



**UNIVERSIDADE FEDERAL DO RECÔNCAVO DA BAHIA - UFRB**  
**CENTRO DE CIÊNCIAS EXATAS E TECNOLÓGICAS - CETEC**  
**BACHARELADO EM CIÊNCIAS EXATAS E TECNOLÓGICAS**

**Thiago Gabriel Da Silva Oliveira**

**Deteção e Reconhecimento de Dígitos em Hidrômetros com Modelos de**  
**Aprendizado de Máquina**

**CRUZ DAS ALMAS - BA**

**2025**

**Thiago Gabriel Da Silva Oliveira**

**Detecção e Reconhecimento de Dígitos em Hidrômetros com Modelos de  
Aprendizado de Máquina**

Trabalho de Conclusão de Curso apresentado  
ao Curso de Bacharelado em Ciências Exatas e Tecnológicas da Universidade Federal do Recôncavo da Bahia (UFRB), pelo Centro de Ciências Exatas e Tecnológicas (CETEC), como requisito parcial para obtenção do Título de Bacharel em Ciências Exatas e Tecnológicas.  
Orientador: Prof. Dr. Tiago Palma Pagano.

**CRUZ DAS ALMAS - BA**

**2025**

Thiago Gabriel Da Silva Oliveira

## Detecção e Reconhecimento de Dígitos em Hidrômetros com Modelos de Aprendizado de Máquina

Esta Monografia foi julgada adequada para obtenção do título de Bacharel em Ciências Exatas e Tecnológicas e aprovada em sua forma final pelo Centro de Ciências Exatas e Tecnológicas (CETEC) da Universidade Federal do Recôncavo da Bahia (UFRB).

Aprovado em: 31 / 07 / 2025

### BANCA EXAMINADORA

---

Prof. Dr. Tiago Palma Pagano (Orientador)  
UFRB - CETEC.

---

Profa. Dra. Camila Bezerra da Silva  
UFRB - CETEC.

---

Prof. Dr. Yuri Tavares dos Passos  
UFRB - CETEC.

## **DEDICATÓRIA**

Dedico este trabalho, primeiramente, a Deus, por me conceder força, saúde e perseverança ao longo desta jornada. À minha mãe, Angelina, pelo amor incondicional, pela coragem, pelo exemplo de vida e por nunca deixar de acreditar em mim, mesmo nos momentos mais difíceis. À minha família, pelo apoio constante. E aos amigos que estiveram ao meu lado durante toda a caminhada acadêmica, oferecendo companhia, conselhos e incentivo. Sem vocês, este sonho não seria possível.

“Astronarta libertado  
Minha vida me urtrapassa  
Em quarqué rota que eu faça.”

(Dois mil e um - Tom Zé)

## **AGRADECIMENTOS**

Primeiramente, agradeço a Deus pela força e sabedoria para concluir esta etapa tão importante da minha vida. Agradeço ao meu orientador, Prof. Dr. Tiago Pagano, pela paciência, orientação, ensinamentos e apoio durante todo o desenvolvimento deste trabalho. Sua dedicação foi fundamental para a realização deste TCC. Sou grato à minha mãe, Angelina, pelo amor, incentivo e por acreditar no meu potencial em todos os momentos. Aos meus familiares e amigos, que ofereceram apoio, compreensão e motivação durante esta caminhada acadêmica, meu muito obrigado. Agradeço à Universidade Federal do Recôncavo da Bahia (UFRB) e, em especial, ao Centro de Ciências Exatas e Tecnológicas (CETEC), por proporcionar uma estrutura de ensino e pesquisa que possibilitou minha formação e a realização deste trabalho. Por fim, agradeço a todos os professores e colegas que contribuíram direta ou indiretamente para a minha trajetória acadêmica.

## RESUMO

Este trabalho desenvolveu um sistema automatizado para leitura de dígitos em hidrômetros, utilizando técnicas de visão computacional e aprendizado de máquina. O objetivo foi criar modelos para segmentação e classificação de dígitos em imagens de hidrômetros, empregando a arquitetura U-Net para segmentar a região de interesse e uma rede neural convolucional (Mini-ResNet) para classificar os dígitos. A metodologia envolveu cinco etapas: seleção de datasets, pré-processamento (redimensionamento, binarização e normalização), seleção e treinamento dos modelos, e avaliação. A segmentação alcançou acurácia de 99%, com IoU de 0,73 e Dice de 0,84, enquanto a classificação atingiu 97% de acurácia e perda de 0,09. Apesar de desafios como inclinações e reflexos nas imagens, o sistema demonstrou robustez. O estudo contribui para a automação da leitura de hidrômetros, reduzindo erros humanos e otimizando a gestão de recursos hídricos, com aplicações em saneamento e transparência em faturas. Perspectivas futuras incluem a correção de perspectiva, uso de arquiteturas avançadas e expansão do dataset para maior generalização.

