Aprendizagem Profunda - Módulo 2

PG55974: Leonardo Barroso, PG55948: Hugo Ramos, PG55951: João Vale, PG55977: Luís Borges, and PG56015: Tomás Oliveira

Universidade do Minho, Portugal

Abstract. Existe uma necessidade crescente de automatizar o processo de avaliação de procedimentos médicos, de forma a melhorar tanto a sua qualidade quanto a sua segurança. No contexto do desafio Open Suturing Skills de 2025, foram desenvolvidas propostas de solução para as três tarefas apresentadas: (1) previsão do desempenho geral do procedimento (GRS), (2) previsão dos oito critérios OSATS, e (3) tracking dos pontos-chave das mãos e das ferramentas cirúrgicas. Para cada uma destas tarefas, foi proposto uma arquitetura, com foco em modelos baseados em redes neuronais profundas, incluindo CNNs, LSTMs e Transformers. As soluções para as tarefas 1 e 2 foram totalmente implementadas, treinadas e avaliadas com base nos dados fornecidos pela organização. Para a tarefa 3, foi feito o estudo preliminar e definidas as abordagens mais promissoras. Os resultados obtidos demonstram o potencial das técnicas de Deep learning na avaliação objetiva e automatizada de competências técnicas em contexto cirúrgico.

Keywords: OSATS · Deep learning · GRS · Tracking · Cirurgical Video

1 Introdução

A avaliação objetiva do desempenho técnico em procedimentos cirúrgicos é um elemento fundamental para a formação de profissionais de saúde, contribuindo diretamente para a melhoria da qualidade e segurança do ato cirúrgico.

No entanto, os métodos tradicionais de avaliação, como o OSATS (Objective Structured Assessment of Technical Skills) e o GRS (Global Rating Score), baseiam-se em observações manuais que estão sujeitas à variabilidade do avaliador e são logisticamente difíceis de aplicar em larga escala.

Com os avanços recentes em Deep learning, tem-se vindo a explorar a sua aplicação em contextos médicos como forma de automatizar a avaliação de competências técnicas. Modelos como redes convolucionais (CNNs), redes recorrentes (LSTM) e arquiteturas baseadas em transformers têm demonstrado elevada eficácia na análise de vídeos, permitindo identificar fases do procedimento, padrões de movimento e manipulação de instrumentos com elevada precisão. Estas abordagens oferecem uma alternativa promissora à avaliação manual, possibilitando uma análise mais objetiva, escalável e menos dependente da variabilidade humana.

Neste trabalho, pretende-se desenvolver um sistema de análise automática de vídeos de suturas médicas com recurso a modelos de Deep learning, com o

objetivo de avaliar o desempenho técnico dos profissionais de saúde de forma objetiva e eficiente.

O Desafio Open Suturing Skills surge da necessidade de tornar os processos de avaliação mais objetivos, reprodutíveis e escaláveis, reduzindo a dependência de avaliadores humanos e promovendo uma formação médica mais eficiente e baseada em dados.

Para alcançar esse objetivo, serão propostas e avaliadas diferentes arquiteturas de Deep learning adaptadas a cada uma das tarefas, combinando modelos como CNNs e Vision Transformers com modelos temporais. A eficácia das abordagens será analisada com base nas métricas definidas pelo desafio, utilizando os dados anotados fornecidos.

Apenas se implementou as soluções para as primeiras suas tarefas devido à falta de dados para a terceira.

2 Estado da Arte

A avaliação de competências técnicas em contexto médico é essencial para garantir a segurança do paciente e otimizar a formação médica. Métodos tradicionais como o OSATS ou o GRS, têm sido amplamente utilizados para este fim, mas são limitados pela sua natureza subjetiva e pela dependência de avaliadores humanos experientes [3].

