Isto permitiu que diversas áreas do conhecimento se beneficiassem destes modelos, uma delas é a agricultura de precisão, na detecção de doenças, estimativa de safras e em outras diversas aplicações. Este trabalho avalia de forma holística a segmentação de imagens, desde a anotação até o treinamento de modelos de aprendizado profundo, com o objetivo de identificar as melhores práticas e ferramentas disponíveis para a segmentação de uvas em imagens de vinhedos. Foram avaliados quatro modelos de segmentação interativa de imagens visando encontrar métodos eficientes para anotar imagens em larga escala. Entre os modelos avaliados, destacaram-se o RITM e o ClickSEG, que apresentaram resultados promissores, permitindo a anotação de imagens sem a necessidade de ajustes manuais extensivos ou alto poder computacional. Além disso, quatro modelos baseados em *Transformers* foram treinados e avaliados na tarefa de segmentação semântica de uvas: SegFormer, MaskFormer, Mask2Former e OneFormer. Uma *Focal Loss* foi aplicada ao SegFormer para melhorar seu desempenho, apesar de não mostrar-se o melhor modelo, sua performance foi expressivamente melhorada. Os resultados gerais foram próximos entre os modelos, com o Mask2Former se destacando, alcançando um F1-Score de 88,56% e um mIoU de 79,87% em um *dataset* diurno. As técnicas de anotação e segmentação semântica desenvolvidas com o *dataset* diurno foram aplicadas a um *dataset* noturno, obtendo resultados satisfatórios, com um F1-Score de 87,03% e um mIoU de 77,53%, demonstrando a robustez dos modelos treinados. Embora este trabalho seja focado na viticultura, as técnicas e conceitos desenvolvidos podem ser facilmente aplicados a outros problemas de segmentação de imagens, inclusive fora do contexto da agricultura de precisão.

Palavras-chave: Visão Computacional, Aprendizado Profundo, Segmentação de Uvas, Anotação de Imagens, *Transformers*.

ABSTRACT

Significant advances have been made in the field of computer vision in recent years, especially in the application of Transformer-based models for image segmentation tasks. This has allowed various fields of knowledge to benefit from these models, one of which is precision agriculture, in the detection of diseases, yield estimation, and other diverse applications. This work holistically evaluates image segmentation, from annotation to the training of deep learning models, aiming to identify the best practices and tools available for grape segmentation in vineyard images. Four interactive image segmentation models were evaluated to find efficient methods for large-scale image annotation. Among the evaluated models, RITM and ClickSEG stood out, showing promising results, allowing image annotation without the need for extensive manual adjustments or high computational power. Additionally, four Transformer-based models were trained and evaluated on the task of semantic segmentation of grapes: SegFormer, MaskFormer, Mask2Former, and OneFormer. A Focal Loss was applied to SegFormer to improve its performance; although it did not prove to be the best model, its performance was significantly improved. The overall results were close among the models, with Mask2Former standing out, achieving an F1-Score of 88.56% and an mIoU of 79.87% on a daytime dataset. The annotation and semantic segmentation techniques developed with the daytime dataset were applied to a nighttime dataset, obtaining satisfactory results, with an F1-Score of 87.03% and an mIoU of 77.53%, demonstrating the robustness of the trained models. Although this work is focused on viticulture, the techniques and concepts developed can be easily applied to other image segmentation problems, even outside the context of precision agriculture.

Keywords: Computer Vision, Deep Learning, Grape Segmentation, Image Annotation, Transformers.

LISTA DE FIGURAS

LISTA DE TABELAS

SUMÁRIO

1 INTRODUÇÃO

Segundo SANTOS; BARBEDO *et al.* (2023), a visão computacional é um campo da inteligência artificial dedicado a extrair informações de imagens digitais, podendo ser aplicada na agricultura para detecção de doenças em plantas, na estimativa e na avaliação não invasiva de atributos de safras.

Vitis vinifera, também conhecida como parreira, é a espécie de videira mais comum e economicamente importante no mundo. Esta espécie é responsável pela grande maioria das variedades de uvas utilizadas para a produção de vinhos, uvas de mesa e uvas passas.

Este Trabalho de Conclusão de Curso pode ser tratado como um módulo de um projeto maior. Intitulado *Incorporación de herramientas de inteligencia artificial y visión computacional para la predicción del desempeño en Vitis vinifera cv Tannat* (Incorporação de ferramentas de inteligência artificial e visão computacional para a predição de rendimento em *Vitis vinifera cv Tannat*), foi financiado pela *International Development Research Centre* (IDRC) do Canadá e selecionado em uma convocatória organizada pela *Agencia Nacional de Investigación e Innovación* (ANII) do Uruguai. O projeto tem como objetivos principais a criação de conjuntos de dados com inflorescências e cachos de uvas *Tannat*, a implementação de um sistema de visão computacional para predição de colheita e a criação de uma rede de pesquisadores das áreas da biologia vegetal e inteligência artificial.

O Uruguai é o maior exportador da *Vitis vinifera cv Tannat* do mundo (DUARTE ALONSO, 2016). Os vinhos *Tannat* se caracterizam por um alto potencial enológico, com uma concentração típica de taninos, antocianos e acidez, o que gera suas particularidades (GONZÁLEZ-NEVES *et al.*, 2004).

A história da viticultura no Uruguai começa com a chegada dos primeiros imigrantes europeus. *Tannat* é a variedade emblemática do Uruguai pelo seu bom comportamento nas condições edafoclimáticas do país, é enfatizada como um símbolo de sua indústria vinícola (DUARTE ALONSO, 2013). Representa 23% da superfície total de vinhedo e 46% das variedades tintas (INAVI, 2023).

Historicamente, a produção da videira até a colheita dos cachos é estimada através de inspeções visuais ou pela amostragem dos cachos. Estes métodos são complicados e pouco precisos, provocando incertezas quanto aos rendimentos e negociação comercial, prejudicando o planejamento da colheita. Uma previsão mais precisa da colheita poderia melhorar o manejo da safra.

Diversos estudos demonstraram a eficácia de modelos de aprendizado profundo para a identificação de cachos de uvas em imagens de parreiras, permitindo a contagem automática de cachos e a estimativa da colheita. A vasta maioria dos métodos de segmentação aplicados são baseados em redes neurais convolucionais, como a *Mask R-CNN* (HE; GKIOXARI *et al.*, 2017).

Transformers (VASWANI *et al.*, 2017) estão se tornando rapidamente uma das arquiteturas de aprendizado profundo mais amplamente aplicadas em várias modalidades, domínios e tarefas (HASSANI; SHI, 2023). Inicialmente desenvolvidas para processamento de linguagem natural, arquiteturas *Transformer* foram empregadas com sucesso no domínio da visão computacional (CARION *et al.*, 2020; DOSOVITSKIY *et al.*, 2021), tornando-se o estado da arte em diversas tarefas.

A anotação de máscaras é trabalhosa e um dos principais gargalos no desenvolvimento de modelos de segmentação (SANTOS; SOUZA *et al.*, 2020; OLENSKYJ *et al.*, 2022). Nos últimos anos, métodos inovadores de anotação interativa para imagens foram publicados (SOFIIUK; PETROV; KONUSHIN, 2021; KIRILLOV; MINTUN *et al.*, 2023), democratizando o acesso a conjuntos de dados anotados de alta qualidade. No entanto, cada método possui características distintas e pode ser mais adequado para diferentes tipos de imagens e tarefas, sendo necessária uma avaliação cuidadosa para determinar o método mais adequado para segmentação de uvas.

Com uma boa arquitetura selecionada e imagens anotadas, é possível treinar um modelo que segmenta os cachos de uma dada imagem, um "classificador de *pixels* de uvas". A etapa seguinte, fora do escopo deste trabalho, consiste em associar os *pixels* de uvas detectados nas imagens aos dados de colheita. Isso permite correlacionar a quantidade de uvas colhidas com a quantidade de *pixels* de uvas nas imagens, proporcionando uma estimativa mais precisa e automatizada da colheita.

Este trabalho, portanto, foca na criação e validação de modelos de visão computacional aplicados à segmentação cachos de uvas *Tannat*, contribuindo para um manejo agrícola mais eficiente e preciso.

1.1 OBJETIVOS

Nas seções abaixo estão descritos o objetivo geral e os objetivos específicos deste TCC.

1.1.1 Objetivo Geral

Este trabalho se concentra no desenvolvimento de ponta a ponta de um sistema de segmentação semântica para identificação de uvas em imagens, aplicando técnicas modernas de aprendizado profundo, com ênfase em modelos *Transformers* e anotação interativa de imagens.

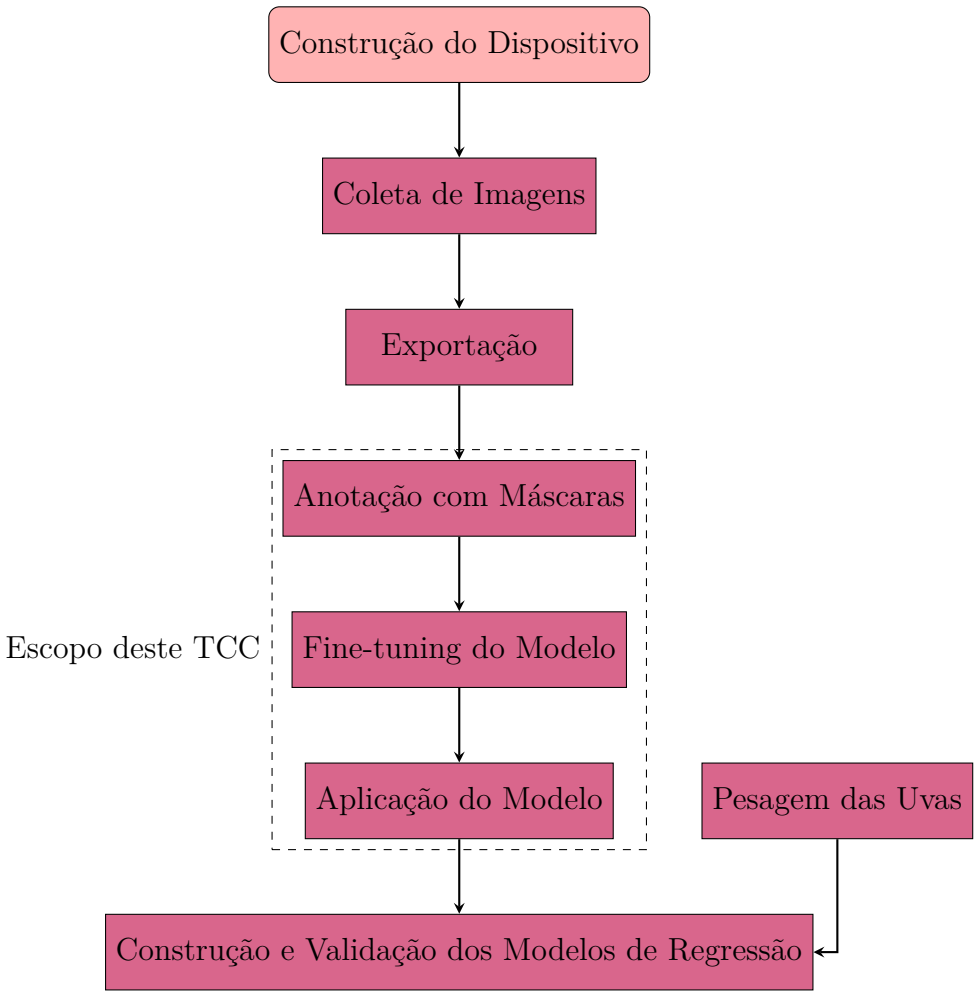
1.1.2 Objetivos Específicos

- Introduzir conceitos e métricas relevantes para a segmentação semântica de imagens;
- Encontrar e comparar modelos interativos para anotação de segmentação em imagens;

- Elaborar um *dataset* (conjunto de dados) de imagens de parreiras com anotações de segmentação semântica de cachos de uvas;
- Realizar o *fine-tuning* (ajuste fino) de modelos pré-treinados de segmentação semântica baseados em arquitetura *Transformer*;
- Avaliar a eficácia dos sistemas desenvolvidos;
- Disponibilizar o código-fonte e os conjuntos de dados utilizados;

O fluxograma abaixo ilustra as etapas básicas envolvidas no sistema na previsão de colheita, destacando o escopo deste Trabalho de Conclusão de Curso:

Figura 1 – Fluxograma das etapas do projeto.



Fonte: Autor.

1. **Construção do Dispositivo de Coleta de Imagens:** Construção de um dispositivo especializado para capturar imagens, fabricado *in-loco* pela célula uruguaia do projeto.

2. **Coleta de Imagens:** Coleta das imagens no vinhedo estudado.
3. **Exportação:** Exportação das imagens coletadas, mantendo controle sobre o seu georeferenciamento.
4. **Anotação com Máscaras de Segmentação:** Um conjunto de imagens imagens é amostrado e as uvas são anotadas com máscaras de segmentação.
5. **Fine-tuning do Modelo de Segmentação Semântica:** *Fine-tuning* de um modelo de segmentação semântica.
6. **Aplicação do Modelo:** Aplicação do modelo treinado às demais imagens coletadas, segmentando cachos de uvas em fotos sem interação humana.
7. **Pesagem das Uvas:** Em paralelo às etapas 4, 5 e 6, a safra é colhida e seu peso associado a pontos georeferenciados.
8. **Construção e Validação dos Modelos de Regressão:** Construção e validação dos modelos de regressão para prever o peso da colheita por setor, utilizando a área de segmentação (número de pixels) como uma feature.

Enquanto a equipe no Uruguai foi responsável pelas atividades em campo e pré-processamento das imagens, a célula brasileira ficou responsável pela seleção e treinamento de modelos de aprendizado profundo.

Os métodos e técnicas adotados por este Trabalho de Conclusão de Curso se adequaram ao cronograma e demandas do projeto principal. O objetivo básico do projeto é a previsão da quantidade de uvas colhidas por fileira de parreiras. O sistema se comportará como um contador de pixels de uvas nas imagens, posteriormente permitindo a correlação entre a quantidade de uvas colhidas e a quantidade de pixels de uvas nas imagens.

1.2 ORGANIZAÇÃO DO TRABALHO

Neste trabalho, inicialmente será realizada uma revisão bibliográfica sobre técnicas de anotação de imagens, segmentação de imagens com modelos *Transformers* e de trabalhos com visão computacional aplicada a viticultura.

Posteriormente, será apresentada de ponta a ponta a metodologia utilizada para o desenvolvimento do sistema de segmentação de uvas, incluindo a descrição/elaboração dos *datasets*, modelos utilizados e configurações aplicadas.

Os resultados obtidos serão discutidos e serão seguidos de uma conclusão e sugestões para trabalhos futuros.

2 FUNDAMENTAÇÃO TEÓRICA

Neste capítulo, conceitos relevantes para o entendimento do trabalho são apresentados, como a tarefa de segmentação de imagens, métricas de avaliação, funções de perda, métricas de erro, conceitos de treinamento de modelos.

2.1 SEGMENTAÇÃO DE IMAGENS

A segmentação de imagens é uma área crítica dentro do campo da visão computacional, cujo objetivo é dividir uma imagem em partes ou regiões significativas, geralmente para facilitar a análise subsequente ou para melhorar a eficiência de outros processos de visão computacional.

Embora este trabalho esteja concentrado na segmentação semântica, é relevante entender quais outra opções estão disponíveis e como elas se relacionam. Esta seção explora as principais tarefas segmentação: semântica, por instância e panóptica.

2.1.1 Segmentação Semântica

A segmentação semântica busca particionar uma imagem em regiões que correspondem a classes de objetos predefinidas. Cada pixel da imagem é classificado em uma das categorias, onde todos os pixels que pertencem à mesma categoria são tratados de maneira uniforme. Este tipo de segmentação não distingue entre diferentes objetos da mesma classe, tratando-os como um único grupo homogêneo. Por exemplo, em uma cena que contenha várias uvas, a segmentação semântica irá rotular todos os pixels de uva de forma indistinta, sem diferenciar entre cada uva individual.

2.1.2 Segmentação por Instância

Diferentemente da segmentação semântica, a segmentação por instância não só categoriza os pixels com base em classes de objetos, mas também diferencia cada instância de objeto dentro da mesma classe. Assim, cada objeto individual é identificado e delimitado, permitindo uma análise mais detalhada. No contexto de segmentação de uvas, este método permitiria identificar e contar cada uva separadamente, mesmo que estejam agrupadas ou sobrepostas na imagem.

2.1.3 Segmentação Panóptica

A segmentação panóptica foi introduzida em *Panoptic Segmentation* (KIRILLOV; HE *et al.*, 2019), ela combina os conceitos de segmentação semântica e por instância, oferecendo uma compreensão holística da cena ao segmentar e identificar todas as instâncias de objetos (*Things*) e categorizar elementos de fundo amorfos (*Stuff*), como céu, estradas

ou grama. Este tipo de segmentação é desafiador pois requer a integração eficaz de dois tipos de segmentação em uma única abordagem coerente.

2.1.4 Desafios da Segmentação de Imagens

A segmentação de imagens enfrenta diversos desafios, principalmente relacionados à variabilidade das formas, tamanhos e contextos em que os objetos podem aparecer. Além disso, condições de iluminação, oclusão entre objetos e a própria qualidade das imagens podem afetar significativamente o desempenho dos algoritmos de segmentação.

Estes métodos e desafios formam a base para a aplicação prática de técnicas de segmentação em diversos campos.

Neste trabalho, dado o objetivo posterior de prever a quantidade colhidas por fileira de parreiras, trabalhar-se-á com segmentação semântica, focando na identificação de uvas em imagens de parreiras e tratando os outros elementos da imagem como fundo.

Embora a segmentação de instâncias de uvas forneça mais informações do que a segmentação semântica sobre a imagem, uvas *Tannat* possuem cachos grandes e compactos (ARRILLAGA *et al.*, 2021), que se sobrepõem e tornam a segmentação de instâncias um desafio ainda maior. Tratar o problema como uma tarefa de segmentação semântica simplifica o processo de anotação e treinamento, permitindo maior precisão nas estimativas enquanto ainda fornece atributos valiosos para a estimativa da colheita.

2.2 MÉTRICAS DE AVALIAÇÃO EM SEGMENTAÇÃO DE IMAGENS

A avaliação de algoritmos de segmentação de imagens é crucial para determinar a eficácia das técnicas utilizadas. Diversas métricas são empregadas para medir o desempenho desses algoritmos, incluindo *Precision* (precisão), *Recall* (sensibilidade), *F1-Score*, e *Intersection over Union* (IoU, Interseção sobre União). Esta seção descreve cada uma dessas métricas e discute sua relevância no contexto da segmentação de imagens.