**Palavras-chave:** visão computacional, aprendizado de máquina, segmentação de imagens, classificação de dígitos, hidrômetros.

## ABSTRACT

This work developed an automated system for digit recognition in water meters using computer vision and machine learning techniques. The objective was to create models for digit segmentation and classification in water meter images, employing the U-Net architecture to segment the region of interest and a convolutional neural network (Mini-ResNet) to classify the digits. The methodology involved five stages: dataset selection, preprocessing (resizing, binarization, and normalization), model selection and training, and evaluation. The segmentation achieved 99% accuracy, with an IoU of 0.73 and a Dice coefficient of 0.84, while the classification reached 97% accuracy and a loss of 0.09. Despite challenges such as image tilts and reflections, the system demonstrated robustness. This study contributes to the automation of water meter reading, reducing human errors and optimizing water resource management, with applications in sanitation and billing transparency. Future perspectives include perspective correction, use of advanced architectures, and dataset expansion for improved generalization.

**Keywords:** Computer vision, machine learning, image segmentation, digit classification, water meters.



## LISTA DE FIGURAS

Figura 1	— Fluxograma das etapas da metodologia para leitura automática de hidrômetros. . . . .	27
Figura 2	— Arquitetura da U-Net para segmentação de imagens. . . . .	31
Figura 3	— Arquitetura da Mini-ResNet para classificação de imagens. .	32
Figura 4	— Curvas de aprendizado do modelo U-Net para as principais métricas nos conjuntos de treino e validação ao longo de 50 épocas. . . . .	38
Figura 5	— As imagens utilizadas abrangem diferentes condições de campo, como variações de iluminação, presença de reflexos no visor e obstruções parciais, representando uma diversidade de cenários reais. . . . .	39
Figura 6	— Curvas de aprendizado Mini-Resnet: Acurácia, Perda (Loss) nos conjuntos de treino e validação ao longo de 22 épocas. .	40
Figura 7	— Exemplos de imagens classificadas pelo modelo Mini-ResNet, incluindo casos de acerto e erro. . . . .	41
Figura 8	— Matriz de confusão do modelo Mini-ResNet no conjunto de teste. A maior parte das amostras foi corretamente classificada (valores na diagonal principal), com poucos erros, sendo o mais frequente 4 confundido com 3 (3 vezes). . . . .	42

## LISTA DE TABELAS

Tabela 1	Matriz de confusão para um problema de classificação binária.	20
Tabela 2	Métricas de desempenho do modelo U-Net no conjunto de teste para segmentação de mostradores. . . . .	37
Tabela 3	Métricas de desempenho Mini-Resnet para classificação de dígitos nos conjuntos de treino, validação e teste. . . . .	40
Tabela 4	Comparação de métricas de segmentação da região de interesse	43
Tabela 5	Comparação de acurácia na tarefa de classificação . . . . .	43

## SUMÁRIO

<b>1</b>	<b>Introdução</b>	<b>12</b>
<b>2</b>	<b>Fundamentação Teórica</b>	<b>15</b>
2.1	Leitura de Hidrômetros	15
2.2	Aprendizagem Supervisionada	16
2.3	Identificação de Objetos	18
2.4	Leitura de Dígitos	19
2.5	Métricas de Avaliação	19
<b>3</b>	<b>Trabalhos Correlatos</b>	<b>24</b>
<b>4</b>	<b>Metodologia</b>	<b>27</b>
4.1	Seleção de Datasets	28
4.2	Pré-processamento dos Dados	28
4.3	Seleção do Modelo	30
4.4	Treinamento do Modelo	34
4.5	Avaliação do Modelo	35
<b>5</b>	<b>Resultados</b>	<b>37</b>
5.1	Desempenho do Modelo de Segmentação	37
5.2	Desempenho do Modelo de Classificação	40
5.3	Comparação com Trabalhos Correlatos	42
<b>6</b>	<b>Conclusão</b>	<b>44</b>
	<b>Conclusão</b>	<b>44</b>
	<b>REFERÊNCIAS</b>	<b>47</b>

## 1 Introdução

A leitura de hidrômetros é uma atividade essencial para o controle do consumo de água em ambientes residenciais, comerciais e industriais. Tradicionalmente, essa leitura é realizada de forma manual por agentes de campo, o que pode gerar erros humanos, atrasos nos registros e altos custos operacionais. Erros em leituras manuais de hidrômetros podem gerar discrepâncias significativas nos registros de consumo, impactando diretamente a gestão de recursos hídricos e os custos operacionais das companhias de saneamento (Inovamobil, 2023).