Nos últimos anos, abordagens baseadas em Deep learning têm emergido como soluções promissoras para automatizar esta avaliação, explorando dados provenientes de diferentes fontes, como vídeos, imagens do produto final e dados cinemáticos. [8]

Em [2], os autores propõem um sistema de reconhecimento automático de fases cirúrgicas com base em vídeos laparoscópicos. O modelo combina uma rede convolucional (CNN) com uma cascata de três redes LSTM, explorando tanto o conteúdo visual como as dependências temporais dos procedimentos. Os resultados obtidos no Dataset Cholec80 demonstram um desempenho elevado na tarefa de reconhecimento de fases, evidenciando a importância da componete tanto espacial, quanto temporal para a compreensão do fluxo cirúrgico.

No trabalho de Khalid et al. [3], são explorados modelos de Deep learning aplicados à classificação de ações cirúrgicas e avaliação de desempenho em vídeos curtos. Através de representações compactas de autoencoders e segmentação semântica, os autores conseguem prever com elevada precisão tanto a ação realizada como o nível de competência do cirurgião, com base apenas no vídeo. Os modelos obtêm valores médios de precisão e recall superiores a 90% na identificação das ações cirúrgicas.

Kasa et al. apresentam uma abordagem multimodal em [4, 5], onde são combinados dados de vídeo, imagem e movimento das mãos capturados por sensores. Os autores desenvolveram três modelos: um baseado em imagem (ResNet-50), outro baseado em dados cinemáticos (ResNet-18 + LSTM) e um modelo multimodal. Este último apresenta os melhores resultados, com desempenhos comparáveis aos avaliadores humanos em múltiplos domínios da escala OSATS. Esta

investigação valida o potencial da fusão multimodal na avaliação técnica automatizada, evidenciando uma correlação Spearman de 0.67 com as avaliações humanas e um ICC até 0.90 para certas dimensões.

Em relação a modelos de tracking, seja de bounding boxes ou keypoints, os métodos mais recentes baseiam-se fortemente em modelos de segmentação seguidos de localização precisa dos pontos de interesse. Ghanekar et al. propõem um método que segmenta regiões de interesse correspondentes aos keypoints e calcula os centróides dessas regiões como estimativas da sua localização. O modelo explora o contexto temporal recorrendo a múltiplos frames consecutivos e complementa os dados com mapas de fluxo ótico e profundidade estimada, melhorando a robustez face a oclusões, desfoque e variações na orientação das ferramentas. O uso de modelos como o DeepLabv3 combinado com MFCNet resultou numa redução significativa do erro médio quadrático (RMSE) e numa precisão de deteção superior a 90% em datasets como o EndoVis e o JIGSAWS [9].

Adicionalmente, uma revisão sistemática conduzida por Yangi et al. demonstra a aplicabilidade crescente destas abordagens em diversos contextos cirúrgicos, destacando a preferência por técnicas baseadas em CNNs, LSTMs e Transformers, bem como o uso de sensores visuais simples, como vídeos gravados, para rastrear o movimento de mãos e instrumentos com elevada precisão e aplicabilidade no treino e avaliação cirúrgica [8].

2.1 Análise Crítica

A literatura destaca o forte potencial das abordagens de Deep learning na avaliação técnica automatizada, com resultados promissores em tarefas como a sutura cirúrgica. Contudo, nota-se que muitos estudos foram conduzidos em contextos controlados, com dados limitados, o que pode comprometer a sua generalização. A abordagem multimodal destaca-se na literatura como uma das estratégias mais eficazes para capturar diferentes dimensões do desempenho técnico. Entre as arquiteturas mais eficazes, destaca-se a combinação de CNNs com LSTMs para captar padrões visuais e temporais, bem como o uso de mecanismos de atenção para reforçar a relevância contextual. Por fim, são referidos modelos como DeepLabv3+MFCNet e SegFormer como opções robustas para tracking, sendo considerados para a tarefa 3 [9].

3 Metodologia aplicada

A abordagem escolhida será a preparação de uma pipeline para englobar todo o processo desde o processamento dos dados até à exposição dos resultados obtidos. Através das palavras-chave apresentadas, e outros termos relacionados, serão estudadas as melhores arquiteturas e soluções para o problema em questão. Serão depois aplicados no nosso contexto para verificar e realizar conclusões próprias sobre o tema. Devido à falta de dados sobre a tarefa 3 não foi possivel realizar a implementação da tarefa proposta, assim como desenvolver uma abordagem baseada na interligação entre as tarefas. Abaixo, neste relatório será explorado essa ligação de forma conceptual.