2.2.1 Precision, Recall e F1-Score

Precision e *Recall* são métricas comumente usadas para avaliar a qualidade dos modelos de segmentação. *Precision* mede a proporção de predições corretas entre as identificações feitas pelo modelo, enquanto *Recall* avalia quantos dos casos relevantes reais foram capturados pelo modelo.

As equações para *Precision* (P) e *Recall* (R) são as seguintes:

$$P = \frac{\text{Verdadeiros Positivos}}{\text{Verdadeiros Positivos} + \text{Falsos Positivos}} \tag{1}$$

$$R = \frac{\text{Verdadeiros Positivos}}{\text{Verdadeiros Positivos} + \text{Falsos Negativos}} \tag{2}$$

O *F1-Score* é o meio harmônico de *Precision* e *Recall*, oferecendo um balanço entre ambas as métricas, especialmente útil quando a distribuição de classes é desigual. É calculado como:

$$F1 = 2 \cdot \frac{P \cdot R}{P + R} \tag{3}$$

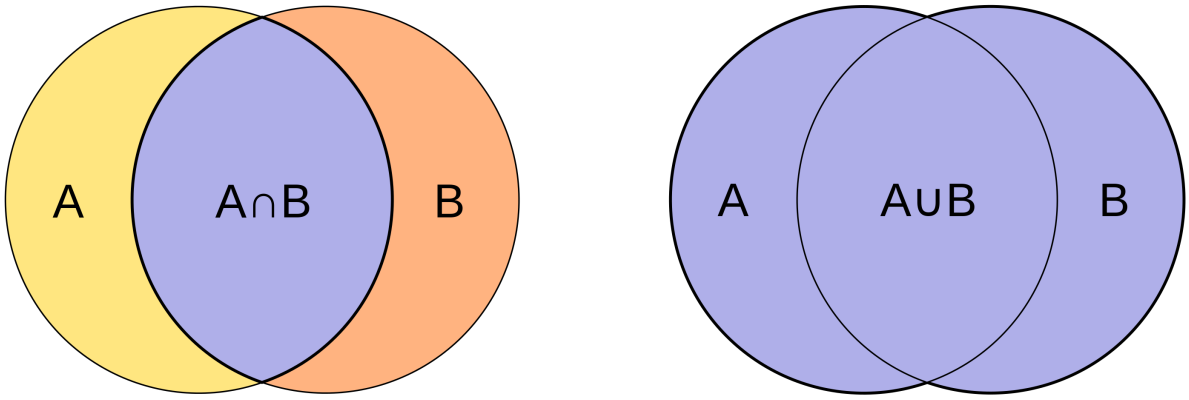
2.2.2 Intersection over Union (IoU)

A *Intersection over Union* (IoU), também conhecida como *Jaccard Index*, é particularmente popular na avaliação de decteção e segmentação de imagens. IoU mede a sobreposição entre a área predita pelo modelo e a área real, dividida pela união destas áreas. A equação para IoU dada por:

$$IoU = \frac{\text{Área de Interseção}}{\text{Área de União}} \tag{4}$$

Essencialmente, se uma máscara for muito maior ou menor do que a máscara real, a IoU será baixa, mesmo que a máscara esteja corretamente posicionada e vice-versa. Caso a máscara possua um tamanho semelhante ao da anotação mas esteja mal posicionada, sua área de união será aumentada e a IoU reduzida.

Figura 2 – Comparação das operações de interseção e união de conjuntos A e B.



(a) Interseção (b) União

Fonte: (COMMONS, 2023).

2.3 FUNÇÕES DE PERDA EM MODELOS DE SEGMENTAÇÃO

As funções de perda são cruciais para treinar modelos de segmentação de imagens, pois guiam o processo de ajuste dos parâmetros do modelo. Para maior compreensão deste trabalho, é importante considerar as definições das funções *Cross-Entropy Loss* (Perda de Entropia Cruzada) e *Focal Loss* (Perda Focal) (LIN *et al.*, 2017).

2.3.1 Cross-Entropy Loss

A *Cross-Entropy Loss* é amplamente utilizada em problemas de classificação, incluindo a segmentação semântica. Ela mede a discrepância entre a distribuição de probabilidade predita pelo modelo e a distribuição real das classes.

Em termos simples, esta função penaliza mais fortemente as previsões incorretas, encorajando o modelo a ajustar seus parâmetros para melhorar a precisão das previsões.

A fórmula da *Cross-Entropy Loss* é dada por:

$$L_{\text{CE}} = -\sum_{i=1}^{C} y_i \log(p_i)$$

onde C é o número de classes, y_i é o valor real (0 ou 1) para a classe i e p_i é a probabilidade predita para a classe i.

A *Cross-Entropy Loss* é eficaz porque se concentra em reduzir a incerteza nas previsões do modelo, ajustando as probabilidades preditas para se alinharem mais estreitamente com os valores reais. Esta função de perda é especialmente útil em problemas de segmentação semântica, onde a precisão de classe individual é crítica.

2.3.2 Focal Loss

A *Focal Loss* é uma variação da *Cross-Entropy Loss* projetada para lidar com desequilíbrios de classes, um problema comum em tarefas de segmentação onde algumas classes podem ser significativamente mais prevalentes que outras. Esta função de perda atribui mais peso às amostras de difícil classificação, reduzindo a contribuição de amostras que já são classificadas corretamente pelo modelo.

$$L_{\text{Focal}} = -\sum_{i=1}^{C} (1 - p_i)^{\gamma} y_i \log(p_i)$$

onde γ é um fator de ajuste que controla o quanto de peso é dado às amostras mal classificadas.

Esta função é particularmente útil em cenários onde há um grande desequilíbrio entre as classes, por exemplo na diferenciação entre cachos de uvas e fundo, permitindo que o modelo preste mais atenção às classes minoritárias ou às amostras mais difíceis. A *Focal Loss* ajuda a mitigar o problema de que as amostras majoritárias dominem o processo de treinamento, o que levaria a um modelo que simplesmente ignora as classes minoritárias.

2.4 MÉTRICAS DE ERRO

As métricas de erro são essenciais para avaliar a performance de modelos preditivos. Elas permitem quantificar a discrepância entre os valores preditos pelo modelo e os valores

reais, fornecendo uma base objetiva para comparação e melhoria dos modelos. Entre as mais comuns estão o Erro Quadrático Médio (*Mean Squared Error - MSE*), a Raiz do Erro Quadrático Médio (*Root Mean Squared Error - RMSE*), o Erro Absoluto Médio (*Mean Absolute Error - MAE*) e o Erro Percentual Absoluto Médio (*Mean Absolute Percentage Error - MAPE*).

Estas métricas são cruciais para avaliar a precisão dos modelos preditivos e identificar áreas onde podem ser melhorados.

2.4.1 Mean Squared Error (MSE)

O *MSE* é calculado como a média dos quadrados das diferenças entre os valores preditos e os valores reais. Ele penaliza mais severamente grandes erros devido ao uso do quadrado, o que o torna útil para identificar grandes desvios. A fórmula é dada por:

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2$$

onde y_i são os valores reais e $\hat{y}_i$ são os valores preditos.

2.4.2 Root Mean Squared Error (RMSE)

O *RMSE* é a raiz quadrada do *MSE* e fornece uma medida do erro nas mesmas unidades da variável de interesse. Isso facilita a interpretação dos resultados, especialmente quando os dados têm uma escala compreensível. A fórmula é:

$$RMSE = \sqrt{MSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}$$

2.4.3 Mean Absolute Error (MAE)

O *MAE* é a média das diferenças absolutas entre os valores preditos e os valores reais. Ele é menos sensível a outliers em comparação ao *MSE*, tornando-o uma métrica robusta para modelos onde a presença de grandes erros isolados não é desejável. A fórmula é:

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i|$$

2.4.4 Mean Absolute Percentage Error (MAPE)

O *MAPE* calcula a média das diferenças absolutas em termos percentuais entre os valores preditos e os valores reais. Esta métrica é particularmente útil quando se deseja avaliar o desempenho do modelo em termos relativos, permitindo a comparação entre diferentes escalas de dados. A fórmula é:

$$MAPE = \frac{1}{n} \sum_{i=1}^{n} \left| \frac{y_i - \hat{y}_i}{y_i} \right| \times 100$$

onde y_i são os valores reais e $\hat{y}_i$ são os valores preditos.

O *MAPE* é amplamente utilizado em finanças e economia, onde a interpretação percentual facilita a compreensão dos erros em relação aos valores reais.

2.5 REDES NEURAIS E APRENDIZADO PROFUNDO

As redes neurais artificiais são modelos matemáticos inspirados no funcionamento do cérebro humano, capazes de aprender padrões complexos a partir de dados. O componente fundamental é o *Perceptron*, um modelo simples que forma a base para redes mais complexas, como o *Multilayer Perceptron* (MLP), utilizado em muitas aplicações de aprendizado profundo.

O treinamento de modelos de aprendizado profundo envolve a preparação de dados, escolha de hiperparâmetros e monitoramento do desempenho do modelo. Alguns conceitos úteis no contexto deste trabalho incluem o tamanho do lote (*batch size*), otimizadores, conjuntos de dados e agendadores de aprendizado (*schedulers*).

2.5.1 Otimizadores

Os otimizadores são algoritmos que ajustam os pesos do modelo para minimizar a função de perda. O *AdamW* (LOSHCHILOV; HUTTER, 2017) é uma variação do otimizador *Adam* (KINGMA; BA, 2017) que inclui uma penalização de peso para evitar o *overfitting*.

Ele é definido pela fórmula:

$$\theta_t = \theta_{t-1} - \eta \left(\frac{\hat{m}_t}{\sqrt{\hat{v}_t} + \epsilon} + \lambda \theta_{t-1} \right)$$

onde θ_t são os parâmetros no passo t, η é a taxa de aprendizado, $\hat{m}_t$ e $\hat{v}_t$ são as médias móveis do primeiro e segundo momentos do gradiente, ϵ é um pequeno número para evitar divisão por zero e λ é o fator de penalização de peso.

2.5.2 Scheduler

Os agendadores (*schedulers*) ajustam a taxa de aprendizado durante o treinamento. O *poly scheduler*, por exemplo, diminui a taxa de aprendizado de acordo com uma potência do número de iterações de treinamento:

$$\eta_t = \eta_0 \left(1 - \frac{t}{T} \right)^p$$

onde η_t é a taxa de aprendizado no passo t, η_0 é a taxa inicial, T é o número total de passos de treinamento, e p é o fator de potência.

2.5.3 Blocos Convolucionais, de Atenção e MLP

Modelos de aprendizado profundo frequentemente utilizam diversos tipos de blocos para capturar diferentes características dos dados. Os blocos *MLP*, convolucionais e de atenção são bastante relevantes no contexto de visão computacional.

2.5.3.1 Blocos MLP

Os blocos *MLP* (*Multilayer Perceptron*) são compostos por camadas totalmente conectadas. Cada camada aplica uma transformação linear seguida de uma função de ativação não linear, permitindo a modelagem de relações complexas nos dados. A transformação linear é dada por:

$$y = Wx + b$$

onde W é a matriz de pesos, x é a entrada, e b é o vetor de *bias*. A função de ativação, como $ReLU$ ou *sigmoid*, introduz não-linearidade no modelo, aumentando sua capacidade de aprendizado.

A utilidade dos blocos *MLP* está na sua capacidade de funcionar como uma camada densa que pode aprender representações complexas a partir de dados tabulares, de séries temporais e até mesmo complementar redes convolucionais em tarefas de visão computacional.

2.5.3.2 Blocos Convolucionais

Blocos convolucionais são a base das redes neurais convolucionais (*CNNs*), especialmente eficazes em tarefas de processamento de imagem. Eles consistem em filtros que convoluem a entrada para extrair características locais, tais como bordas, texturas e formas.

A operação de convolução é definida por:

$$(f * g)(t) = \sum_{\tau} f(\tau) g(t - \tau)$$

onde f é a função de entrada e g é o filtro.

A utilidade dos blocos convolucionais está na sua capacidade de capturar características espaciais e hierárquicas dos dados, sendo extremamente eficazes em tarefas de reconhecimento de imagem, detecção de objetos e segmentação semântica. Eles permitem a extração automática de características relevantes dos dados de entrada, reduzindo a necessidade de engenharia manual de atributos.

2.5.3.3 Blocos de Atenção

Blocos de atenção são usados em modelos como *Transformers*, permitindo que o modelo foque em diferentes partes da entrada ao processar cada elemento. Essa capacidade é particularmente útil em tarefas de processamento de linguagem natural e tradução automática. A atenção é calculada como:

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V$$

onde Q é a matriz de consultas, K é a matriz de chaves, V é a matriz de valores e d_k é a dimensão das chaves.

A utilidade dos blocos de atenção reside na sua habilidade de modelar dependências longas e complexas entre elementos da sequência de entrada, superando limitações de arquiteturas tradicionais como *RNNs* e *LSTMs*. Isso os torna extremamente eficazes em tarefas como tradução automática, resumo de texto, e qualquer aplicação que requeira a compreensão de contextos extensos dentro dos dados.

3 REVISÃO DO ESTADO DA ARTE

Neste capítulo, será realizada uma revisão de trabalhos relacionados a segmentação interativa, em seguida serão abordados estudos sobre segmentação semântica e, por fim, serão apresentados estudos de visão computacional aplicados à viticultura.

3.1 MODELOS DE SEGMENTAÇÃO INTERATIVA

Tradicionalmente, máscaras de segmentação foram desenhadas manualmente, pixel a pixel, a partir de *brush tools* (ferramentas de pincel) ou contornando objetos de interesse com polígonos. Este processo é demorado e propenso a erros, especialmente em imagens com muitos objetos ou com bordas difusas, sendo um dos principais gargalos na produção de *datasets* de segmentação.

A segmentação interativa de imagens é uma técnica que combina a eficiência dos modelos de aprendizado de máquina com a avaliação humana, permitindo desenhar máscaras de segmentação de forma mais rápida e precisa. Ao contrário dos modelos de segmentação tradicionais (semântica, de instâncias ou panóptica), esses modelos aceitam entradas do usuário durante o processo de segmentação. As formas de interação com o usuário podem incluir cliques, desenhos ou até mesmo prompts de texto.

O feedback do usuário como etapa intermediária muda o paradigma e a avaliação desses modelos. A experiência de usuário, número de interações e tempo de anotação são fatores que devem ser considerados ao avaliar a eficiência de um modelo, além de atributos tradicionais, como a robustez do modelo e a qualidade das máscaras.

Os seguintes estudos e/ou modelos de segmentação interativa foram avaliados:

- **RITM:** Descrito em *Reviving Iterative Training with Mask Guidance for Interactive Segmentation* (SOFIIUK; PETROV; KONUSHIN, 2021).
- **ClickSEG:** A implementação oficial dos métodos propostos nos trabalhos *Conditional Diffusion for Interactive Segmentation* (CHEN *et al.*, 2021) e *FocalClick: Towards Practical Interactive Image Segmentation* (XU *et al.*, 2022).
- **SAM:** Acrônimo de Segment Anything Model, introduzido em *Segment Anything* (KIRILLOV; MINTUN *et al.*, 2023).
- **FastSAM:** Descrito em *Fast Segment Anything* (ZHAO *et al.*, 2023).

3.1.1 RITM

Desenvolvido pela equipe da *Samsung Research*, *Reviving Iterative Training with Mask Guidance for Interactive Segmentation* (RITM) é um artigo que compara diversas abordagens de arquiteturas para segmentação interativa baseada em cliques. Este trabalho representa um marco na área de segmentação interativa.

O estudo baseou-se em métodos anteriores, como o *Backpropagating Refinement Scheme* (BRS) (JANG; KIM, 2019), onde se propôs um processo de otimização onde a cada interação do usuário são executadas passagens *forward* e *backward* no modelo. Este processo garantiu melhor qualidade nas máscaras geradas, mas aumentou significativamente o custo computacional.

Posteriormente, o f-BRS (SOFIIUK; PETROV; BARINOVA *et al.*, 2020), também liderado pela *Samsung Research*, propôs uma versão leve do BRS, reduzindo consideravelmente o tempo de execução ao otimizar parâmetros intermediários em vez das entradas do modelo, requerendo *backward passes* em uma parte bastante reduzida da rede.

RITM propôs uma arquitetura *feedforward* simples, utilizando máscaras de segmentação das interações anteriores, ainda dispondo de suporte nativo ao método f-BRS. Excluindo a necessidade de otimizações a cada clique, o modelo torna-se mais eficiente e rápido, sem perda significativa na qualidade das máscaras.

O estudo também comprovou que a escolha do *dataset* influencia significativamente o desempenho de modelos, com COCO e LVIS apresentado resultados superiores na segmentação, por sua diversidade de classes e máscaras de alta qualidade.

Na época de sua publicação, RITM foi um divisor de águas e tornou-se o estado da arte de ferramentas de anotação interativa. Sua *codebase* e princípio de funcionamento foram a base para o desenvolvimento de diversos outros modelos basedos em cliques.

3.1.2 SAM

Foi produto do projeto *Segment Anything*, desenvolvido pela *Meta AI Research* e lançado em abril de 2023. Seu objetivo foi de criar um modelo fundacional para visão computacional, capaz de segmentar qualquer objeto em uma imagem, sem a necessidade de treinamento auxiliar. Desta maneira, a segmentação interativa é apenas uma das tarefas dentro do escopo do modelo.

Além da elaboração deste novo modelo, *Segment Anything* envolveu a construção de um novo *dataset* e uma nova *task* (tarefa) de segmentação, chamada *Segment Anything Task* ("segmentação de qualquer coisa", em tradução livre).

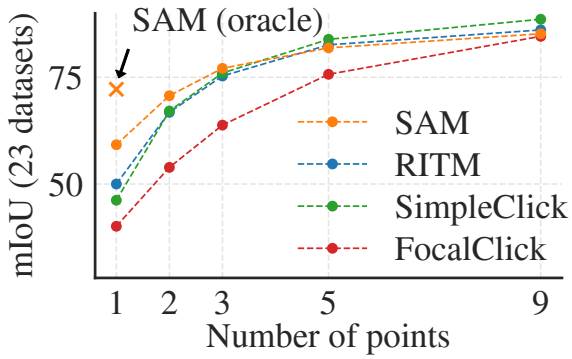
Sua tarefa de treinamento inovadora envolveu a predição de um conjunto de máscaras válidas para qualquer *prompt* fornecido, podendo esse *prompt* ser um ponto, uma caixa, uma máscara, um texto ou mesmo combinações desses. O *dataset* final *SA-1B* foi composto por mais de 1 bilhão de máscaras contidas em 11 milhões de imagens.

Na avaliação do desempenho do modelo para segmentações a partir de pontos (emulando cliques), o SAM foi comparado principalmente com o RITM. Essas comparações foram feitas de diversas maneiras em 23 *datasets*, incluindo a medição do IoU após um clique no centro da máscara *ground truth*, IoU a partir de pontos aleatórios, qualidade da máscara avaliada por humanos, entre outros.