Esse processo manual, além de estar sujeito a falhas humanas, geralmente demanda revisitas para conferência, resultando em verificações presenciais demoradas e custosas. Essa situação é especialmente comum em países em desenvolvimento, como Índia, Brasil e diversas nações africanas, onde limitações financeiras e infraestrutura precária tornam inviável a adoção de medidores digitais inteligentes. Nesses locais, o uso de medidores analógicos ainda é predominante, dificultando a modernização e a automação da leitura de consumos de água e energia (Clou Global, 2025).

Nesse contexto, soluções automatizadas baseadas em técnicas de visão computacional têm se mostrado promissoras para substituir ou complementar esse processo. Um exemplo é o sistema desenvolvido pela empresa Anyline (Anyline GmbH, 2023), que utiliza técnicas avançadas de visão computacional e reconhecimento óptico de caracteres (OCR). Inspirado por abordagens semelhantes, este trabalho propõe o desenvolvimento de um sistema próprio para leitura automática de hidrômetros, capaz de garantir precisão mesmo em mostradores sujos ou parcialmente obstruídos, reduzindo erros humanos e custos operacionais.

Com os avanços recentes em aprendizado de máquina e redes neurais profundas, possibilitou-se desenvolver sistemas capazes de identificar padrões visuais complexos, como os dígitos presentes em hidrômetros. Esses métodos oferecem uma alternativa economicamente viável para empresas de saneamento em regiões com menor infraestrutura tecnológica, uma vez que não exigem a substituição dos hidrômetros existentes.

Diante do exposto, este trabalho busca desenvolver modelos de visão computacional para segmentação e classificação de dígitos em imagens de hidrômetros.

Para que o objetivo geral seja atendido, os seguintes objetivos específicos devem ser alcançados:

- Treinar a arquitetura U-Net para segmentar a região dos dígitos nas imagens de hidrômetros.
- Desenvolver um método automatizado para extrair os dígitos a partir das regiões segmentadas.
- Treinar uma rede neural convolucional para classificar individualmente os dígitos extraídos.
- Avaliar a acurácia e a robustez do sistema integrado para a leitura automática dos valores exibidos.

Diversos estudos vêm explorando abordagens semelhantes. Limeira et al. (LIMEIRA, 2021) aplicaram a arquitetura U-Net para segmentação de mapas históricos, demonstrando sua eficácia na separação de regiões complexas em imagens, o que valida sua aplicação na segmentação de dígitos em hidrômetros. Para o reconhecimento de números, o uso de redes neurais convolucionais (CNNs) é amplamente consolidado, sendo o dataset MNIST uma referência clássica para o desenvolvimento e avaliação de modelos de classificação de dígitos manuscritos (LECUN et al., 1998). Em cenários mais realistas, como o dataset SVHN (NETZER et al., 2011), desafios como variação de iluminação e sobreposição de caracteres também são enfrentados, e estudos mostraram que modelos baseados em aprendizado profundo são capazes de superar essas dificuldades.

Estudo realizado por Souza (SOUZA, 2022) aplicou a arquitetura Faster R-CNN para detectar e reconhecer dígitos em hidrômetros. Os resultados obtidos reforçam a relevância do problema tratado neste trabalho e demonstram o potencial das técnicas profundas em contextos reais. Essas abordagens não apenas aumentam a eficiência operacional, mas também possibilitam a integração com sistemas inteligentes de gestão de recursos hídricos, viabilizando leituras remotas, em tempo real e com menor margem de erro.

Além disso, ao oferecer dados confiáveis de consumo, essas soluções tecnológicas beneficiam tanto os usuários quanto os prestadores de serviço, possibilitando melhor planejamento, controle e conservação dos recursos hídricos. O uso de técnicas como segmentação semântica e redes convolucionais para classificação viabiliza a construção de sistemas robustos mesmo em condições adversas de iluminação, posicionamento ou obstrução parcial do visor dos hidrômetros.

Este trabalho está organizado da seguinte forma: no Capítulo 2, apresenta-se o referencial teórico, abordando os fundamentos de visão computacional, redes neurais e segmentação de imagens. No Capítulo 3, detalha-se a metodologia proposta, incluindo

o desenvolvimento do sistema e os experimentos realizados. No Capítulo 4, são apresentados os resultados e a análise de desempenho do sistema. Por fim, no Capítulo 5, discutem-se as conclusões e perspectivas para trabalhos futuros.