4 Análise e tratamento dos dados

Foram disponibilizados vídeos de treino de suturas cirúrgicas, assim como dados pré-processados pelo docente (3 preparações diferentes). Utilizou-se todos estes datasets assim como foi realizada um processamento próprio.

4.1 Análise dos dados

Cada vídeo no conjunto de dados possui três entradas distintas no ficheiro OSATS.csv, correspondentes a diferentes avaliações realizadas por especialistas. Para consolidar esta informação, optou-se por calcular a média ponderada das avaliações. No caso do GRS, os valores contínuos foram posteriormente agrupados em classes discretas, como o pedido pelo desafio. Já os valores dos critérios OSATS variam entre 1 e 5 mas, dado que alguns modelos requerem classes a partir de zero, foi aplicada a transformação adequada.

Durante a análise exploratória dos dados, verificou-se um forte desbalanceamento entre classes, com algumas categorias completamente ausentes como mostra a fig 1 do anexo. Esta assimetria deve ser tida em conta na escolha da função de perda e na avaliação dos modelos.

Em relação ao conteúdo visual, observou-se que os elementos mais relevantes para a tarefa — como as mãos, a sutura e as ferramentas — tendem a concentrar-se no centro da imagem. Por esse motivo, foi aplicada uma operação de center crop aos vídeos, testando-se vários tamanhos (fig 2). Concluiu-se que a dimensão 896x896 oferecia o melhor compromisso entre foco na região de interesse e preservação do contexto visual necessário à tarefa.

Relativamente aos processamentos disponibilizados pelo docente temos três preparações distintas.

Num dos formatos, o vídeo é representado por um tensor com forma (300, 384, 20, 20), correspondendo a 300 frames, cada um com 384 feature maps de dimensão 20×20. Esta estrutura sugere que os dados já passaram por uma ou mais camadas convolucionais.

Noutro formato, o ficheiro contém um vetor plano com aproximadamente 140 milhões de valores do tipo float64, acompanhado de um rótulo de classificação. A partir da dimensão total e do número de frames (900), estima-se que cada frame seja descrito por 153.600 valores, o que poderá corresponder a 384 * 20 * 20 canais, mantendo assim coerência com a estrutura anterior, embora não haja meta-informação sobre a resolução ou pré-processamento.

Por fim, outro tipo de representação encontrada corresponde a arrays de forma (300, 300), em que cada linha representa um frame com 300 características. Este formato indica a utilização de embeddings por frame, possivelmente extraídos por uma rede já treinada (como uma CNN ou MLP).

4.2 Processamento realizado

Para a extração de features visuais, optou-se por utilizar uma rede convolucional pré-treinada ResNet50, mantendo-se a arquitetura original. A saída gerada por

esta rede corresponde a um vetor de 2048 características por frame. Como a ResNet50 requer imagens de entrada com dimensão 224×224, todos os frames foram previamente redimensionados para essa resolução.

Dado o elevado número de frames por vídeo, mesmo após amostragem a 3 FPS, tornou-se inviável processar a totalidade das sequências com os recursos computacionais disponíveis. Para mitigar este problema, aplicou-se um processo de redução de frames com base em técnicas de clustering. Especificamente, recorreu-se ao algoritmo K-Means para identificar os 300 frames mais representativos de cada vídeo. Esta abordagem permitiu selecionar os frames mais informativos com base na variabilidade das suas features, reduzindo a redundância temporal. Após a seleção, garantiu-se a preservação da ordem temporal original, de forma a manter a coerência sequencial necessária para modelos temporais.

Os dados finais, compostos pelos vetores de features e pelas respetivas labels da Task 1 e Task 2, foram guardados em ficheiros NumPy, facilitando o carregamento e processamento durante a fase de treino dos modelos.