O SAM apresentou resultados superiores na vasta maioria dos casos, principalmente

em segmentações obtidas a partir de um ponto. O RITM foi capaz de se equiparar ao SAM somente quando múltiplos cliques eram fornecidos. Outros modelos construídos a partir da codebase do RITM também foram avaliados, como o *SimpleClick* (LIU, Q. *et al.*, 2023) e o *FocalClick*, sendo que o primeiro apresentando IoU superior e o último resultados bastante inferiores ao RITM.

Figura 3 – IoU médio em função do número de cliques em uma comparação de modelos de segmentação interativa.



Adaptado de (KIRILLOV; MINTUN *et al.*, 2023).

3.1.3 ClickSEG

Criado a partir da *codebase* de RITM, ClickSEG consolida a implementação das técnicas propostas em *Conditional Diffusion for Interactive Segmentation* (CHEN *et al.*, 2021) e *FocalClick: Towards Practical Interactive Image Segmentation* (XU *et al.*, 2022).

O primeiro, contemporâneo ao RITM, apresenta a *Conditional Diffusion Network* (CDNet), uma rede que propaga informações obtidas por cliques do usuário para destinos condicionados e introduz a auto-atenção no domínio da segmentação interativa. Estas regiões são propostas a partir de dois níveis de afinidade: *Feature Diffusion Module* (FDM), que espalha *features* da interação do usuário para regiões-alvo potenciais com base em similaridade global, e *Pixel Diffusion Module* (PDM), que difunde os *logits* previstos dos cliques do usuário dentro de regiões localmente conectadas.

A CDNet apresentou resultados promissores em segmentações a partir de cliques, obtendo resultados superiores ao f-BRS, além de incorporar algumas das descobertas relevantes descritas no RITM.

FocalClick propõe uma abordagem mais prática para a segmentação interativa, visando uma maneira mais eficiente de anotar imagens em dispositivos com recursos limitados. Sua principal inovação consiste em dividir a inferência pesada em duas predições leves.

Primeiro, o *Target Crop* escolhe um pedaço ao redor do objeto alvo, redimensiona-o para uma escala menor e envia-o para o segmentador para predizer uma máscara grosseira. Em seguida, o *Focus Crop* seleciona uma região local que precisa de refinamento e alimenta

o pedaço ampliado no *Refiner*. Finalmente, o *Progressive Merge* alinha as predições locais de volta para as máscaras em escala completa.

Dessa forma, refinam-se apenas pequenas regiões locais após cada clique, distribuindo o cálculo em diferentes rodadas, o que é ideal para dispositivos de baixa potência. Sua abordagem também garante que os detalhes em outras regiões sejam preservados durante as interações.

3.1.4 FastSAM

Embora robusto e inovador em sua proposta, SAM é um modelo pesado, dificultando sua implementação prática. Baseado no YOLOv8-seg (JOCHER; CHAURASIA; QIU, 2023), *Fast Segment Anything* (FastSAM) propõe uma alternativa mais leve e simplificada ao modelo introduzido pela Meta, focando em soluções industriais que requerem segmentação em tempo real.

O modelo foi treinado com apenas 1/50 do *dataset SA-1B* e obteve resultados comparáveis ao SAM, com a vantagem de ser 50 vezes mais rápido. FastSAM pode trabalhar com a mesma variedade de entradas que o SAM, incluindo pontos, caixas, máscaras e texto.

Os pesquisadores dividiram a *Segment Anything Task* em duas etapas: detecção e segmentação de todos os objetos em uma imagem (similar a segmentação panóptica), seguida de uma separação específica das máscaras, baseada no tipo de *prompt* fornecido. Este desacoplamento reduz consideravelmente a complexidade do problema.

Além disso, o modelo é baseado em CNNs, o que reduz significativamente o custo computacional associado a modelos baseados em *Transformers*.

3.1.5 Benchmarking de Modelos de Segmentação Interativa

Segundo os autores de *TETRIS: Towards Exploring the Robustness of Interactive Segmentation* (MOSKALENKO *et al.*, 2024) padrões de cliques humanos em segmentação interativa não foram estabelecidos, sendo baseados em intuição e senso comum dos avaliadores. Neste trabalho, estratégias de avaliação de modelos de segmentação interativa são discutidas.

Conduzido pela *Samsung Labs*, foram analisados padrões de interação de 1800 anotadores reais. Após calcular a distância dos cliques humanos até o centro das áreas de menor erro, metodologia aplicada na maioria dos métodos atuais, percebeu-se que esta estratégia de cliques só é consistente com cliques humanos na segmentação de objetos convexos e de forma simples.

Embora o estudo aponte SimpleClick (LIU, Q. *et al.*, 2023) e CRF-ICL (SUN *et al.*, 2024), ambos inspirados em RITM, como os modelos de melhor desempenho, comparativos entre RITM e SAM foram destaque durante os testes. Em contraste com os

resultados apresentados em *Segment Anything* (KIRILLOV; MINTUN *et al.*, 2023), O SAM apresentou IoU similar ou inferior ao RITM nos *datasets* utilizados.

No trabalho *Interactive segmentation in aerial images: a new benchmark and an open access web-based tool* (WANG, Z. *et al.*, 2024), modelos de segmentação interativa baseada em cliques foram comparados no contexto de anotação de imagens de satélite. Os testes, assim como em *Segment Anything*, mostraram que o SAM atinge um resultado promissor ao trabalhar com poucos cliques. Porém, mesmo ao aumentar o número de cliques, o SAM mostra-se incapaz de aprimorar significativamente o IoU, inviabilizando a anotação de máscaras de alta qualidade.

3.2 MODELOS DE SEGMENTAÇÃO SEMÂNTICA

Nesta seção, serão revisados trabalhos e arquiteturas *Transformers* de visão computacional voltados a segmentação semântica de objetos.

Vision Transformer (ViT) (DOSOVITSKIY *et al.*, 2021) foi um marco na visão computacional, introduzindo uma arquitetura estado da arte baseada em *Transformers* para classificação de imagens. A arquitetura trata imagens como sequências de patches e aplica mecanismos de atenção para capturar relações globais entre eles.

O trabalho reimaginou a forma como imagens são processadas no aprendizado profundo, rompendo com a dependência de convoluções tradicionais e demonstrando desempenho comparável ou superior em *benchmarks* de classificação de imagens. A arquitetura ViT inspirou uma série de *backbones* e arquiteturas subsequentes em tarefas de *dense prediction* (predição densa) em imagens, como a segmentação de objetos.

Antes mesmo do ViT, arquiteturas baseadas em *Transformers* já vinham sendo exploradas em tarefas de visão computacional. Um exemplo é o *DEtection TRansformer* (DETR), proposto pela equipe da *Facebook AI Research* em *End-to-End Object Detection with Transformers* (CARION *et al.*, 2020).

Combinando uma CNN em seu *backbone* com um codificador/decodificador *Transformer*, DETR foi o primeiro modelo a tratar a detecção de objetos como um problema de *set prediction*, onde a ordem dos objetos é irrelevante, eliminando a necessidade de *non-maximum suppression* (NMS) e *anchor boxes*. O trabalho também apresentou a versatilidade da arquitetura ao apresentar que é possível adaptá-la para a tarefa de segmentação panóptica substituindo a cabeça de saída do modelo e obtendo resultados promissores.

SETR (*SEmantic TRansformer*) (ZHENG *et al.*, 2020), também desenvolvido pela *Facebook AI Research*, concentra-se exclusivamente na segmentação semântica. Sua arquitetura utiliza o *backbone* ViT e emprega decodificadores baseados em *CNNs*, atingindo na época o estado da arte no *dataset ADE20K*.

O PVT (Pyramid Vision Transformer) (WANG, W. *et al.*, 2021) introduziu a estrutura piramidal em *Transformers*, onde a entrada é dividida em múltiplas resoluções e processada em paralelo, demonstrando o potencial de um *backbone* puramente baseado

em *Transformers* para tarefas de *dense prediction*. No entanto, sua complexidade ainda é quadrática em relação ao tamanho da imagem.

Outro *backbone* popular alternativo ao ViT é o *Swin Transformer*, descrito em *Swin Transformer: Hierarchical Vision Transformer using Shifted Windows* (LIU, Z. *et al.*, 2021). Este trabalho aplica janelas deslizantes, reduzindo a complexidade computacional associada à auto-atenção dos *Transformers*, permitindo ao modelo processar imagens de diferentes tamanhos e escalas de forma eficiente. Além de melhorar a eficiência, tornando-a linear, também manteve a capacidade de capturar dependências locais dentro das imagens.

Aplicando o *Swin Transformer* como *backbone* em arquiteturas simples pré-estabelecidas, a equipe da *Microsoft Research Asia* superou o SETR no *dataset ADE20K*. Este *backbone* também apresentou resultados robustos na classificação de imagens e detecção de objetos, demostrando ser uma alternativa muito viável para tarefas gerais de visão computacional.

3.2.1 SegFormer

Buscando uma alternativa para segmentação semântica em tempo real, *SegFormer: Simple and Efficient Design for Semantic Segmentation with Transformers* (XIE *et al.*, 2021) revisa tanto as arquiteturas do codificador quanto do decodificador em relação a modelos anteriores que empregam *Transformers*.

Inspirado no sucesso do *Vision Transformer* (ViT), os pesquisadores introduziram os *backbones Mix Transformer* (MiT), que variam do MiT-B0 ao MiT-B5, treinados no *ImageNet*.

Seu codificador *Transformer* hierárquico, ao contrário do ViT que produz um mapa de *features* de uma única resolução, produz *features* de vários níveis. Isto permite capturar informações contextuais em diferentes resoluções, fornecendo representações grosseiras de alta resolução e refinadas de baixa resolução ao decodificador.

Assumindo que codificação posicional não é necessária para a segmentação semântica, foram incluídas convoluções 3×3 seguidas de *multilayer perceptrons* (MLPs) no bloco *Transformer*, rede nomeada como *Mix-FFN* pelos autores. Como o principal gargalo nos codificadores *Transformer* está na auto-atenção, os autores utilizaram um fator de redução R no comprimento da sequência para reduzir a complexidade computacional, reduzindo a complexidade de $O(n^2)$ para $O(n^2)/R$.

Além de concentrar seus esforços na arquitetura do codificador, SegFormer propõe um decodificador simples, puramente baseado MLPs. Sua decodificação é leve, fundindo os mapas de *features* do codificador e produzindo uma segmentação mais precisa.

A ausência de operações convolucionais ou auto-atenção no decodificador contribui significativamente para a eficiência do modelo, especialmente em cenários de segmentação a partir de dispositivos de baixa potência.

3.2.2 MaskFormer

Em *Per-Pixel Classification is Not All You Need for Semantic Segmentation* (CHENG; SCHWING; KIRILLOV, 2021), os autores propõem uma nova abordagem para a segmentação: alterar o paradigma de classificação por *pixels* nas tarefas de segmentação para classificação por máscaras, unificando a segmentação semântica e a de instâncias em uma única arquitetura.

Nessa abordagem, máscaras binárias são geradas para cada objeto na imagem, e a segmentação é tratada como um problema de *set prediction*. A saída do modelo é uma coleção de máscaras de segmentação, cada uma associada a um objeto na imagem. Essa saída corresponde ao formato de máscaras de segmentação por instâncias, sendo que a conversão dessa saída em máscaras de segmentação semântica consiste em um simples produto escalar entre as estimativas de classes e máscaras.

Segundo seus próprios autores, MaskFormer pode ser tratado como uma versão "sem caixas"de DETR, uma vez que poucas alterações foram necessárias. Além de substituir a função de perda baseada em *bounding-boxes* por uma função de perda baseada em máscaras, MaskFormer adota o Swin Transformer como *backbone*. Embora compartilhem o mesmo decodificador *Transformer*, no MaskFormer os *embeddings* do *pixel decoder* são compartilhados entre todas as *queries*, o que não ocorre no DETR e garante N vezes mais eficiência computacional no processo, onde N é o número de *queries*.

Durante sua avaliação, a equipe percebeu que este modelo tem melhor desempenho quando avaliado em *datasets* com maior vocabulário de classes. Como este Trabalho de Conclusão de Curso se concentrará em um problema de classificação binária, onde haverá apenas um objeto e o fundo, essa peculiaridade deve ser destacada. Embora eficiente na segmentação panóptica e tendo se tornado o estado da arte na segmentação semântica época, MaskFormer não foi competitivo quando comparado a arquiteturas especializadas em segmentação de instâncias.

3.2.3 Mask2Former

Mask2Former, introduzido em *Masked-attention Mask Transformer for Universal Image Segmentation* (CHENG; MISRA *et al.*, 2022), adota a mesma metaarquitetura do MaskFormer: um *backbone*, um decodificador de *pixels* e um decodificador *Transformer*.

Partindo da hipótese de que *features* locais são suficientes para atualizar as *query features* e que informações de contexto podem ser obtidas via auto-atenção, a atenção cruzada no decodificador *Transformer* é substituída por uma atenção mascarada, enquanto ordem da auto-atenção e da atenção cruzada (substituída pela atenção mascarada) são invertidas.

Ao invés de alimentar o decodificador *Transformer* apenas com os *feature maps* de alta resolução da saída do *backbone*, trabalha-se com uma pirâmide de *features* contendo

atributos de alta e baixa resolução. Estas *features* multiescalares ajudam o modelo a segmentar pequenos objetos/regiões.

Avançando no paradigma proposto em MaskFormer, Mask2Former tornou-se o estado da arte em segmentação semântica, de instâncias e panóptica em sua época, mostrando que uma arquitetura universal e a unificação destas tarefas poderiam ser um caminho definitivo para a segmentação de imagens.

3.2.4 OneFormer

O trabalho *OneFormer: One Transformer to Rule Universal Image Segmentation* (JAIN *et al.*, 2022) apresenta um *framework* universal para tarefas de segmentação. Embora Mask2Former tenha se consagrado como o estado da arte nas três tarefas de segmentação, para obter o título em cada tarefa foi necessário treinamento específico para cada uma delas, exigindo três vezes mais treinamento, mesmo tendo uma única arquitetura.

Supondo que esta limitação em métodos prévios era causada pela ausência de orientação, os pesquisadores incluíram um *token* de entrada de tarefa, guiando o modelo em cada tarefa.

Como os dados panópticos contêm anotações semânticas e de instâncias, o OneFormer foi idealizado para ser treinado a partir de um único *dataset* panóptico. Com o objetivo de generalizar as três tarefas de segmentação, durante o treinamento cada tarefa foi amostrada uniformemente. Caso a tarefa selecionada fosse a de segmentação semântica ou de instâncias, os rótulos foram derivados a partir da máscara de segmentação panóptica.

Embora OneFormer tenha superado Mask2Former com o *backbone* Swin Transformer, os autores destacam que houve ainda melhor desempenho com o *backbone* DiNAT (HASSANI; SHI, 2023), e portanto foi empregado neste trabalho.

3.3 VISÃO COMPUTACIONAL NA VITICULTURA

Diversos estudos se propuseram a desenvolver conjuntos de dados e sistemas de visão computacional para a análise de imagens de vinhedos, com o objetivo de automatizar a contagem de cachos de uvas e a estimativa de produtividade.

O clássico trabalho *Grape detection, segmentation, and tracking using deep neural networks and three-dimensional association* (SANTOS; SOUZA *et al.*, 2020) se concentrou na detecção, segmentação de instâncias e contagem de cachos uvas de diversas variedades.

O conjunto de dados produzido no estudo, denominado WGISD (Wine Grape Instance Segmentation Dataset), contou com 300 imagens contendo 4432 cachos de uvas anotados manualmente para detecção. 137 dessas imagens foram anotadas para segmentação de instâncias, totalizando 4432 máscaras de segmentação. Este *dataset* consolidou-se para a tarefa de segmentação de cachos de uvas, sendo citado em diversos trabalhos posteriores e inclusive utilizado neste Trabalho de Conclusão de Curso.

Um trabalho que se propôs a comparar diferentes arquiteturas na segmentação de uvas foi *Segmentation and weight prediction of grape ear based on SFNet-ResNet18* (LIANG *et al.*, 2022), confrontando o desempenho três *backbones* aplicados a quatro arquiteturas de segmentação com o objetivo de encontrar um modelo rápido e preciso para a segmentação semântica de cachos de uvas. Segundo os autores, a SFNET (LI *et al.*, 2023) aliada a um *backbone* ResNet-18 obteve os melhores resultados, atigindo um mIoU de 79,45% e uma precisão média de 92,58%.

Neste trabalho foi produzido um *dataset* original de uvas da variedade ZaoHeiBao, contendo 433 imagens de cachos de uvas anotadas com polígonos a nível de instâncias. Com um eficiente modelo de segmentação semântica treinado, os autores então introduziram um modelo de regressão para estimar o peso dos cachos de uvas, obtendo um R^2 de 0,8903.

No trabalho *End-to-End Deep Learning for Directly Estimating Grape Yield from Ground-Based Imagery* (OLENSKYJ *et al.*, 2022), foram testadas três arquiteturas: um modelo YOLO v5 (JOCHER, s.d.) treinado para detecção de cachos de uvas, um modelo de regressão CNN e outro modelo de regressão baseado em *Transformers*. Semelhante ao projeto liderado pela UdelaR, um dispositivo terrestre foi construído para filmar as parreiras. No entanto, a colheita das uvas foi realizada por meio de uma colheitadeira industrial, equipada com um dispositivo GPS para georreferenciamento, permitindo uma pesagem muito mais granular e uma associação espacial direta entre os pontos de colheita e as imagens.

Para a detecção de objetos, 150 imagens foram manualmente anotadas com *bounding boxes* nos cachos de uvas, e os autores associaram tanto a área (*pixels*) quanto a contagem de *bounding boxes* aos pesos em toneladas/ha nos pontos de colheita. No procedimento de regressão com uma CNN, *frames* foram amostrados próximos aos pontos de colheita. A camada final de uma rede ResNet-18 (HE; ZHANG *et al.*, 2016) foi adaptada para representar o peso das uvas colhidas em sua saída.

A arquitetura *Transformer* empregada foi baseada no ViT (DOSOVITSKIY *et al.*, 2021), mas, em vez de *patches* linearizados da imagem, cada imagem passou por um modelo ResNet-18, e o conjunto final de mapas de ativação foi utilizado como *features* de entrada para o codificador. Metadados de posição foram incluídos, e a saída do *Transformer Encoder* foi encaminhada a uma camada MLP para regressão, representando o peso das uvas colhidas.

Os autores obtiveram resultados semelhantes na detecção de objetos seguida da contagem de cachos e na regressão com *Transformers*, com um erro percentual médio absoluto de 18,5% e 18%, respectivamente. Além dos resultados promissores, a arquitetura *Transformer* apresentou a vantagem de não exigir anotações manuais, sendo capaz de aprender diretamente a partir de imagens e pesos de uvas colhidas.

Em *A Grape Dataset for Instance Segmentation and Maturity Estimation* (BLEKOS *et al.*, 2023), percebendo que conjuntos de dados anteriores tinham tamanho limitado

para aplicações em aprendizado profundo, os autores introduziram o *dataset* CERTH. Eles criaram um conjunto de dados de 2502 imagens, contendo 9832 cachos de uvas da variedade *Crimson Seedless*, anotados utilizando o modelo RITM (SOFIIUK; PETROV; KONUSHIN, 2021).

As imagens foram coletadas a partir de um iPhone 11 e, além das anotações de máscaras de instâncias, também possuem rótulos de maturidade, correspondendo a três classes: maduras, imaturas e semi-maduras. Outra característica do CERTH é que ele inclui condições desafiadoras de iluminação e visualização, com o objetivo de simular cenários reais de campo.

Para elaborar uma linha de base para a segmentação de cachos de uvas, modelos de detecção e segmentação de imagens foram treinados e avaliados no CERTH, incluindo variações do Mask R-CNN (HE; GKIOXARI *et al.*, 2017) e o Mask2Former (CHENG; MISRA *et al.*, 2022) com *backbone* Swin *Small* (LIU, Z. *et al.*, 2021). O Mask2Former apresentou os resultados mais promissores em comparação com outros modelos.

4 METODOLOGIA

A execução deste trabalho foi dividida em etapas distintas, cada uma com objetivos e tarefas específicas, visando a construção de um sistema robusto de segmentação de imagens para vinhedos.

4.1 VISÃO GERAL DOS MACRO-PASSOS

1. **Anotação dos Dados Preliminares:** Coleta e anotação das imagens das parreiras para criar um conjunto de dados inicial.
2. **Teste dos Modelos de Segmentação Interativa:** Avaliação de diversos modelos e técnicas de segmentação interativa para identificar a melhor abordagem.
3. **Treinamento Supervisionado e Teste com Dados Preliminares:** Treinamento e validação dos modelos de segmentação usando o conjunto de dados preliminar.
4. **Treinamento e Teste com Dados Finais:** Treinamento e avaliação final dos modelos utilizando o conjunto de dados definitivo.

A primeira etapa do projeto consistiu na coleta de imagens relevantes das parreiras, realizada pela célula uruguaia da equipe, que desenvolveu um dispositivo específico para filmagem das fileiras de videiras estudadas.

Dada a limitação de tempo entre a coleta das imagens no Uruguai, sua anotação e a associação destas com os dados de colheita, foram investigadas alternativas para a anotação eficiente das imagens. Diversos modelos e serviços de segmentação interativa foram comparados com o objetivo de identificar as técnicas mais adequadas às necessidades do projeto.

Com uma técnica eficaz de anotação de imagens estabelecida, o projeto avançou para a etapa de avaliação e posterior *fine-tuning* dos modelos de segmentação de imagens. Esta fase foi crucial para garantir a precisão e a eficiência dos modelos desenvolvidos.

Por fim, com os modelos definidos, repositórios preparados e hiperparâmetros ajustados, o projeto progrediu para a fase de treinamento e avaliação final dos modelos utilizando um conjunto de dados definitivo. Este conjunto foi composto por imagens de uvas da variedade Tannat, coletadas no início de 2024.

4.2 MATERIAIS NECESSÁRIOS

Os materiais físicos e coleta de dados necessários para o desenvolvimento do projeto foram fornecidos e utilizados pela equipe no Uruguai.

Na área de visão computacional e aprendizado de máquina, foi necessário estritamente poder computacional para treinamento e avaliação dos modelos, além de acesso

a internet para pesquisa e comunicação. As simulações foram realizadas nas seguintes configurações:

- Computador pessoal do autor deste trabalho, um com GPU NVIDIA GeForce GTX 1060 3GB e um processador AMD FX8300.
- Google Colab Pro, selecionando placas de vídeo como a Tesla T4 e L4.
- Servidor da SeTIC (Secretaria de Tecnologia da Informação e Comunicação) UFSC, contando com a GPU RTX 4090 24GB de memória de vídeo.

4.3 AQUISIÇÃO DE DADOS (*DATASETS*)

As seções seguintes detalham a aquisição e processamento dos conjuntos de dados utilizados na avaliação e treinamento dos modelos analisados neste estudo.

4.3.1 Tannat 2023 (TAN-23)

O conjunto de dados Tannat 2023 (TAN-23) foi disponibilizado no início do projeto, contendo fotos produzidas no início de 2023 com uma câmera Nikon D3200 na vitícola localizada próximo a Montevidéu, no Uruguai. Este *dataset* é composto por imagens de altíssima resolução (6016x4000) de uvas da variedade Tannat. Por ser amplamente utilizado neste trabalho, serviu como base para a comparação de ferramentas e serviços de anotação interativa, bem como para a avaliação preliminar e seleção de modelos de segmentação semântica supervisionada.

Para a avaliação preliminar de modelos de segmentação supervisionada, foram utilizadas 128 imagens com máscaras de segmentação semântica, divididas em conjuntos de treinamento, validação e teste.

Para aferições de técnicas de segmentação interativa, foram selecionadas e anotadas 32 imagens com máscaras de segmentação semântica. Adicionalmente, foram anotadas *bounding boxes* e pontos (emulando cliques) para a comparação dos métodos.

A disposição agrupada dos cachos da variedade de uvas Tannat representa um desafio significativo para a segmentação individual de uvas. Por essa razão, foram desenhadas *bounding boxes* em torno de grupos de cachos nas imagens com máscaras previamente anotadas. Em cada caixa, foram anotados três pontos, organizados da seguinte forma: o primeiro ponto é necessariamente positivo, enquanto o segundo e o terceiro podem ser positivo e/ou negativo, dependendo do formato e da disposição de cada cacho ou grupo de cachos. Esses pontos anotados serão utilizados como *inputs* de cliques nas simulações de anotação subsequentes.

As anotações auxiliares das 32 imagens foram exportadas em *Supervisely format*, com um arquivo JSON para cada imagem que armazena as máscaras de segmentação semântica, *bounding boxes* e pontos.

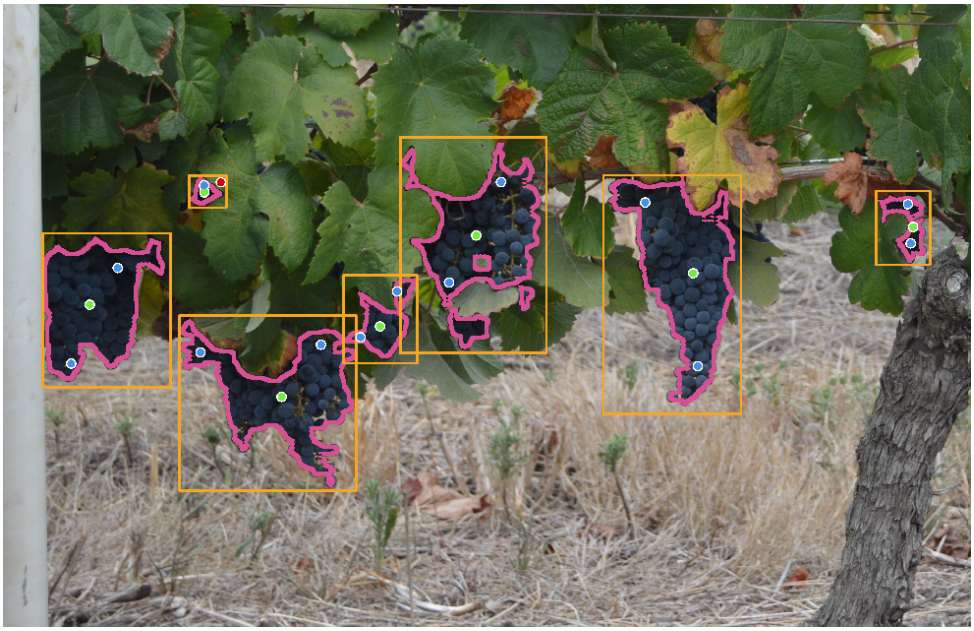


Figura 4 – Exemplo de anotações de pontos e *bounding boxes* em uma imagem do *dataset* TAN-23. Elaboração própria,anotação realizada pela plataforma Supervisely.

As anotações auxiliares das 32 imagens foram exportadas no formato *Supervisely*, gerando um arquivo JSON para cada imagem. Esses arquivos armazenam as máscaras de segmentação semântica, as *bounding boxes* e os pontos.

4.3.2 WGISD

O *Embrapa Wine Grape Instance Segmentation Dataset* (WGISD) contém 300 imagens de 4.432 cachos de uva de cinco variedades diferentes, capturadas no campo em 2019.

Para este Trabalho de Conclusão de Curso, foi utilizada uma parcela das imagens do WGISD para a avaliação de modelos de segmentação interativa. Como o foco do projeto está nas uvas da variedade Tannat, foram selecionadas imagens das variedades Cabernet Sauvignon e Syrah deste *dataset*, devido à sua semelhança visual com a Tannat. As imagens das variedades Syrah e Cabernet Sauvignon do WGISD foram tratadas como conjuntos de dados distintos, denominados neste trabalho como WGISD-SYH e WGISD-CSV, respectivamente, ou apenas como SYH e CSV afim de evitar repetições.

O conjunto WGISD-CSV contou com 28 imagens e WGISD-SYH com 23 imagens, obtidas do repositório do WGISD, disponível em (SANTOS, 2021). As máscaras de instâncias dos cachos de uvas foram fornecidas como uma sequência de matrizes em um arquivo NPZ, as *bounding boxes* em um arquivo TXT, e a localização das bagas, anotadas por (KHOROSHEVSKY; KHOROSHEVSKY; BAR-HILLEL, 2021), foram salvas em um documento TXT distinto. Os pontos referentes às bagas foram combinados utilizando a técnica K-means, com $K = 1$ e $K = 3$, simulando cenários de cliques nos objetos.

Figura 5 – Exemplo de anotações de *bounding boxes* e máscaras binárias do *dataset* WGISD, junto aos pontos produzidos pela técnica K-means para $K = 3$.

Fonte: Autor.

4.3.3 Tannat 2024 (TAN-24)

O conjunto de dados final, contendo imagens de uvas da variedade Tannat, foi coletado na mesma vitícola do TAN-23, no início de 2024, utilizando uma câmera GoPro 11 Black Edition.

O projeto uruguaio teve como objetivo coletar imagens das inflorescências e dos cachos de uvas do vinhedo para estimar a colheita a partir de dados coletados em momentos distintos. No entanto, até o final do semestre letivo, apenas análises iniciais das imagens dos cachos foram realizadas. Essas imagens foram disponibilizadas para este trabalho e utilizadas para o treinamento e avaliação dos modelos de segmentação supervisionada, partindo do modelo e hiperparâmetros selecionados a partir das avaliações e inferências do TAN-23.

O dispositivo de suporte das câmeras, projetado para ser guinchado por um trator, foi desenvolvido pela célula uruguaia do projeto. Este dispositivo foi equipado com dois pares de câmeras, duas para cada fileira de videiras, o que possibilita a reconstrução tridimensional da vinha no futuro.

Sistemas baseados em GPS não possuem precisão suficiente para o acompanhamento preciso de fileiras de videiras. Embora existam dispositivos auxiliares que podem ser acoplados a tratores para este fim, esses não foram utilizados no projeto. O georre-

ferenciamento das imagens foi realizado de forma automatizada, utilizando QR Codes fixados nos postes. Posteriormente, foi realizado o pós-processamento das imagens e o agrupamento por fileiras.

Como as imagens foram coletadas durante a noite, foram necessárias luminárias LEDs montadas no próprio dispositivo de coleta. A escolha de trabalhar com o cenário noturno reduziu a complexidade cromática e de contrastes, além de evitar a segmentação de fileiras adjacentes.

Figura 6 – Exemplo de fotografia noturna do *dataset* TAN-24.

Fonte: UdelaR.

O *dataset* TANNAT-24 foi composto por 120 imagens de uvas da variedade Tannat, contando com 90 imagens para treinamento, 15 para validação e 15 para teste, selecionadas aleatoriamente. Este conjunto não foi utilizado para a avaliação de modelos de segmentação interativa ou para a validação cruzada dos modelos de segmentação supervisionada, sendo reservado exclusivamente para o treinamento e avaliação final dos modelos.

4.4 MODELOS DE SEGMENTAÇÃO INTERATIVA

Dada a complexidade intrínseca da anotação de máscaras em imagens, a seleção de uma plataforma eficiente e precisa foi crucial para o sucesso deste trabalho. Nesta etapa, foram realizadas avaliações qualitativas e quantitativas de diferentes ferramentas e plataformas de anotação de imagens.

Modelos de segmentação mais complexos tendem a oferecer maior precisão, mas requerem mais recursos computacionais e tempo de processamento. Em contraste, modelos mais leves podem ser menos eficientes na segmentação, aumentando o tempo necessário para anotação.

As avaliações preliminares dos modelos de segmentação interativa e aqueles submetidos ao *fine-tuning* foram realizadas utilizando imagens distintas do *dataset* final (TAN-24). Essa abordagem permitiu a construção antecipada de um *workflow* eficiente, liberando mais tempo para pesquisas, desenvolvimento de repositórios de código e testes.

O desenvolvimento de uma interface de anotação é uma tarefa complexa e está fora do escopo deste trabalho. Portanto, foram comparadas plataformas de anotação de código aberto e/ou com licenças gratuitas.

4.4.1 Avaliação Qualitativa

A anotação de dados, por natureza, requer a intervenção humana em alguma de suas etapas, tornando a experiência do usuário uma prioridade na elaboração de um processo eficiente. O principal objetivo é obter máscaras de alta qualidade com o mínimo de interações do anotador e no menor tempo possível.

Foram investigadas plataformas populares de anotação de imagens, como Label Studio (HEARTEX, 2024), Supervisely (TEAM, S., 2024), Label Anything (OPENMMLAB, 2024), LabelBox (TEAM, L., 2024), CVAT (TEAM, C., 2024) e RoboFlow (TEAM, R., 2024). *Jupyter Notebooks* disponibilizados nos repositórios oficiais das arquiteturas analisadas também foram avaliados e serviram de base para o *script* da avaliação quantitativa.

Inerentemente, a anotação de dados exigirá um humano em alguma de suas etapas, tornando a experiência do usuário uma prioridade ao planejar um processo eficiente. O principal objetivo nesta é a obtenção máscaras de alta qualidade a partir do mínimo de interações do anotador e no menor tempo possível.

Plataformas populares de anotação de imagens foram investigadas, como Label Studio (HEARTEX, 2024), Supervisely (TEAM, S., 2024), Label Anything (OPENMMLAB, 2024), LabelBox (TEAM, L., 2024), CVAT (TEAM, C., 2024) e RoboFlow (TEAM, R., 2024). *Jupyter Notebooks* disponibilizados nos repositório oficiais das arquiteturas analisadas também foi avaliado e serviu de base para o *script* da avaliação quantitativa.

4.4.1.1 SAM

Entre os modelos avaliados, o SAM destacou-se por oferecer a maior quantidade de *prompts* disponíveis, sendo o primeiro modelo investigado. Lançado ao público em abril de 2023, pouco antes do início deste trabalho, representou um avanço significativo na segmentação de imagens. Sua popularidade foi tão grande que todas as plataformas online de anotação de imagens avaliadas incluíram alguma versão deste modelo.

As funcionalidades do SAM foram implementadas de forma diferente em cada serviço. Por exemplo, alguns permitiam o uso de pontos e *bounding boxes*, enquanto outros aceitavam apenas uma dessas opções por vez. Enquanto alguns serviços permitiam o uso do *crop* das imagens antes da inferência, outros não. Nas plataformas avaliadas, os testes com o SAM foram concentrados na plataforma *LabelBox*, devido à sua parceria pré-estabelecida com a UFSC, embora algumas imagens também tenham sido anotadas em outras plataformas.

O modelo SAM não teve sucesso na segmentação de uvas utilizando *prompts* de texto. No entanto, as segmentações a partir de *bounding boxes* e pontos mostraram-se promissoras.

Desenhar *bounding boxes* em torno dos objetos é uma tarefa prática, comumente utilizada em anotações de imagens para detecção de objetos. Fornecendo *bounding boxes* como *prompts*, as máscaras obtidas apresentaram qualidade razoável.

Na segmentação a partir de pontos, há duas formas distintas de interação: a adição de cliques positivos/negativos na imagem completa ou o *crop* da imagem a partir de uma caixa seguido dos cliques/pontos. Trabalhando com a imagem completa, o modelo foi incapaz de identificar corretamente os cachos de uvas, confundindo-se com sombras e arbustos. Fornecendo os cliques após o *crop* das regiões mostrou-se mais promissor, mas os resultados ainda foram razoáveis, não atingindo a qualidade desejada.

Embora não haja um limite para o número de cliques, o modelo demonstrou pouca adaptação em relação à sua primeira inferência. Tanto na segmentação por cliques quanto por caixas, foi frequentemente necessário ajustar manualmente as máscaras fornecidas utilizando *brush tools* para alcançar máscaras de alta qualidade.

Após anotar algumas imagens e corrigir manualmente as máscaras, constatou-se que esse ajuste com *brush tools* demandava praticamente o mesmo tempo que a anotação manual completa, podendo levar vários minutos para ajustar uma única máscara. Esse resultado não está alinhado com o propósito de uma anotação interativa, que deveria ser mais rápida e eficiente que a anotação manual.

4.4.1.2 RITM e ClickSEG

Os modelos RITM e ClickSEG são mais simples e focados exclusivamente em interações com cliques, com *codebases* semelhantes. Esses modelos trabalham apenas com a função de *crop*, seguida de cliques positivos e/ou negativos, não permitindo o uso de *bounding boxes* como *prompts*.

Além disso, a popularidade desses modelos não é comparável à do SAM no domínio da anotação interativa. Isso é evidenciado pelo fato de que, além de uma interface em *Python* construída com *Tkinter*, disponibilizada pelos desenvolvedores de RITM e reutilizada no repositório de ClickSEG, apenas uma plataforma de anotação online oferece suporte a esses modelos: a Supervisely.

Conforme descrito por *kirillov2023segment*, percebeu-se que esses modelos não segmentaram os objetos tão bem quanto o SAM nos primeiros cliques. No entanto, ao trabalhar com cachos de uvas menos visíveis, com anotações mais difíceis, utilizando RITM ou ClickSEG, não foi necessário nenhum tipo de correção manual nas imagens.

Esses modelos demonstraram maior adaptabilidade a cada novo *input* fornecido, permitindo a criação de máscaras muito mais refinadas em comparação com a primeira interação.