5 Tarefa 1

A Tarefa 1 consiste na previsão do desempenho geral de cada vídeo, tratando-se de um problema de classificação baseada na análise sequencial do procedimento. Dada a semelhança com trabalhos existentes na literatura, tanto ao nível dos dados (tipo e contexto) como da própria tarefa de avaliação global, decidiu-se adaptar uma arquitetura previamente proposta no artigo [2].

O modelo adotado combina uma camada convolucional inicial para extração ou refinamento de padrões com três camadas LSTM dispostas em cascata, permitindo uma modelação progressiva e hierárquica da informação temporal ao longo do vídeo. A implementação foi testada com dois formatos mencionados anteriormente: features comprimidas por frame com forma (300, 300) (GRS01) e representações espaciais completas com forma (300, 384, 20, 20)(GRS02). Devido a isso foram implementados dois codificadores diferentes.

5.1 Arquitetura

Encoder CNN 2D (para entradas de forma $(300, 384, 20, 20)$): O encoder CNN 2D, é composto por duas camadas Conv2d (384→128 e 128→64), cada uma seguida de BatchNorm2d e ReLU. Em seguida, aplica-se AdaptiveAvgPool2d((1, 1)) e Flatten, resultando numa saída final de forma (B, 64).

Encoder 1D (para entradas de forma (300, 300)) Consiste em duas camadas Conv1d (300→128 e 128→64) com kernel 3 e padding 1, cada uma seguida de uma ReLU.

LSTM-clip Corresponde à primeira LSTM da arquitetura, responsável por processar padrões temporais de curto alcance. A sua saída tem a forma $(B, T, hidden\_clip)$.

LSTM-video Esta segunda LSTM é encarregue de captar dependências temporais globais ao longo de toda a sequência de vídeo. A saída mantém a estrutura sequencial, com forma $(B, T, hidden\_video)$.

LSTM-final A terceira LSTM atua como etapa de refinação da representação temporal. Apenas a saída do último *timestep* é retida, resultando numa saída de forma $(B, hidden\_final)$.

Camada Totalmente Ligada Responsável por mapear a representação final extraída pela LSTM para o espaço das classes do GRS. A saída final tem a forma $(B, num\_classes)$.

Formato de Saída O modelo produz *logits* de classificação correspondentes às quatro categorias do GRS.

5.2 Resultados Obtidos

Ambos os modelos foram treinados com um tamanho de batch de 16, um learning rate de 0.001 e com a função de loss sendo Cross entropy. Testou se um intervalo de valores para o número de ephochs (10-50) e os resultados mais satisfatórios foram 20 ephochs para o GRS02 e 30 para o GRS01. Foram também atribuídos pesos na função de loss para ressaltar as classes minoritárias.

Os gráficos de loss referentes às tarefas GRS01 e GRS02 apresentam comportamentos distintos (anexo fig 3). No caso do GRS01, observa-se uma evolução estável com baixa variância entre treino e validação, o que indica uma generalização aceitável, embora com pouca capacidade de aprendizagem, dado o estagnamento da loss. Já o gráfico do GRS02 revela sinais de overfitting: a loss de treino diminui de forma consistente, mas a de validação aumenta progressivamente, o que sugere que o modelo está a memorizar os dados de treino em vez de aprender padrões generalizáveis.

Apesar disso, os resultados apresentados na Tabela 1 mostram um desempenho superior na versão GRS02 em ambas as métricas utilizadas. Isto sugere que, mesmo com overfitting, o modelo conseguiu capturar melhor os padrões relevantes presentes nos dados. Enquanto o GRS01 se limitou a prever essencialmente as duas classes mais frequentes, o GRS02 foi capaz de distinguir um maior número de categorias, contribuindo para um F1-score mais elevado e menor custo esperado.

GRS	F1-score (macro)	Expected Cost
01	0.1682	1.0806
02	0.4124	0.5161

Table 1. Resultados obtidos para as duas versões do GRS.

6 Tarefa 2

A Tarefa 2 consiste na previsão dos oito critérios da escala OSATS, tratando-se de um problema de classificação do tipo multilabel. A literatura recente aponta os mecanismos de atenção como abordagens promissoras para este tipo de tarefa, dado o seu potencial para focar dinamicamente nas partes mais relevantes da sequência de entrada.