4.4.1.3 FastSAM

Com o alongado tempo de inferência do SAM agregado a seu baixo desempenho com pontos, buscou-se uma alternativa de validar alternativas a *segment anything task* de maneira mais rápida. FastSAM não está disponível nas plataformas avaliadas e seu uso prático não seria viável no projeto, mas sua performance foi avaliada em um ambiente controlado para fins de comparação.

4.4.2 Avaliação Quantitativa

Os *datasets* TANNAT 2023 (TAN-23) e WGISD (CSV e SYH) foram utilizados para as avaliações.

Além da percepção qualitativa dos métodos e ferramentas de anotação, foram elaborados testes automatizados de cliques e *bounding boxes*, visando uma avaliação sistemática e consistente dos modelos.[1]

Os modelos de segmentação interativa foram avaliados nos seguintes cenários/*prompts*:

- **1 ponto:** A caixa delimitadora foi utilizada para recortar a imagem, e um ponto foi fornecido como entrada para o modelo.
- **3 pontos:** A caixa delimitadora foi utilizada para recortar a imagem, e três pontos foram fornecidos como entrada para o modelo.
- **BB:** A imagem completa foi fornecida para inferência do modelo, e cada *bounding box* foi utilizada como *prompt*.
- **BB+Ponto:** Similar ao caso anterior, mas com a adição de um ponto positivo fornecido como *prompt* dentro de cada caixa.

RITM, ClickSEG e FastSAM foram avaliados na simulação com 1 e 3 cliques.

O modelo SAM lida com a ambiguidade retornando três máscaras válidas, cada uma associada a um *score* (pontuação) de confiança. A saída do modelo é obtida a partir

[1] O repositório com os *scripts* personalizados para testar cada arquitetura está disponível em `https://github.com/vitordj/semantic-segmentation-transformers`

Figura 7 – Exemplos de fotografias dos *datasets* utilizados na avaliação de modelos de segmentação interativa.

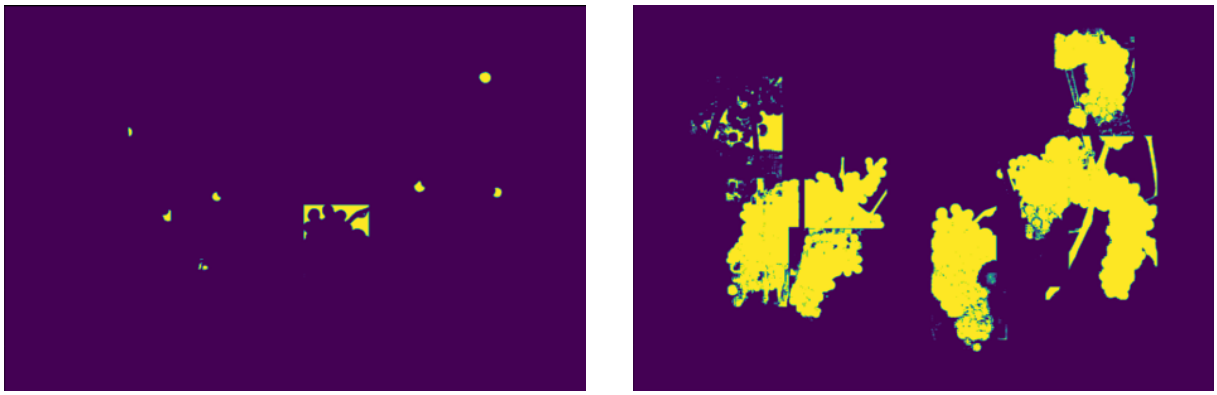
(a) WGISD-CSV (b) WGISD-SYH (c) TAN-23

Fonte: Embrapa WGISD e UdelaR.

da combinação dessas três máscaras, exceto em cenários onde o único *input* fornecido é um ponto.

Quando apenas um ponto é fornecido, o modelo pode retornar as três máscaras ou apenas a máscara com a maior pontuação de confiança. Na simulação automatizada com 1 clique, as máscaras retornadas correspondiam à segmentação das bagas individuais das uvas ou a apenas partes dos cachos. Considerando o elevado tempo de inferência do modelo e os recursos computacionais limitados nesta etapa, o cenário de avaliação com 1 clique não foi estudado para o SAM.

Figura 8 – Exemplos de simulações de cliques com o modelo SAM na mesma imagem.



(a) 1 ponto (b) 3 pontos

Fonte: Autor.

A simulação com 3 pontos foi realizada, apresentando resultados mais promissores,

mas ainda significativamente inferiores às expectativas, dada a popularidade do modelo.

Supondo que esse baixo desempenho estivesse relacionado ao *crop* das imagens, foram realizados testes com a imagem completa, fornecendo apenas as *bounding boxes* como entrada. Com resultados mais promissores, os testes foram repetidos com a inclusão de um ponto positivo dentro das caixas (BB+Ponto) para o modelo *Segment Anything*. Esses dois últimos cenários foram aplicados exclusivamente ao SAM.

Para cada arquitetura, foram utilizados os seguintes modelos e configurações:

- **RITM**: *Backbone* HRNet32+OCR IT-M, treinado nos *datasets* COCO e LVIS. *Itermask* (inclusão da máscara anterior na inferência) foi ativado, enquanto o BRS foi desativado.
- **SAM**: *Backbone* ViT Huge, treinado no *dataset SA-1B*.
- **ClickSEG**: *Backbone* SegFormer MiT-B3, com a arquitetura ClickSEG S2 (maior resolução), treinado nos *datasets* COCO e LVIS. O BRS foi desativado.
- **FastSAM**: *Backbone* YOLO-X, treinado com apenas 1/50 do *dataset SA-1B*.

A avaliação dos cenários de anotação de imagens de uvas foi baseada nas anotações *ground truth*, utilizando a métrica Intersection over Union (IoU). Todos os testes das metodologias de anotação interativa foram realizados em um computador pessoal, equipado com uma GPU NVIDIA GeForce GTX 1060 3GB e um processador AMD FX8300.

4.5 TREINAMENTO E AVALIAÇÃO DE MODELOS DE SEGMENTAÇÃO SUPERVISIONADA

Com uma técnica eficaz para anotação de imagens estabelecida, o foco do trabalho passou a ser o *fine-tuning* e a avaliação de modelos de segmentação supervisionada de imagens, que constituem o principal objetivo deste Trabalho de Conclusão de Curso.[2] Dado que o *dataset* definitivo ainda não estava disponível, foi necessário utilizar um conjunto de dados intermediário para o treinamento e validação dos modelos.

Este conjunto de dados intermediário foi composto por 128 imagens de uvas da variedade Tannat do *dataset* TAN-23, com máscaras de segmentação semântica anotadas manualmente. O conjunto foi dividido em 96 imagens para treinamento, 16 para validação e 16 para teste.

Para a seleção de métodos e hiperparâmetros, os testes em cada arquitetura foram realizados com base na configuração padrão de cada modelo, conforme descrito em suas respectivas publicações. Cada fold contou com 60 épocas, dado que a precisão na segmentação de uvas era mais relevante para este projeto do que a segmentação em tempo real. Assim, foram selecionadas e avaliadas as maiores versões de cada arquitetura.

[2] O repositório com os *scripts* de treinamento, validação e inferência estão disponíveis em `github.com/vitordj/semantic-segmentation-transformers`

- **SegFormer**: SegFormer MiT-B5, utilizando o otimizador AdamW, com uma taxa de aprendizado de 0.00006 e um *scheduler* polinomial com fator de 1.
- **MaskFormer**: MaskFormer com *backbone* Swin-Large, treinado no ADE20k, utilizando o otimizador AdamW, com uma taxa de aprendizado de 0.00006 e um *scheduler* polinomial com fator de 1.
- **Mask2Former**: MaskFormer com *backbone* Swin-Large, treinado no ADE20k, utilizando o otimizador AdamW, com uma taxa de aprendizado de 0.0001 e um *scheduler* polinomial com fator de 1.
- **OneFormer**: OneFormer com *backbone* DiNAT-Large, treinado no ADE20k, utilizando o otimizador AdamW, com uma taxa de aprendizado de 0.0001 e um *scheduler* polinomial com fator de 1.

Por questões de simplicidade e otimização computacional, não foram aplicados *weight decay* nem *data augmentation*. Além disso, não foi possível utilizar os mesmos *batch sizes* dos treinamentos originais. Para cada modelo, o *batch size* foi ajustado para o máximo possível sem causar estouro de memória de vídeo.

O modelo selecionado a partir da inferência no *dataset* TANNAT-23 e sua configuração foram utilizados para o treinamento e avaliação final de desempenho no *dataset* TANNAT-24.

4.5.1 *Focal Loss* no SegFormer

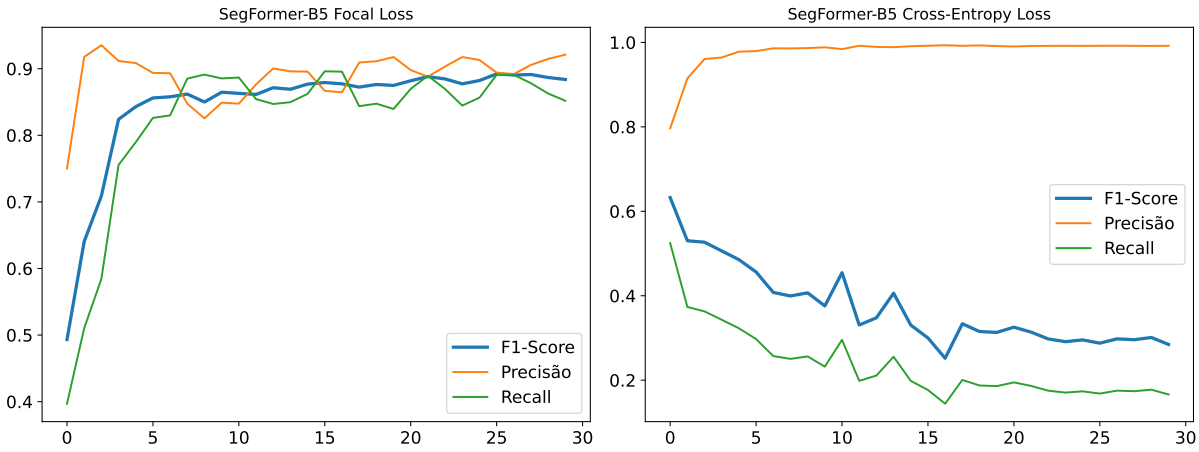
Uma das principais contribuições deste trabalho é a implementação da *Focal Loss* (LIN *et al.*, 2017) na saída do SegFormer e a análise de seus resultados.

Durante os treinamentos preliminares com o SegFormer, observou-se um *recall* muito baixo, acompanhado de alta precisão na segmentação de uvas. Uma análise exploratória nas anotações de *ground truth* revelou que aproximadamente 4% dos pixels anotados eram referentes a uvas.

A arquitetura do SegFormer aplica uma função de perda de entropia cruzada em seus *logits*, que é pouco eficiente em casos de desbalanceamento de classes. Hipotetizando que este comportamento seria reflexo do desbalanceamento de classes, implementou-se a *Focal Loss* nos *logits* do SegFormer.

Os autores de *Assessing Macro Disease Index of Wheat Stripe Rust Based on Segformer with Complex Background in the Field* (DENG *et al.*, 2022) também enfrentaram um problema semelhante ao trabalhar com o SegFormer em imagens de ferrugem listrada do trigo com um *dataset* desbalanceado. Nesse estudo, a *Focal Loss*, juntamente com a reamostragem de dados, aumentou significativamente a F1-score da classe de ferrugem de 72,6% para 86,6%, demonstrando que essas técnicas são eficazes para resolver problemas de desbalanceamento de classes.

Figura 9 – Desempenho do modelo SegFormer com a perda focal comparado a perda de entropia cruzada.



Fonte: Autor.

Após a troca da função de perda, o uso do SegFormer não apenas se tornou viável, mas também resultou em desempenho significativamente superior, como pode ser constatado na Figura 9. Portanto, a *Focal Loss* foi mantida para este modelo.

4.5.2 Validação cruzada *K-fold*

Para avaliar os hiperparâmetros dos modelos, considerando o tamanho modesto do *dataset* intermediário TAN-23, utilizou-se a técnica de validação cruzada *K-fold*, com 7 *folds*. Além do erro de treino e validação, foram acompanhadas as métricas de precisão, recall e F1-score no conjunto de validação.

O principal objetivo nesta etapa foi a compreensão de como as diferentes arquiteturas se comportam com diferentes *schedulers* de taxa de aprendizado.

Considerando que todos os modelos foram originalmente treinados com um *scheduler* "poly"de potência 1 (queda linear) e que foram utilizados modelos pré-treinados, ao invés de partir de pesos aleatórios, foram também avaliados modelos com o mesmo *scheduler* com uma queda quadrática, reduzindo a taxa de aprendizado de forma mais acelerada. Sem êxito, conforme detalhado nos resultados, os modelos também foram avaliados com uma taxa de aprendizado constante, sem um *scheduler*.

Para uma comparação justa entre os modelos nesta etapa, utilizou-se uma resolução fixa de 512×512 pixels para todas as imagens e um *batch size* $= 8$ nos experimentos de validação cruzada. Parte dos experimentos foi realizada nas máquinas hospedadas pela SeTIC UFSC, equipadas com uma GPU RTX 4090 com 24GB de memória de vídeo, enquanto outra parte foi executada no *Google Colab Pro*.

4.5.3 Treinamento e Avaliação (TANNAT-23)

Após a validação e seleção de bons hiperparâmetros para cada arquitetura, o modelo com melhor desempenho na segmentação foi selecionado. Para garantir maior robustez na análise, adicionalmente para cada arquitetura foi treinado um modelo com a melhor configuração encontrada via validação cruzada, em seguida foram exportados.

Os modelos foram treinados com a mesma resolução de seus treinamentos originais. MaskFormer, Mask2Former e OneFormer foram treinados com resolução de 640×640 pixels, enquanto SegFormer foi treinado com 768×768 pixels. Embora seja vantajoso trabalhar com uma resolução ainda maior para todos os modelos, isso exigiria uma quantidade proibitiva de memória de vídeo. Esta limitação de memória também influenciou o *batch size*, que foi configurado para 4 imagens em todos os treinamentos.

Os modelos treinados foram utilizados para inferência no conjunto de teste, composto por 16 imagens de uvas da variedade Tannat do *dataset* TAN-23. Parte do conjunto de teste foi composta por imagens de fileiras utilizadas nos conjuntos de treinamento e validação, enquanto outra parcela foi composta por imagens de fileiras não utilizadas anteriormente.

Para todos os modelos, a inferência foi realizada na resolução de 1024×1024 pixels. Resoluções maiores não apresentaram ganhos significativos na qualidade das máscaras.

O objetivo principal desta abordagem foi avaliar como os modelos iriam se comportar na tarefa final do sistema desenvolvido neste trabalho: a contagem de *pixels* de uvas. A partir da área obtida para cada imagem, os erros de contagem foram calculados a partir das anotações *ground truth* e comparados.

4.5.4 Treinamento e Avaliação (TANNAT-24)

Com um entendimento sólido do funcionamento dos modelos e de suas capacidades, o projeto progrediu para a fase de treinamento e avaliação final dos modelos utilizando o *dataset* definitivo TANNAT-24.

Nesta etapa, foram aplicados os mesmos hiperparâmetros e configurações de treinamento selecionados a partir dos experimentos com o TAN-23, o conjunto de validação foi utilizado apenas para o *early stopping* e a avaliação final foi realizada no conjunto de teste.

O conjunto foi dividido em 90 imagens para treinamento, 15 para validação e 15 para teste. Apenas o melhor modelo encontrado na etapa anterior foi aplicado ao TAN-24, com a mesma resolução de inferência e *batch size*.

Assim como no TAN-23, a inferência foi realizada na resolução de 1024×1024 pixels, e os resultados foram comparados com as anotações *ground truth* tanto para as métricas de segmentação quanto para a contagem de *pixels* de uvas e análise das métricas de erros.

5 RESULTADOS

Nesta seção, serão apresentados os resultados dos experimentos realizados para este trabalho.

Inicialmente, aborda-se a avaliação dos modelos de segmentação interativa, seguida pela análise de desempenho dos modelos a partir do *dataset* intermediário TANNAT-23, para que finalmente a melhor arquitetura e configuração seja aplicada ao *dataset* definitivo, TANNAT-24.

5.1 MODELOS DE SEGMENTAÇÃO INTERATIVA

5.1.1 Interseção sobre União (IoU) Média

Após aplicar a metodologia de testes de *prompts* em diferentes modelos de segmentação interativa, foi possível avaliar a eficácia de cada método.

Tabela 1 – IoU média por modelos, *prompts* e *datasets* avaliados.

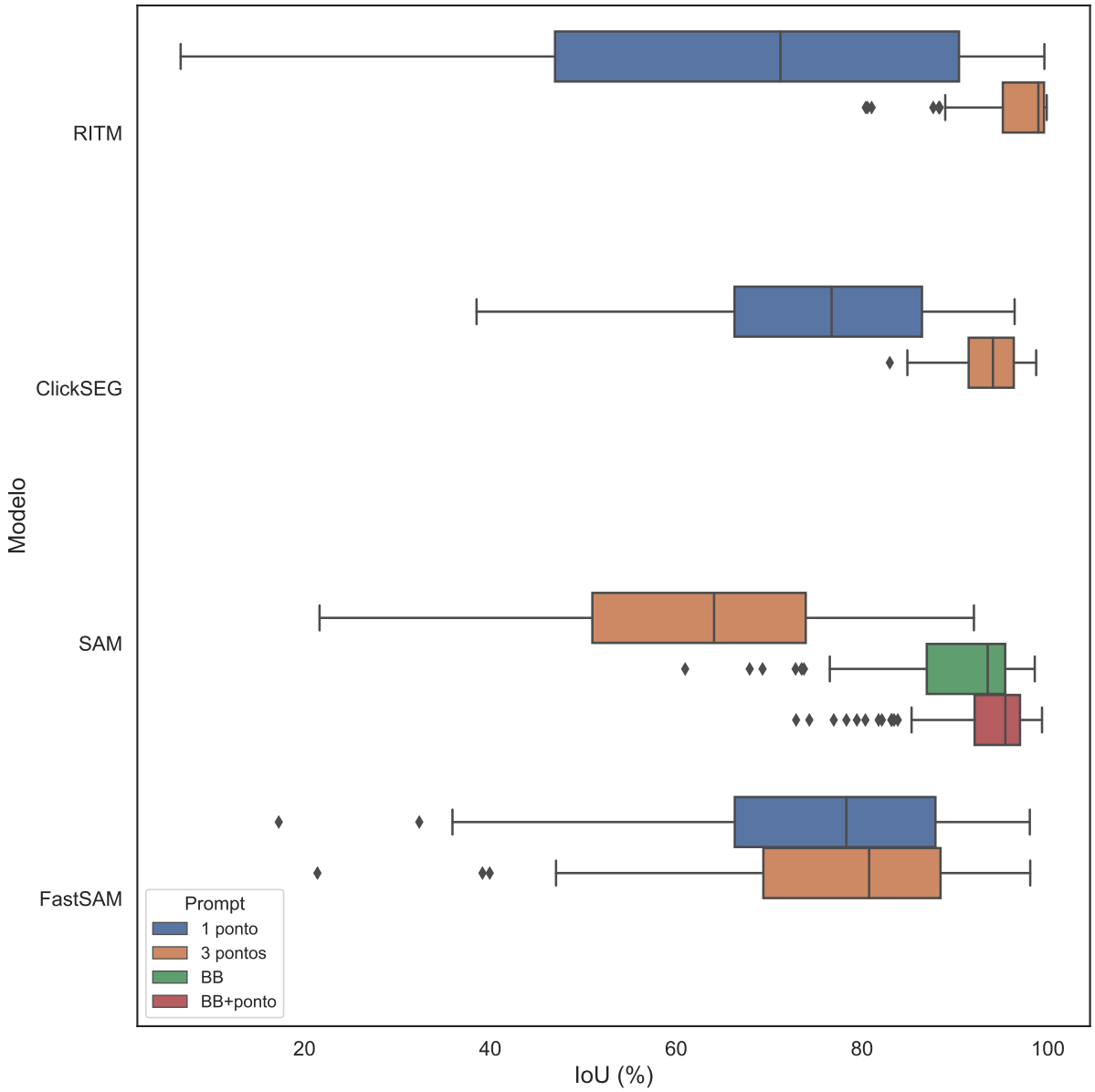
Modelo	Prompt	Dataset		
		TAN-23	CSV	SYH
RITM	1 ponto	47.2%	79.5%	84.3%
	3 pontos	92.8%	99.1%	99.0%
ClickSEG	1 ponto	78.1%	74.2%	74.8%
	3 pontos	93.5%	92.8%	94.3%
SAM	3 pontos	57.2%	59.9%	71.1%
	BB	82.7%	95.0%	94.0%
	BB+ponto	88.2%	96.0%	95.6%
FastSAM	1 ponto	71.4%	74.6%	79.4%
	3 pontos	75.2%	76.3%	80.5%

Fonte: Autor.

A análise dos resultados de IoU revelou que os modelos RITM e ClickSEG, quando utilizados com 3 pontos de interação, bem como o modelo SAM com *bounding boxes* (BB e BB + ponto), apresentaram desempenho superior em termos de precisão de segmentação. Notavelmente, todos os modelos enfrentaram desafios significativos ao segmentar o *dataset* TAN-23, enquanto o *dataset* SYH, composto por uvas verdes, facilitou a segmentação devido à menor oclusão causada pela similaridade cromática entre as uvas e o fundo.

Os resultados indicam que a utilização de um maior número de pontos de interação melhora a consistência dos resultados de segmentação, conforme evidenciado pela menor

Figura 10 – *Boxplot* da métrica IoU por *dataset* e *prompt*.



Fonte: Autor.

variância observada entre diferentes imagens (Figura 10). No experimento com 1 ponto, o modelo RITM apresentou resultados inferiores, mas no cenário com 3 pontos demonstrou-se preciso e consistente, obtendo os melhores resultados entre os experimentos.

Para o modelo SAM, a abordagem utilizando *bounding boxes* aplicadas à imagem inteira, sem recortes (*crops*), foi mais eficiente do que a abordagem com 3 cliques. Este resultado é particularmente interessante, pois sugere que um *prompt* com apenas a *bounding box* pode simular uma interação mínima, enquanto a adição de pontos incrementa a precisão sem a necessidade de múltiplos cliques.

A tabela 2 apresenta resultados de IoU médio por *prompt* fornecido e modelo,

enquanto a tabela 3 analisa a IoU por *prompt* e *dataset*.

Tabela 2 – IoU média por modelo e método

Modelo	1 ponto	3 pontos	BB	BB + ponto
RITM	68,4	96,6	-	-
SAM	-	62,0	90,0	92,9
ClickSEG	75,9	93,5	-	-
FastSAM	74,7	77,0	-	-

Fonte: Autor.

Tabela 3 – IoU média por dataset

Dataset	1 ponto	3 pontos	BB	BB + ponto
TAN-23	65,6	79,7	82,7	88,2
CSV	76,1	82,0	95	96
SYH	79,5	86,2	94	95,6

Fonte: Autor.

5.1.2 Tempo de Execução

Todos os experimentos de segmentação interativa foram realizados em uma máquina com uma placa de vídeo NVIDIA GeForce GTX 1060 3GB. O tempo de execução total de cada rotina foi registrado e a tabela 4 apresenta o tempo médio de execução por imagem.

Tabela 4 – Tempo de execução médio por tipo de uva, modelo e método de avaliação em minutos.

		RITM		SAM			FastSAM		ClickSEG	
Dataset	Qtd. imagens	1 ponto	3 pontos	3 pontos	BB	BB + ponto	1 ponto	3 pontos	1 ponto	3 pontos
TAN-23	32	0,26	0,26	15,10	1,26	1,15	0,28	0,28	0,04	0,04
CSV	28	0,45	0,45	12,95	1,15	0,99	0,49	0,48	0,33	0,32
SYH	23	0,48	0,57	12,86	1,40	1,21	0,54	0,57	0,39	0,39
min. / imagem		0,39	0,41	13,72	1,26	1,11	0,42	0,43	0,24	0,23

Fonte: Autor.

5.1.3 Índice de Eficiência Experimental (IEE)

Qualidade e peso computacional são fatores importantes para a escolha de um modelo de segmentação, especialmente considerando um ambiente de anotação de imagens, onde o anotador precisa aguardar cada inferência para continuar seu trabalho.

O Índice de Eficiência Experimental (IEE) é uma métrica proposta por este trabalho para avaliar o desempenho de um modelo de segmentação em relação ao tempo de execução.

Definição:

$$IEE = \frac{\text{IoU}_{\text{médio}}}{\text{Tempo de execução}_{\text{médio}}(\text{minutos})}$$

A partir dos dados calculados nas etapas anteriores, o IEE foi calculado para cada modelo, *prompt* e *dataset* de segmentação.

Tabela 5 – Índice de Eficiência Experimental (IEE) dos métodos de segmentação.

Métricas dos Métodos de Segmentação									
Modelo	RITM		SAM			FastSAM		ClickSEG	
Prompt	1 ponto	3 pontos	3 pontos	BB	BB + ponto	1 ponto	3 pontos	1 ponto	3 pontos
TAN-23	1,82	3,61	0,04	0,66	0,77	2,55	2,68	19,70	23,53
CSV	1,75	2,19	0,05	0,83	0,96	1,53	1,58	2,23	2,89
SYH	1,76	1,74	0,06	0,67	0,79	1,46	1,42	1,92	2,43

Fonte: Autor.

Os modelos ClickSEG e RITM destacaram-se no Índice de Eficiência Experimental (IEE), refletindo uma combinação eficaz de qualidade de segmentação e tempo de execução.

A arquitetura mais simples e otimizada desses modelos provavelmente contribuiu para seu desempenho superior, tornando-os candidatos ideais para aplicações em ambientes de anotação de imagens onde a inferência rápida é crucial.

A assimetria observada entre os tempos de execução do *dataset* TAN-23, especialmente na avaliação do modelo ClickSEG, pode ser atribuída aos diferentes formatos de anotação utilizados nos *datasets*. A leitura do JSON no formato Supervisely do *dataset* TAN-23 mostrou-se mais eficiente do que a leitura de planilhas (pontos obtidos por *k-means*) e TSVs (*bounding boxes*) do WGISD. Outro fator relevante é o tamanho do modelo base do SAM utilizado, que possui mais de 2GB. A placa de vídeo com 3GB de memória de vídeo foi um gargalo significativo na execução deste modelo.

Embora o SAM seja altamente eficiente utilizando apenas uma *bounding box*, frequentemente é necessário um ajuste manual para obter uma segmentação precisa, resultando em tempos de anotação de até 20 minutos por imagem.

Em função do desempenho equilibrado entre qualidade e eficiência, optou-se por adotar o modelo ClickSEG para as anotações. A plataforma Supervisely, que oferece suporte nativo para os modelos RITM e ClickSEG, foi escolhida para a anotação.

Ressalta-se que Supervisely permite uma cota diária gratuita de anotação apenas ao trabalha-ser com ClickSEG, destacando o excelente *trade-off* entre qualidade e eficiência computacional encontrado nesta arquitetura.

5.2 MODELOS DE SEGMENTAÇÃO SUPERVISIONADA

5.2.1 *Dataset* TANNAT-23

5.2.1.1 Validação Cruzada

Os modelos de segmentação supervisionada foram inicialmente treinados utilizando um *scheduler* "poly"com uma queda linear da taxa de aprendizado, de acordo com os procedimentos estabelecidos em seus treinamentos originais. No entanto, esta configuração

do *scheduler* não proporcionou resultados satisfatórios. Quando ajustado para uma queda quadrática, o desempenho dos modelos foi ainda mais comprometido.

Os resultados mais consistentes para os quatro modelos analisados foram obtidos ao desativar o *scheduler* e manter uma taxa de aprendizado constante. Essa abordagem resultou em métricas de precisão, *recall* e F1 mais robustas, indicando uma melhor convergência dos modelos.

A análise detalhada dessas métricas corrobora que a manutenção de uma taxa de aprendizado constante favorece o *fine-tuning* de modelos pré-treinados, os quais já possuem pesos estabelecidos e, portanto, beneficiam-se de uma abordagem de treinamento mais estável.

A Figura 11 ilustra a função perda por época do treinamento, ao passo que a Figura 12 apresenta as métricas de avaliação por época.

Dado que o MaskFormer compartilha a mesma meta-arquitetura que o Mask2Former, e considerando que SegFormer, Mask2Former e OneFormer demonstraram maior eficiência com uma taxa de aprendizado constante, optou-se por treinar o MaskFormer exclusivamente com essa configuração. Essa abordagem simplifica a comparação entre as arquiteturas e otimiza o uso de recursos computacionais.

O resultado pode ser observado na Figura 13.

Ao analisar os resultados obtidos na validação cruzada, verifica-se que a manutenção de uma taxa de aprendizado constante representa a abordagem mais eficaz para o treinamento dos modelos de segmentação supervisionada investigados. O Mask2Former destacou-se como o modelo com as melhores métricas de segmentação, seguido por OneFormer, SegFormer e MaskFormer.

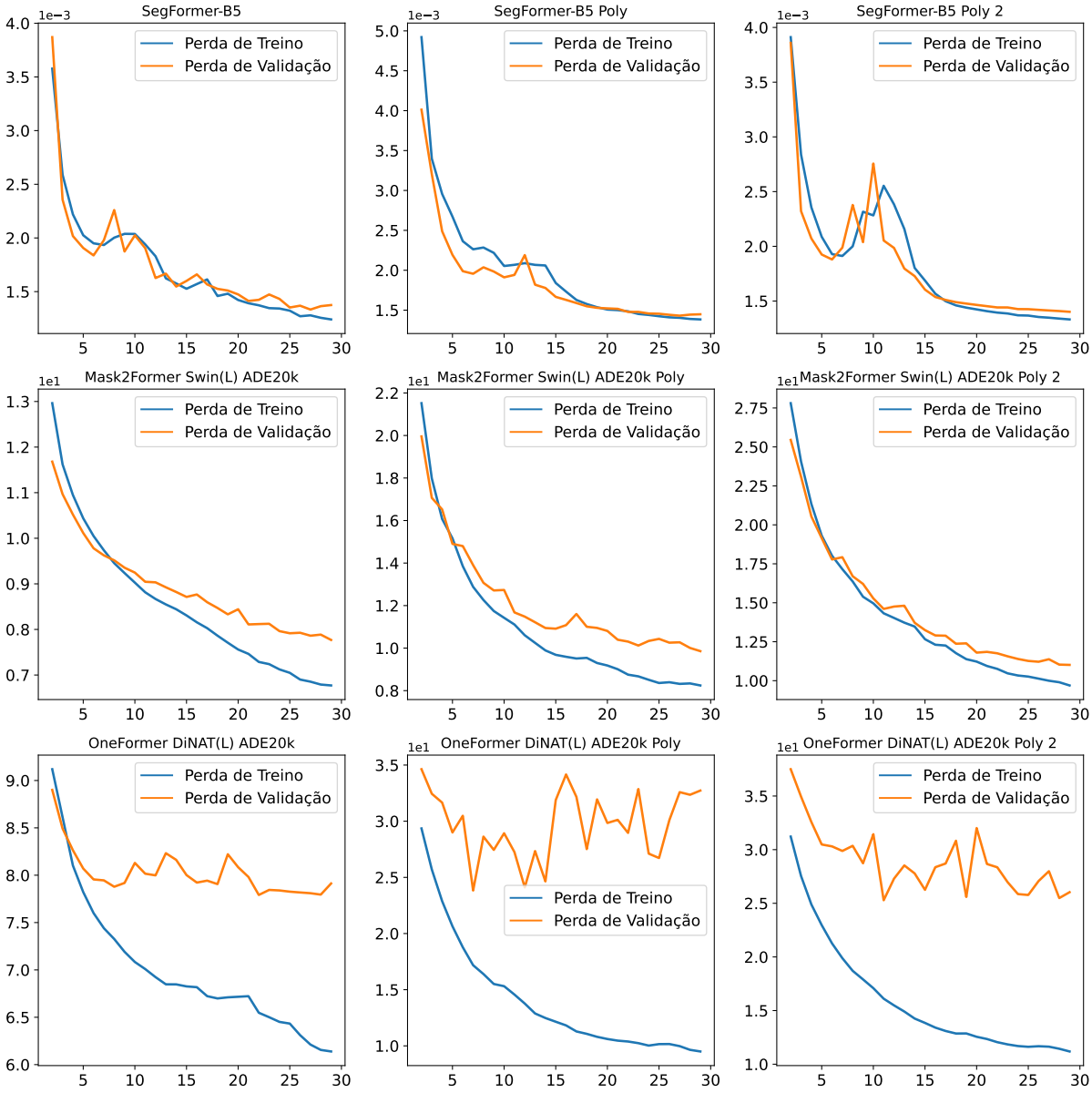
Embora a inclusão de um *poly scheduler* resulte em desempenho inferior, esta configuração tende a promover uma convergência mais estável. Essa estratégia é particularmente eficaz em treinamentos com pesos iniciais aleatórios e em *datasets* maiores, onde ajustes na taxa de aprendizado são necessários para evitar a convergência em mínimos locais. No entanto, para o *fine-tuning* de modelos pré-treinados, a taxa de aprendizado constante mostrou-se mais vantajosa, provavelmente devido aos pesos pré-estabelecidos dos modelos, o que simplifica o processo de convergência.

Adicionalmente, é relevante destacar os resultados do SegFormer. A simples inclusão da *Focal Loss* nos *logits* do modelo o tornou significativamente mais competitivo em relação aos demais, sem a necessidade de alterações consideráveis em sua arquitetura. O SegFormer também se manteve como o modelo mais eficiente em termos de recursos computacionais, conforme era esperado.

5.2.1.2 Treinamento e Avaliação

Embora o Mask2Former tenha se destacado em termos de métricas de segmentação e demonstrado ser a melhor escolha para a segmentação das uvas, sua vantagem em relação

Figura 11 – Função perda por época do treinamento, comparando o efeito do *scheduler* no treinamento dos modelos no *dataset* TAN-23.



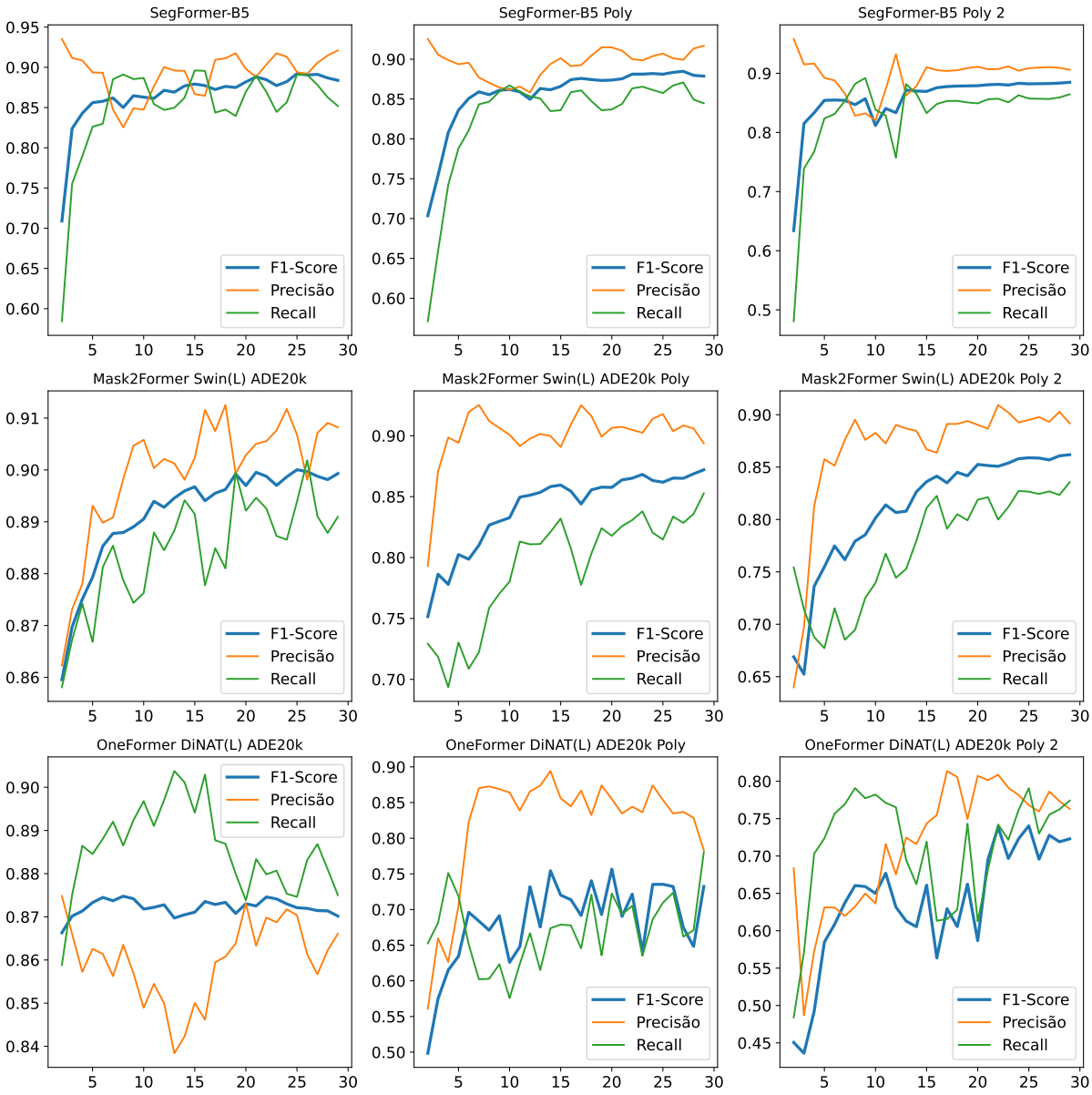
Fonte: Autor.