Para esta tarefa, optou-se pela utilização de uma arquitetura baseada em BiLSTM, uma vez que este tipo de rede permite capturar dependências temporais em ambas as direções, o que é particularmente relevante para vídeos de procedimentos cirúrgicos, onde ações podem depender de contextos anteriores e subsequentes. Esta bidirecionalidade permite enriquecer a representação temporal dos dados, contribuindo para uma avaliação mais precisa de cada critério OSATS.

Os dados utilizados foram obtidos a partir do processamento realizado.

Paralelamente, foi implementada uma versão baseada em Vision Transformer (ViT), dada a sua capacidade de modelar relações globais entre frames. No entanto, mesmo após diversas alterações nos hiperparâmetros, arquitetura e função de loss, o ViT revelou-se incapaz de ultrapassar o problema das classes desbalanceadas, prevendo exclusivamente a classe maioritária em todos os casos.

Por este motivo, apesar de estar documentado no repositório, este modelo foi desconsiderado nos resultados apresentados.

6.1 Arquitetura

Input: Tem como entrada um tensor [B, 300, 2048], onde B representa o tamanho do batch e 2048 corresponde às features por frame, extraídas previamente durante a fase de preparação dos dados.

BiLSTM: BiLSTM com duas camadas e dimensão oculta 512 por direção. Esta configuração permite capturar dependências temporais em ambas as direções. A saída tem a forma [B, T, 1024].

Atenção Temporal: Aplica-se um mecanismo de atenção sobre a sequência codificada, agregando a informação relevante num vetor de contexto único com forma [B, 1024]. Este vetor foca-se nos frames mais informativos para a tarefa de predição.

Dropout: É aplicado dropout ao vetor de contexto, com uma taxa entre 0.1 e 0.3, com o objetivo de melhorar a generalização, no final, optou-se por escolher 0.3.

Saída: O vetor de contexto é passado por 8 cabeças paralelas, cada uma composta por uma camada linear que mapeia 1024 para 5 classes. Cada cabeça prevê a distribuição de classes (0–4) para um critério OSATS. A saída final tem a forma [B, 8, 5].

6.2 Resultados

Utilizou-se Focal Loss em substituição da CrossEntropyLoss tradicional, permitindo dar maior ênfase a exemplos difíceis e menos representados. Para reforçar esse efeito, foram calculados pesos de classe específicos por critério OSATS. O valor inicial da learning rate foi definido como 0.0005, inferior ao habitual, dada a sensibilidade dos mecanismos de atenção. Foram utilizadas 100 ephochs em conjunto com early stopping para maximizar a generalização sem overfitting.

O gráfico de loss (anexo fig 4) revela um padrão clássico de overfitting, apesar de mais leve em relação à tarefa anterior. Seria recomendável aplicar técnicas adicionais, como reduzir a patience no early stop ou explorar data augmentation para mitigar este comportamento.

OSATS	**F1-Score**	**Expected Cost**
Respect	0.404	0.532
Motion	0.429	0.532
Instrument	0.5369	0.452
Suture	0.344	0.677
Flow	0.544	0.484
Knowledge	0.405	0.5968
Performance	0.379	0.5968
Final	0.402	0.597

Table 2. Resultados por critério OSATS

Os resultados da Tabela 2 mostram que os critérios Flow e Instrument obtiveram os melhores F1-Scores, indicando maior previsibilidade. Por outro lado, Suture e Performance apresentaram os piores resultados, com F1-Scores baixos e custos esperados elevados, o que sugere maior dificuldade na sua classificação.

7 Task 3: Tracking de pontos das mãos e objetos cirúrgicos

Segundo as orientações do entidade organizadora, os dados que serão fornecidos para esta tarefa consistem em 1 frame por vídeo (amostragem de 1Fpm em clipes de 1 minuto), a sua respetiva máscara de segmentação e anotações dos pontos chave.