às outras arquiteturas não foi tão significativa. Portanto, todos os modelos foram treinados e posteriormente avaliados no conjunto de teste por questões de completude.

Com base nos insights obtidos a partir da validação cruzada, optou-se por treinar os modelos com uma taxa de aprendizado constante. O *fine-tuning* foi realizado utilizando a resolução original de treinamento de cada modelo, sendo 640×640 para todos os modelos, exceto o SegFormer, que foi treinado com resolução de 768×768. O *batch size* máximo possível foi de 4 para todos os modelos, e os treinamentos foram realizados em 30 épocas, com *early stopping* definido com paciência de 8 épocas.

Subsequentemente, os modelos treinados foram avaliados no *dataset* de teste. As

Figura 12 – Métricas de avaliação por época do treinamento, comparando o efeito do *scheduler* no desempenho dos modelos no *dataset* TAN-23.



Fonte: Autor.

inferências realizadas no *dataset* de teste geraram métricas de desempenho que são apresentadas na Tabela 6 e na Figura 14. Esses resultados fornecem uma visão abrangente sobre a eficácia dos modelos de segmentação de imagens após o treinamento.

Figura 13 – Métricas de avaliação por época do treinamento do MaskFormer comparado Mask2Former no *dataset* TANNAT-23.

Fonte: Autor.

Tabela 6 – Média dos resultados das métricas de segmentação para a avaliação dos modelos no conjunto de teste do *dataset* TANNAT-23.

Métricas / Modelo	**Precisão (%)**	**Recall (%)**	**F1-Score** (%)	**IoU** (%)
SegFormer	**88,30**	86,03	87,04	77,81
MaskFormer	82,63	**90,92**	86,25	76,53
Mask2Former	87,83	89,41	**88,56**	**79,87**
OneFormer	86,31	90,63	88,37	79,45

Fonte: Autor.

Uma análise qualitativa também foi conduzida, enfocando as máscaras geradas pelos modelos em imagens do *dataset* de teste.

Conforme ilustrado na Figura 15, embora não tenham sido anotadas, algumas uvas no fundo das imagens foram segmentadas pelos modelos.

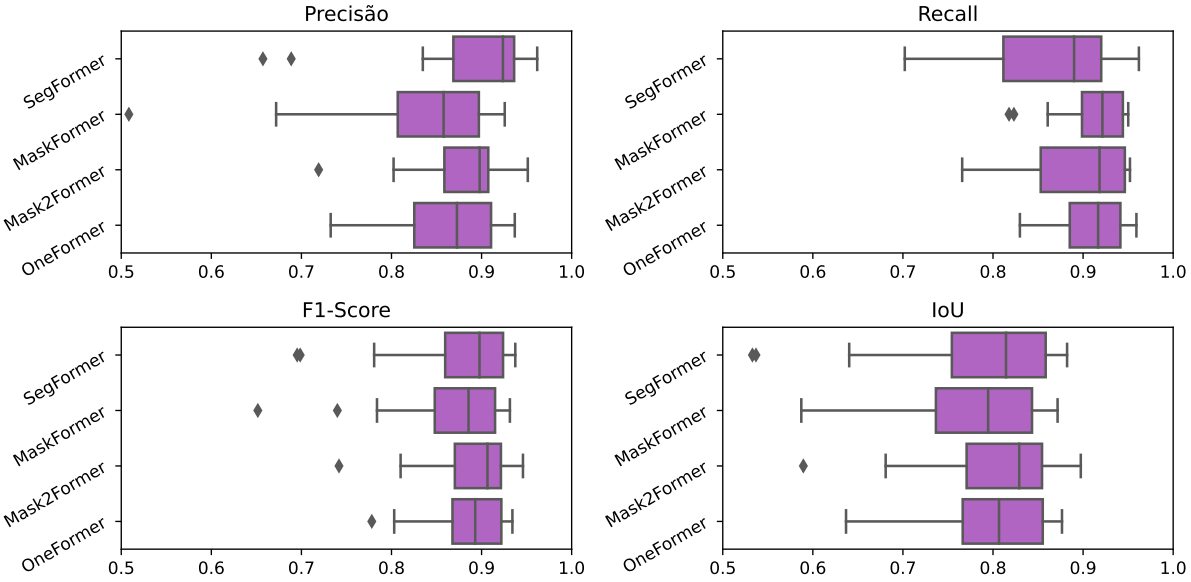
Ademais, todos os modelos mostraram-se ineficazes na segmentação de uvas verdes (Figura 16). Embora estas tenham sido anotadas, aparentemente não foram bem representadas no *dataset*.

A imagem com as melhores métricas de segmentação, maior IoU e F1-Score, foi selecionada para análise e é apresentada na Figura 17.

Adicionalmente, a Figura 18 apresenta a imagem com os menores valores das métricas IoU e F1-Score.

Mesmo na imagem de menor desempenho, observa-se que a segmentação das uvas

Figura 14 – *Boxplot* das métricas de avaliação de segmentação de imagens no conjunto de teste.



Fonte: Autor.

foi satisfatória, com a maior parte das uvas segmentadas corretamente. Nota-se que a oclusão das uvas foi o principal fator que prejudicou a segmentação. Além disso, a presença de menos uvas na imagem inferida torna as métricas de segmentação mais sensíveis aos cachos não segmentados.

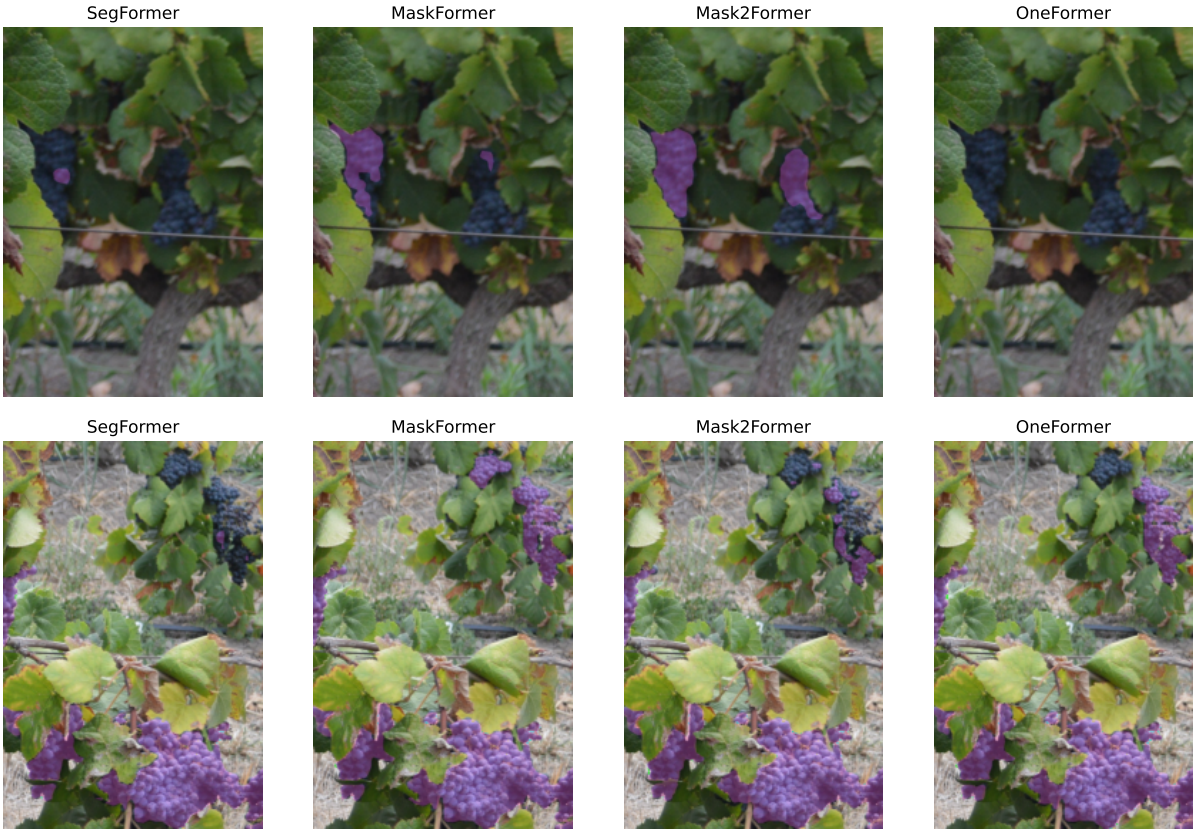
Estes resultados ressaltam a importância de um conjunto de dados bem distribuído e representativo para o treinamento eficaz dos modelos.

5.2.1.3 Contagem de *Pixels*

Adicionalmente, considerando que o atributo final para a estimativa da colheita é baseado na área da imagem composta por *pixels* de uvas, realizou-se a análise dos erros na contagem de *pixels*. É importante ressaltar que, anteriormente a esta etapa, a escolha do modelo foi realizada a partir da análise das métricas de segmentação na validação cruzada, sendo a contagem de *pixels* de caráter ilustrativo.

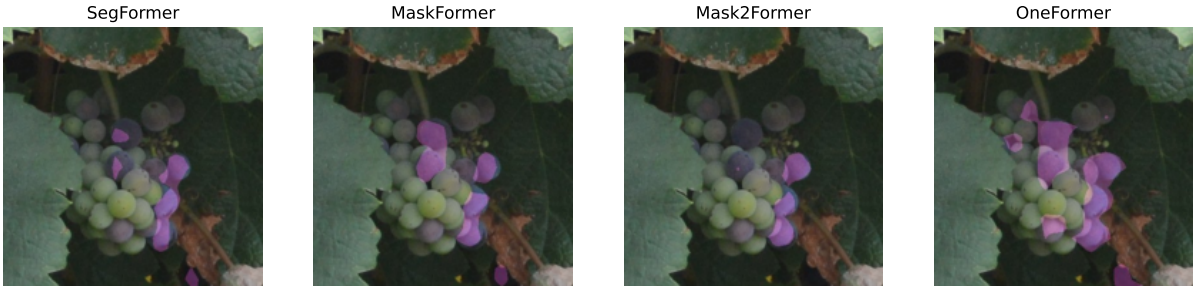
A análise das métricas de erro (MAE, MSE, RMSE, MAPE) das inferências dos modelos de segmentação em comparação com as máscaras de *ground truth* revelou os seguintes resultados:

Figura 15 – Exemplo de segmentação de uvas com fundo.



Fonte: Autor.

Figura 16 – Exemplo de segmentação de uvas verdes.



Fonte: Autor.

Tabela 7 – Média dos resultados das métricas de erro para a contagem de *pixels* de uvas no *dataset* TANNAT-23.

Modelo	**MAE**	**MSE**	**RMSE**	**MAPE** (%)
SegFormer	41.021,00	637.475.048,75	7.984,21	6,51
MaskFormer	56.198,75	2.983.378.922,38	17.272,46	12,21
Mask2Former	**34.294,25**	**3.236.009.952,00**	**17.988,91**	**4,31**
OneFormer	37.447,06	1.731.302.844,44	13.157,90	5,75

Fonte: Autor.

Figura 17 – Imagem com melhores métricas de segmentação no *dataset* TANNAT-23.

Fonte: UdelaR e autor.

Os resultados obtidos apontam que:

- **SegFormer** demonstrou um desempenho sólido, com o menor RMSE, indicando que possui menores grandes discrepâncias em suas previsões. Além disso, seu MAPE moderado sugere uma boa precisão relativa, tornando-o eficiente em situações onde grandes erros são menos toleráveis.
- **MaskFormer** apresentou um MSE mais alto do que SegFormer e OneFormer, mas não o maior entre os modelos analisados. Seu desempenho foi inferior em comparação com os outros modelos, com valores elevados de MAE e RMSE, indicando que suas previsões estão mais distantes da *ground truth* em média. O MAPE mais alto entre os modelos analisados confirma que possui o maior percentual de erro relativo.
- **Mask2Former** destacou-se com a menor média de erro absoluto (MAE), sugerindo

Figura 18 – Imagem com menores métricas de segmentação no *dataset* TANNAT-23.

Fonte: Fonte: UdelaR e autor.

que suas previsões são, em média, mais próximas da *ground truth*. O modelo também apresentou o menor MAPE, indicando a melhor precisão relativa. No entanto, teve um MSE elevado, o que pode apontar para algumas grandes discrepâncias em suas previsões.

- **OneFormer** apresentou métricas intermediárias em todas as avaliações, indicando um desempenho consistente e equilibrado. Seu MAPE relativamente baixo mostra que o modelo tem uma boa precisão relativa, mesmo não sendo o melhor em termos absolutos.

Portanto, a partir desta análise adicional percebe-se que o Mask2Former é o melhor modelo termos de precisão média absoluta (MAE) e precisão relativa (MAPE), apesar de seu MSE ser mais elevado, o que sugere algumas discrepâncias grandes. Mask2Former

foi escolhido como o modelo mais adequado para a segmentação de uvas no conjunto de dados TANNAT-24.

5.2.2 *Dataset* Definitivo (TANNAT-24)

Após o comparativo entre arquiteturas na seção anterior, o modelo foi treinado com o *dataset* definitivo e aferições foram realizadas para avaliar o desempenho do modelo.

5.2.2.1 Treinamento e Avaliação

O *dataset* TANNAT-24 é composto por 120 imagens anotadas pelo modelo ClickSEG na plataforma Supervisely. O Mask2Former foi treinado com uma taxa de aprendizado constante, *batch size* de 4, 30 épocas e *early stopping* com paciência de 8 épocas.

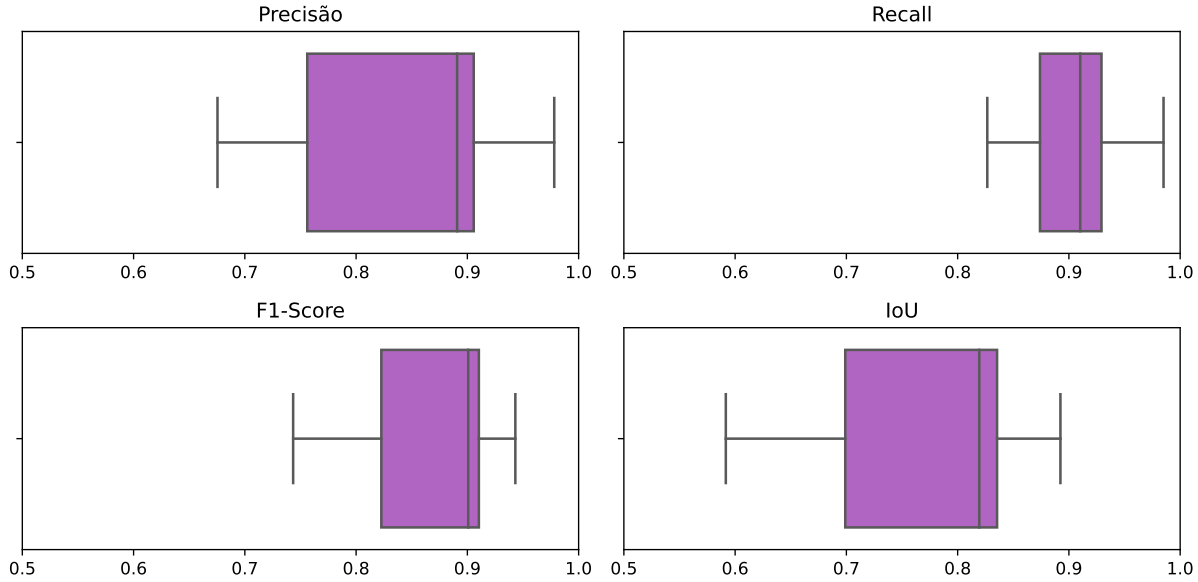
Os resultados das métricas de segmentação de imagens no *dataset* definitivo são apresentados na Tabela 9 e na Figura 19.

Tabela 8 – Média dos resultados das métricas de segmentação para a avaliação dos modelos no conjunto de teste do *dataset* TANNAT-24

Métricas / **Modelo**	**Precisão (%)**	**Recall (%)**	**F1-Score** (%)	**IoU** (%)
Mask2Former	84,46	90,35	87,03	77,53

Fonte: Autor.

Figura 19 – *Boxplot* das métricas de avaliação de segmentação de imagens no conjunto de teste de TANNAT-24.



Fonte: Autor.

Tabela 9 – Média dos resultados das métricas de erro para a contagem de *pixels* de uvas no *dataset* TANNAT-24.

Modelo	**MAE**	**MSE**	**RMSE**	**MAPE** (%)
Mask2Former	4.538,93	52.058.970,13	7.215,19	10,20

Fonte: Autor.

Ao analisar a imagem com menores valores de IoU e F1-Score após a inferência, observa-se que a segmentação das uvas foi bastante satisfatória, com uma segmentação precisa dos cachos de uvas. Ao comparar as previsões do Mask2Former com a *ground truth*, percebe-se que o baixo desempenho nesta imagem deveu-se à qualidade superior das máscaras geradas pelo Mask2Former em relação às próprias anotações, como pode ser observado na Figura 21.

A câmera deste *dataset* foi configurada para capturar as inflorescências de uvas, caracterizadas pela cor verde, durante a noite. Essa configuração foi mantida para capturar as imagens dos cachos de uvas, contribuindo para a forte oclusão das imagens e, consequentemente, aumentando significativamente a dificuldade das anotações.

A Tabela 9 apresenta as métricas de segmentação de imagens para o modelo Mask2Former no *dataset* definitivo TANNAT-24.

Embora a maioria das aferições dos modelos de segmentação semântica tenha sido realizada em imagens diurnas, observa-se que o desempenho manteve-se consistente ao aplicar as mesmas configurações de treinamento do modelo Mask2Former nas imagens noturnas.

Também é relevante destacar que as imagens do *dataset* TANNAT-24 foram coletadas na proporção 9:16, enquanto as imagens do *dataset* TANNAT-23 foram coletadas na proporção 4:3. Considerando que os modelos foram treinados com imagens na proporção 1:1, o redimensionamento das imagens provavelmente influenciou negativamente o desempenho do Mask2Former neste *dataset*. Caso fosse aplicado um corte nas partes superiores e/ou inferiores das imagens, onde não há uvas, o desempenho do modelo poderia ser melhorado.

Figura 20 – Imagem com melhores métricas de segmentação no *dataset* TANNAT-24.

Fonte: UdelaR e autor.

Figura 21 – Imagem com menores métricas de segmentação no *dataset* TANNAT-24.

Fonte: UdelaR e autor.

6 CONCLUSÃO

Neste trabalho, foi realizada uma revisão bibliográfica abrangente sobre técnicas de segmentação de imagens e os avanços na visão computacional desde a aplicação das arquiteturas baseadas em *Transformers*. A partir dessa revisão, foram treinados e avaliados quatro modelos de segmentação interativa de imagens: SAM, RITM, ClickSEG e FastSAM, utilizando uma combinação de conjuntos de dados produzida especificamente para este estudo.

Com base nos dados anotados, as arquiteturas de segmentação semântica foram comparadas utilizando métricas tradicionais de segmentação, sendo posteriormente aplicadas na contagem de *pixels* de uvas em imagens de *dataset* intermediário. O desempenho e o conhecimento adquiridos nestes experimentos foram fundamentais na escolha do modelo a ser utilizado na segmentação de um *dataset* final, destinado a um projeto de estimativa de colheitas de uvas.

Os resultados obtidos, aliados à bibliografia consultada, evidenciam avanços significativos nas tecnologias de segmentação de imagens, especialmente com a aplicação de modelos baseados em *Transformers* nos últimos anos. Embora as redes neurais convolucionais tenham perdido parte de sua relevância, a arquitetura dos modelos estudados ainda dependem de convoluções em algum nível.

As técnicas comtemporâneas de anotação de imagens democratizaram a criação de conjuntos de dados, permitindo que uma única pessoa, trabalhando com a ferramenta certa, possa anotar centenas de imagens em um único dia.

Apesar de ser o modelo interativo mais sofisticado estudado, o SAM demonstrou fraca capacidade no refinamento de máscaras a partir de múltiplos *prompts*. Ainda assim, mostrou-se muito eficiente em segmentar imagens com base em uma única *bounding box*, principalmente com a adição de um ponto como *prompt* adicional.

Em contrapartida, RITM e ClickSEG apresentaram uma capacidade extraordinária de adaptação aos *inputs* do usuário, eliminando a necessidade de correção manual das máscaras, desde que um número suficiente de cliques seja fornecido. A eficiência desses modelos decorre, em grande parte, da possibilidade de usar uma máscara prévia como entrada para cada inferência, permitindo até mesmo o trabalho com máscaras já anotadas por outras técnicas.

Em um contexto de anotação em escala industrial, uma abordagem eficiente poderia basear-se na segmentação inicial das imagens pelo SAM, com ajustes finais realizados por um modelo de segmentação mais refinado, como o RITM ou um de seus derivados.

Quanto ao *benchmarking* de arquiteturas de aprendizado profundo baseadas em *Transformers* na segmentação de cachos de uvas, embora por uma margem pequena, o Mask2Former destacou-se em relação aos outros, atingindo um F1-Score de 88,56% nas imagens diurnas e 87,03% nas noturnas.

A inclusão da *Focal Loss* no treinamento do SegFormer revelou-se poderosa, permitindo que um modelo mais simples e menos custoso computacionalmente atingisse resultados comparáveis aos de modelos muito mais complexos.

Nas imagens diurnas, os modelos demonstraram eficiência na compreensão do contexto das imagens durante o treinamento, embora ainda tenham capturado algumas uvas no fundo de fileiras adjacentes e não tenham segmentado uvas verdes, que foram anotadas. Foi possível perceber que a qualidade de segmentação obtida nas imagens diurnas se manteve nas imagens noturnas, apesar da forte oclusão causada pela escuridão.

Entretanto, ao comparar a soma das áreas segmentadas com as anotações de *ground truth*, evidencia-se um MAPE de 4,31% nas imagens diurnas e 10,20% no *dataset* noturno, o que indica que a precisão relativa das segmentações diminui significativamente em condições de pouca luz. Esse aumento no erro pode ser atribuído à dificuldade dos modelos em distinguir os cachos de uvas do fundo escuro, ressaltando a importância de melhorar as técnicas de anotação e segmentação para cenários noturnos. Ainda que QR Codes pendurados nos postes tenham sido confundidos com uvas, foi evidenciado que a capacidade do modelo de segmentar cachos de uvas pode superar a dos anotadores. Esse fato ressalta a importância de anotações precisas para o treinamento e avaliação de modelos de segmentação de imagens, melhorando a qualidade do treinamento e evitando equívocos na avaliação do desempenho do modelo.

Com um conjunto de dados maior e mais diversificado, boa parte dessas limitações poderia ser superada. Além disso, técnicas como *data augmentation* e *weight decay* poderiam reduzir o *overfitting* e melhorar a generalização dos modelos.

Embora este trabalho seja aplicado à viticultura, as técnicas e conceitos desenvolvidos podem ser facilmente aplicados a outras áreas de conhecimento, mesmo fora da agricultura de precisão.

Evidencia-se que uma rotina de *scheduler* polinomial, aplicada no treinamento original de todos os modelos, prejudica a performance do *fine-tuning* dos modelos, embora torne sua convergência mais estável. Possivelmente, uma rotina baseada em outro *scheduler* poderia apresentar resultados mais satisfatórios ao trabalhar com uma taxa de aprendizado constante.

Os algoritmos, a base de conhecimento e análises exploratórias desenvolvidas para este trabalho serão posteriormente aplicados no projeto, acelerando seu desenvolvimento e aprimorando a qualidade dos resultados.

REFERÊNCIAS

ARRILLAGA, Leandro *et al.* Response of Tannat (*Vitis vinifera* L.) to pre-flowering leaf removal in a humid climate. **OENO One**, v. 55, n. 2, p. 251–266, 2021. DOI: 10.20870/oeno-one.2021.55.2.4613. Disponível em: https://oeno-one.eu/article/view/4613. Acesso em: 3 jul. 2024.

BLEKOS, Achilleas *et al.* A Grape Dataset for Instance Segmentation and Maturity Estimation. **Agronomy**, v. 13, n. 8, 2023. ISSN 2073-4395. DOI: 10.3390/agronomy13081995. Disponível em: https://www.mdpi.com/2073-4395/13/8/1995.

CARION, Nicolas *et al.* **End-to-End Object Detection with Transformers**. [*S.l.*: *s.n.*], 2020. arXiv: 2005.12872 [cs.CV].

CHEN, Xi *et al.* Conditional Diffusion for Interactive Segmentation. *In:* 2021 IEEE/CVF International Conference on Computer Vision (ICCV). [*S.l.*: *s.n.*], 2021. P. 7325–7334. DOI: 10.1109/ICCV48922.2021.00725.

CHENG, Bowen; MISRA, Ishan *et al.* **Masked-attention Mask Transformer for Universal Image Segmentation**. [*S.l.*: *s.n.*], 2022. arXiv: 2112.01527 [cs.CV]. Disponível em: https://arxiv.org/abs/2112.01527.

CHENG, Bowen; SCHWING, Alexander G.; KIRILLOV, Alexander. **Per-Pixel Classification is Not All You Need for Semantic Segmentation**. [*S.l.*: *s.n.*], 2021. arXiv: 2107.06278 [cs.CV]. Disponível em: https://arxiv.org/abs/2107.06278.

COMMONS, Wikimedia. **Jaccard Index**. [*S.l.*: *s.n.*], 2023. Accessed: 2024-06-21, licensed under Creative Commons CC0 1.0 Universal (Public Domain Dedication). Disponível em: https://en.wikipedia.org/wiki/Jaccard_index.

DENG, Jie *et al.* Assessing Macro Disease Index of Wheat Stripe Rust Based on Segformer with Complex Background in the Field. **Sensors**, v. 22, n. 15, 2022. ISSN 1424-8220. DOI: 10.3390/s22155676. Disponível em: https://www.mdpi.com/1424-8220/22/15/5676.

DOSOVITSKIY, Alexey *et al.* **An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale**. [*S.l.*: *s.n.*], 2021. arXiv: 2010.11929 [cs.CV]. Disponível em: https://arxiv.org/abs/2010.11929.

DUARTE ALONSO, Abel. Collaboration as a means to stand out in a competitive market: An exploratory study of Uruguay's exporting wineries. **Journal for International Business and Entrepreneurship Development**, InderScience Publishers, v. 9, n. 4, p. 359–378, out. 2016. DOI: 10.1504/JIBED.2016.10000708. Disponível em: http://researchonline.ljmu.ac.uk/id/eprint/3833/.

DUARTE ALONSO, Abel. Tannat: the positioning of a wine grape as symbol and 'referent' of a nation's gastronomic heritage. **Journal of Heritage Tourism**, Routledge, v. 8, n. 2-3, p. 105–119, 2013. DOI: 10.1080/1743873X.2013.767806. Disponível em: https://doi.org/10.1080/1743873X.2013.767806.

GONZÁLEZ-NEVES, G. *et al.* Anthocyanic composition of Tannat grapes from the South region of Uruguay. **Analytica Chimica Acta**, v. 513, n. 1, p. 197–202, 2004.

HASSANI, Ali; SHI, Humphrey. **Dilated Neighborhood Attention Transformer**. [*S.l.*: *s.n.*], 2023. arXiv: 2209.15001 [cs.CV].

HE, Kaiming; GKIOXARI, Georgia *et al.* **Mask R-CNN**. [*S.l.*: *s.n.*], 2017. arXiv: 1703.06870 [cs.CV].

HE, Kaiming; ZHANG, Xiangyu *et al.* Deep Residual Learning for Image Recognition. *In:* 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). [*S.l.*: *s.n.*], 2016. P. 770–778. DOI: 10.1109/CVPR.2016.90.

HEARTEX. **Label Studio: Open Source Data Labeling Tool**. [*S.l.*], 2024. Accessed: 2024-06-17.

INAVI. **Estadísticas de Viñedos 2023: datos nacionales**. Canelones: [*s.n.*], 2023. https://www.inavi.com.uy/uploads/vinedo/b001699f9585532cfd1fa06a35756544f917eda5.pdf. 73p.

JAIN, Jitesh *et al.* **OneFormer: One Transformer to Rule Universal Image Segmentation**. [*S.l.*: *s.n.*], 2022. arXiv: 2211.06220 [cs.CV]. Disponível em: https://arxiv.org/abs/2211.06220.

JANG, Won-Dong; KIM, Chang-Su. **Interactive Image Segmentation via Backpropagating Refinement Scheme**. [*S.l.*: *s.n.*], 2019. P. 5292–5301. DOI: 10.1109/CVPR.2019.00544.

JOCHER, Glenn. **YOLOv5 by Ultralytics**. [*S.l.*: *s.n.*]. DOI: 10.5281/zenodo.3908559. Disponível em: https://github.com/ultralytics/yolov5.

JOCHER, Glenn; CHAURASIA, Ayush; QIU, Jing. **Yolo by ultralytics**. [*S.l.*: *s.n.*], 2023. Acesso em: 14 maio 2024. Disponível em: https://github.com/ultralytics/ultralytics.

KHOROSHEVSKY, Faina; KHOROSHEVSKY, Stanislav; BAR-HILLEL, Aharon. Parts-per-Object Count in Agricultural Images: Solving Phenotyping Problems via a Single Deep Neural Network. **Remote Sensing**, v. 13, n. 13, 2021. ISSN 2072-4292. DOI: 10.3390/rs13132496. Disponível em: https://www.mdpi.com/2072-4292/13/13/2496.

KINGMA, Diederik P.; BA, Jimmy. **Adam: A Method for Stochastic Optimization**. [*S.l.*: *s.n.*], 2017. arXiv: 1412.6980 [cs.LG]. Disponível em: https://arxiv.org/abs/1412.6980.

KIRILLOV, Alexander; HE, Kaiming *et al.* **Panoptic Segmentation**. [*S.l.*: *s.n.*], 2019. arXiv: 1801.00868 [cs.CV].

KIRILLOV, Alexander; MINTUN, Eric *et al.* Segment Anything. **arXiv preprint arXiv:2304.02643**, 2023. Disponível em: https://arxiv.org/abs/2304.02643.

LI, Xiangtai *et al.* Sfnet: Faster and Accurate Semantic Segmentation Via Semantic Flow. **International Journal of Computer Vision**, Springer Science e Business Media LLC, v. 132, n. 2, p. 466–489, set. 2023. ISSN 1573-1405. DOI: 10.1007/s11263-023-01875-x. Disponível em: http://dx.doi.org/10.1007/s11263-023-01875-x.

LIANG, Chang-Mei *et al.* Segmentation and weight prediction of grape ear based on SFNet-ResNet18. **Systems Science & Control Engineering**, Taylor & Francis, v. 10, n. 1, p. 722–732, 2022. DOI: 10.1080/21642583.2022.2110541. eprint: https://doi.org/10.1080/21642583.2022.2110541. Disponível em: https://doi.org/10.1080/21642583.2022.2110541.

LIN, Tsung-Yi *et al.* **Focal Loss for Dense Object Detection**. [*S.l.*: *s.n.*], 2017. arXiv: 1708.02002 [cs.CV].

LIU, Qin *et al.* **SimpleClick: Interactive Image Segmentation with Simple Vision Transformers**. [*S.l.*: *s.n.*], 2023. arXiv: 2210.11006 [cs.CV]. Disponível em: https://arxiv.org/abs/2210.11006.

LIU, Ze *et al.* **Swin Transformer: Hierarchical Vision Transformer using Shifted Windows**. [*S.l.*: *s.n.*], 2021. arXiv: 2103.14030 [cs.CV].

LOSHCHILOV, Ilya; HUTTER, Frank. **Decoupled Weight Decay Regularization**. [*S.l.*: *s.n.*], 2017. arXiv: 1711.05101 [cs.LG].

MOSKALENKO, Andrey *et al.* **TETRIS: Towards Exploring the Robustness of Interactive Segmentation**. v. 38. [*S.l.*]: Association for the Advancement of Artificial Intelligence (AAAI), mar. 2024. P. 4287–4295. DOI: 10.1609/aaai.v38i5.28225. Disponível em: http://dx.doi.org/10.1609/aaai.v38i5.28225.

OLENSKYJ, Alexander G. *et al.* **End-to-end deep learning for directly estimating grape yield from ground-based imagery**. [*S.l.*: *s.n.*], 2022. Disponível em: https://arxiv.org/abs/2208.02394.

OPENMMLAB. **Label Anything: An Interactive Semi-Automatic Annotation Tool**. [*S.l.*: *s.n.*], 2024. https://github.com/open-mmlab/playground/blob/main/label_anything/readme.md. Accessed: 2024-06-17.

SANTOS, Thiago T. **WGISD: Wine Grape Image Segmentation Dataset**. [*S.l.*: *s.n.*], 2021. https://github.com/thsant/wgisd/. Accessed: 2024-06-16.

SANTOS, Thiago T.; BARBEDO, Jayme G. A. *et al.* Computer vision applied to agriculture. *In:* MASSRUHÁ, Silvia *et al.* (Ed.). **Digital agriculture: research, development and innovation in production chains**. Brasília, DF: Embrapa, 2023. cap. 6, p. 109–123. ISBN 978-65-89957-72-0. Disponível em: http://www.alice.cnptia.embrapa.br/alice/handle/doc/1156749.

SANTOS, Thiago T.; SOUZA, Leonardo L. de *et al.* Grape detection, segmentation, and tracking using deep neural networks and three-dimensional association. **Computers and Electronics in Agriculture**, Elsevier BV, v. 170, p. 105247, mar. 2020. ISSN 0168-1699. DOI: 10.1016/j.compag.2020.105247. Disponível em: http://dx.doi.org/10.1016/J.COMPAG.2020.105247.

SOFIIUK, Konstantin; PETROV, Ilia; BARINOVA, Olga *et al.* **F-BRS: Rethinking Backpropagating Refinement for Interactive Segmentation**. [*S.l.*]: IEEE, jun. 2020. DOI: 10.1109/cvpr42600.2020.00865. Disponível em: http://dx.doi.org/10.1109/cvpr42600.2020.00865.

SOFIIUK, Konstantin; PETROV, Ilia A.; KONUSHIN, Anton. **Reviving Iterative Training with Mask Guidance for Interactive Segmentation**. [*S.l.*: *s.n.*], 2021. arXiv: 2102.06583 [cs.CV]. Disponível em: https://arxiv.org/abs/2102.06583.

SUN, Shoukun *et al.* **CFR-ICL: Cascade-Forward Refinement with Iterative Click Loss for Interactive Image Segmentation**. [*S.l.*: *s.n.*], 2024. arXiv: 2303.05620 [cs.CV].

TEAM, CVAT. **CVAT: Computer Vision Annotation Tool**. [*S.l.*], 2024. Accessed: 2024-06-17.

TEAM, Labelbox. **Labelbox: The Leading Training Data Platform for AI**. [*S.l.*], 2024. Accessed: 2024-06-17.

TEAM, Roboflow. **Roboflow: End-to-End Computer Vision Platform**. [*S.l.*], 2024. Accessed: 2024-06-17.

TEAM, Supervisely. **Supervisely: End-to-End Computer Vision Platform**. [*S.l.*], 2024. Accessed: 2024-06-17.

VASWANI, Ashish *et al.* **Attention Is All You Need**. [*S.l.*: *s.n.*], 2017. arXiv: 1706.03762 [cs.CL].

WANG, Wenhai *et al.* **Pyramid Vision Transformer: A Versatile Backbone for Dense Prediction without Convolutions**. [*S.l.*: *s.n.*], 2021. arXiv: 2102.12122 [cs.CV].

WANG, Zhe *et al.* **Interactive segmentation in aerial images: a new benchmark and an open access web-based tool**. [*S.l.*: *s.n.*], 2024. Disponível em: https://arxiv.org/abs/2308.13174.

XIE, Enze *et al.* **SegFormer: Simple and Efficient Design for Semantic Segmentation with Transformers**. [*S.l.*: *s.n.*], 2021. arXiv: 2105.15203 [cs.CV]. Disponível em: https://arxiv.org/abs/2105.15203.

XU, Ning *et al.* FocalClick: Towards Practical Interactive Image Segmentation. *In:* PROCEEDINGS of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). [*S.l.*]: IEEE, 2022. P. 16453–16462. Disponível em: https://arxiv.org/abs/2204.02574.

ZHAO, Xu *et al.* Fast Segment Anything. **arXiv**, abs/2306.12156, jun. 2023. Disponível em: https://arxiv.org/abs/2306.12156.

ZHENG, Sixiao *et al.* **Rethinking Semantic Segmentation from a Sequence-to-Sequence Perspective with Transformers**. [*S.l.*: *s.n.*], 2020. arXiv: 2012.15840 [cs.CV].