A solução proposta para esta tarefa é inspirada em [9] devido a ser uma abordagem para detetar keypoints de ferramentas em contextos cirúrgicos. Apesar de ser direcionada a contextos multiframe, consegue-se facilmente adaptar para single frame.

A arquitetura divide-se em duas etapas fundamentais: um modelo de segmentação é aplicado para gerar mapas de regiões de interesse (blobs) que correspondem às localizações esperadas dos keypoints dos instrumentos; numa segunda fase, calcula-se o centróide de cada região segmentada, obtendo-se assim a estimativa final da posição dos keypoints.

Primeira etapa: Utilizou-se o modelo SegFormer, que combina CNNs e Transformers para gerar máscaras segmentadas com classes de 0 a 5. Esta arquitetura foi escolhida pela sua eficácia comprovada em tarefas de segmentação com elevada variação espacial e contextual, como é o caso dos vídeos cirúrgicos. O treino é feito com pares de frames e máscaras anotadas, usando Cross Entropy como função de perda. Técnicas de data augmentation são recomendadas para melhorar a robustez e generalização.

Segunda etapa: Extrai-se as coordenadas dos keypoints a partir das máscaras segmentadas pelo SegFormer. Para cada classe, são detetados blobs com OpenCV e calculado o centróide do maior blob, assumindo que este representa a localização mais fiável do ponto-chave no frame.

Treino: Baseia-se na minimização do erro quadrático médio (MSE) entre os keypoints preditos e as coordenadas reais. Apenas os pontos com visibilidade igual a 2 são considerados, filtrando automaticamente casos dos pontos escondidos ou fora do quadro. Desta forma, o modelo foca-se na aprendizagem de localizações fiáveis, ajustando os pesos para melhorar a precisão na extração dos keypoints.

Output: Assume-se que o formato de output para o tracking de keypoints seguirá uma estrutura semelhante a este formato:

```
<frame>, <id>, <kp_id>, <x>, <y>, <visibility>
```

adaptada do MOTChallenge, onde se representam as coordenadas 2D e a visibilidade de cada ponto por frame e objeto.

8 Interligação entre tarefas

A implementação dos modelos assim como os seus respetivos testes foram condicionados pela falta de informação da tarefa 3, tarefa esta extremamente relevante para a ligação entre as 3 tarefas.

Uma ideia explorada para enriquecer o processo de avaliação geral consiste em utilizar os resultados da tarefa de tracking (Task 3) como fonte adicional de informação para as restantes tarefas. Para isso, os frames já amostrados e recortados seriam passados pelo modelo de tracking, permitindo obter as coordenadas dos keypoints mais relevantes. Com base nessas posições, seria gerado um heatmap por frame, destacando visualmente as regiões de maior importância.

Esta representação fornece contexto espacial explícito sobre os pontos estratégicos envolvidos no procedimento, o que é particularmente útil para avaliação técnica. A seguir, foi realizada a mesma preparação de dados da Task 2, mas agora com os frames enriquecidos com os heatmaps. O modelo de OSATS foi então treinado com esta nova entrada, mais informativa.

Por fim, os resultados deste modelo, refletindo critérios técnicos detalhado, seriam utilizados como features auxiliares para apoiar a previsão do desempenho global na Task 1.

9 Conclusão e Trabalho Futuro

Em conclusão, este trabalho propôs soluções para as três tarefas do desafio Open Suturing Skills 2025, tendo sido implementadas e testadas as duas primeiras com os dados disponíveis, previamente tratados. As soluções desenvolvidas foram inspiradas em arquiteturas reconhecidas na literatura, adaptadas ao contexto específico da avaliação de competências técnicas em sutura.Para além da implementação, foi também apresentada uma proposta conceptual de interligação entre os modelos das diferentes tarefas, com o objetivo de reforçar a robustez e consistência dos resultados obtidos.

Como trabalho futuro, destaca-se a implementação da tarefa 3, com a possibilidade de a expandir para um cenário multiframe caso existam dados adequados. Adicionalmente, será relevante ajustar as soluções existentes para permitir uma integração mais eficaz entre tarefas, bem como realizar uma afinação cuidada dos hiperparâmetros, de forma a maximizar o desempenho global dos modelos.

References

1. *Deep learning based multi-label classification for surgical tool presence detection.* https://ieeexplore.ieee.org/abstract/document/7950597
2. *A Deep learning Framework for Recognising Surgical Phases in Laparoscopic Videos.* https://www.sciencedirect.com/science/article/pii/S2405896321016815
3. *Evaluation of Deep learning Models for Identifying Surgical Actions and Measuring Performance.* https://jamanetwork.com/journals/jamanetworkopen/fullarticle/2763474
4. *Multi-Modal Deep learning for Assessing Surgeon Technical Skill.* https://www.mdpi.com/1424-8220/22/19/7328
5. *Multi-modal Deep learning for Assessing Surgeon Technical Skill on a Surgical Knot-tying Task.* https://www.techrxiv.org/doi/full/10.36227/techrxiv.20085425
6. *Focal Loss.* https://paperswithcode.com/method/focal-loss
7. *Hand and instrument tracking: a systematic literature review* https://www.frontiersin.org/journals/surgery/articles/10.3389/fsurg.2025.1528362/full
8. *SurgiTrack: Fine-grained multi-class multi-tool tracking in surgical videos* https://pubmed.ncbi.nlm.nih.gov/39708509/
9. *Video-based Surgical Tool-tip and Keypoint Tracking using Multi-frame Context-driven Deep learning Models* https://arxiv.org/abs/2501.18361

Anexos

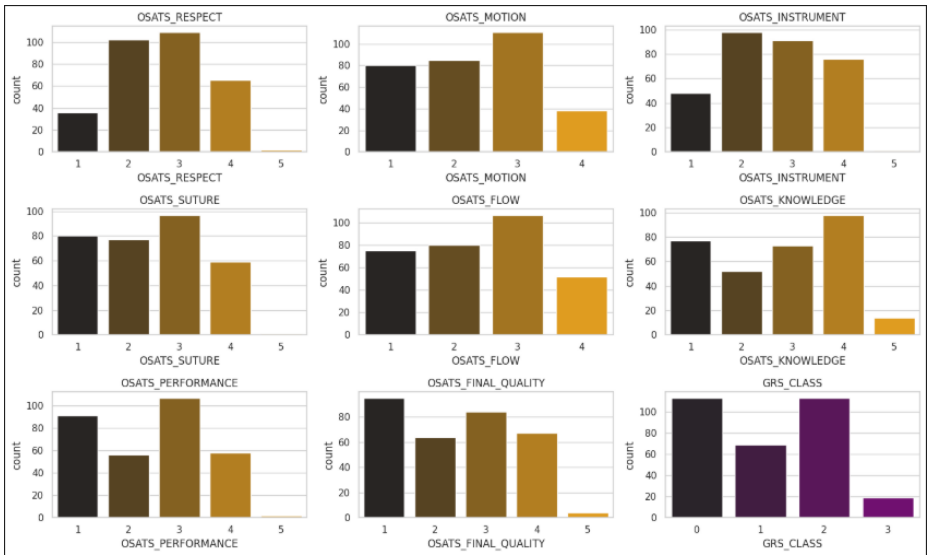

Fig. 1. Distribuição das diferentes labels

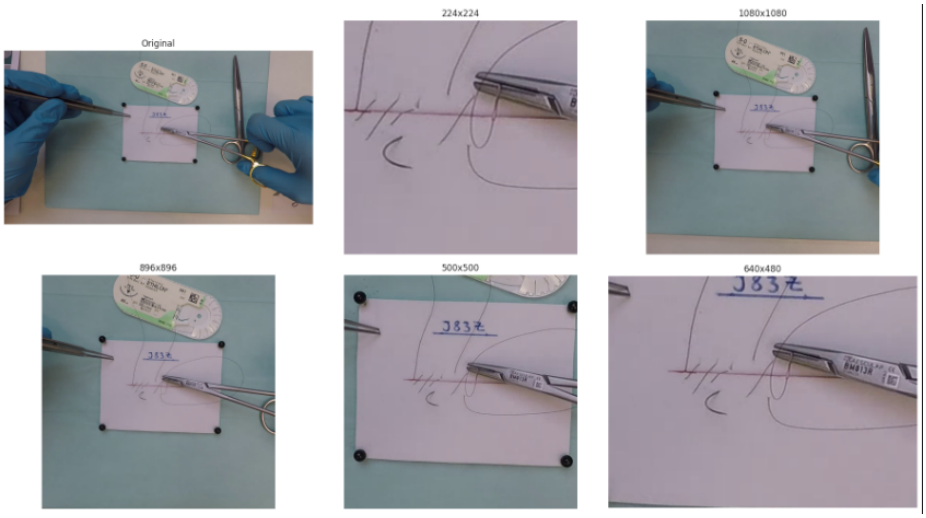

Fig. 2. Tamanhos procurados para crop

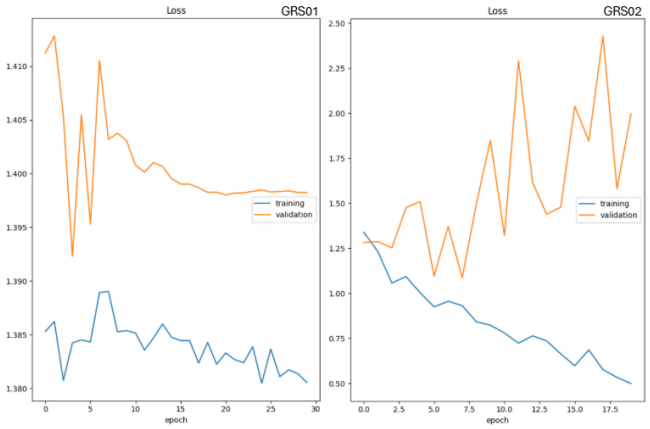

Fig. 3. Gráficos de loss para a tarefa 1

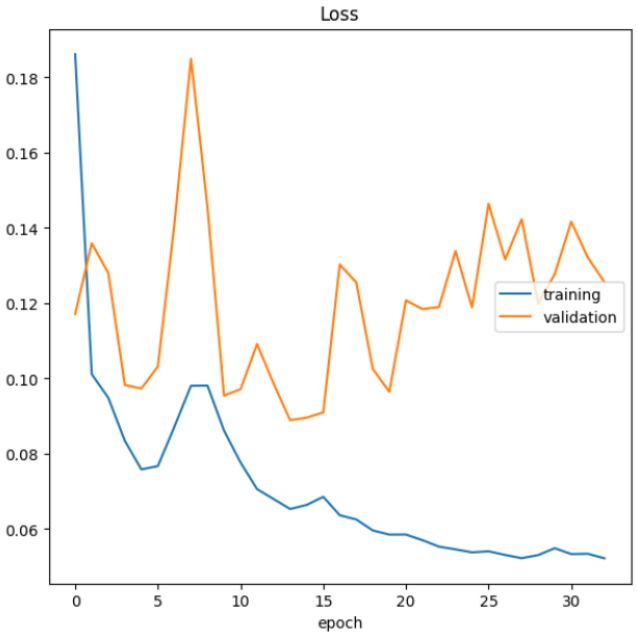

Fig. 4. Gráficos de loss para a tarefa 2

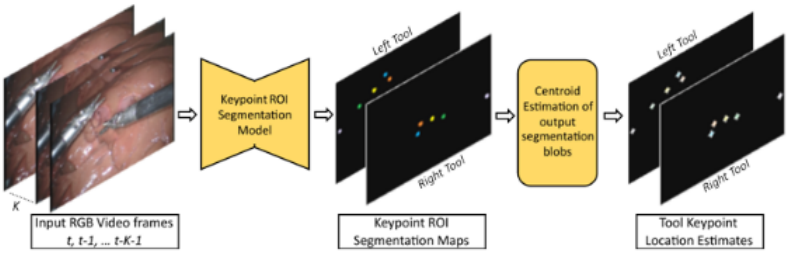

Fig. 5. Arquitetura inspirada para a tarefa 3