## **2 Fundamentação Teórica**

Este capítulo apresenta os conceitos fundamentais relacionados à leitura automática de hidrômetros, explorando os princípios que sustentam o uso de tecnologias modernas para essa tarefa. São abordados temas como a leitura de hidrômetros, aprendizagem de máquina, visão computacional e métricas de avaliação. Cada seção é estruturada para apresentar informações de forma organizada e acessível. Com propósito de contextualizar o problema e fornecer as bases teóricas para aplicações práticas.

### **2.1 Leitura de Hidrômetros**

A gestão eficiente de recursos hídricos é um pilar essencial para a sustentabilidade, particularmente em regiões onde a água é um recurso escasso ou sujeito a alta demanda. Hidrômetros são instrumentos amplamente utilizados para medir o consumo de água em residências, indústrias e estabelecimentos comerciais. Esses dispositivos fornecem dados críticos para o faturamento, o monitoramento do uso e a elaboração de políticas de conservação hídrica. A leitura de hidrômetros, tradicionalmente realizada de forma manual, envolve a visita de técnicos aos locais de instalação para registrar os valores exibidos nos mostradores. No entanto, esse processo apresenta desafios significativos, incluindo erros humanos, custos operacionais elevados e dificuldades de acesso a áreas remotas ou de alta densidade populacional (BANK, 2016).

A automação da leitura de hidrômetros surge como uma solução para superar essas limitações, oferecendo maior precisão, eficiência e escalabilidade. Tecnologias modernas, como sistemas baseados em sensores, comunicação sem fio e visão computacional, têm sido exploradas para digitalizar as leituras. Esses sistemas permitem a captura remota ou semiautomática dos dados, reduzindo a necessidade de intervenção humana e minimizando erros. Além disso, a automação é capaz de operar em condições adversas, como variações de iluminação, desgaste físico dos hidrômetros ou presença de sujeira no mostrador, contribuindo para a modernização da gestão de recursos hídricos e para a criação de sistemas mais robustos e confiáveis.

## 2.2 Aprendizagem Supervisionada

A aprendizagem de máquina é um campo da inteligência artificial que capacita sistemas computacionais a aprenderem padrões a partir de dados, sem a necessidade de programação explícita para cada tarefa. Essa área tem revolucionado diversas indústrias, incluindo a gestão de recursos hídricos, ao permitir a análise de grandes volumes de dados e a automatização de processos complexos (GOODFELLOW; BENGIO; COURVILLE, 2016). Na leitura de hidrômetros, a aprendizagem de máquina é usada para processar imagens dos mostradores, identificar dígitos e interpretar os valores exibidos, mesmo em cenários desafiadores, como imagens com ruídos, ângulos distorcidos ou iluminação irregular (AZEEM et al., 2020).

Os algoritmos de aprendizagem de máquina podem ser divididos em três categorias principais: supervisionada, não supervisionada e por Reforço (ALPAYDIN, 2020). A aprendizagem supervisionada, descrita em detalhes na próxima seção, é particularmente relevante para a leitura de hidrômetros, pois permite treinar modelos com base em exemplos rotulados, como imagens de hidrômetros associadas aos valores numéricos corretos. Uma das principais vantagens da aprendizagem de máquina é sua capacidade de generalização, permitindo que os modelos se adaptem a variações nos dados, como diferentes tipos de hidrômetros ou condições de captura de imagem como iluminação variável, ângulos distorcidos e ruídos. Essa flexibilidade é crucial para aplicações práticas em larga escala, onde os dados podem variar significativamente e os modelos precisam ser robustos a essas mudanças para fornecer resultados precisos e confiáveis.

Esse processo de adaptação é fortemente guiado pela otimização por gradiente descendente, onde a atualização dos parâmetros do modelo é feita com base no cálculo do gradiente. O gradiente de uma função multivariada é um vetor que aponta na direção de maior variação da função em um ponto específico, ou seja, fornece as informações necessárias para o modelo ajustar seus parâmetros de forma eficaz, independentemente das variações nos dados.

Em termos matemáticos, o gradiente é representado como o vetor de derivadas parciais da função em relação a cada uma de suas variáveis. Se a função  $f(x_1, x_2, \dots, x_n)$  é uma função escalar, o gradiente de  $f$  é dado pela equação 2.1:

$$\nabla f(x) = \left( \frac{\partial f}{\partial x_1}, \frac{\partial f}{\partial x_2}, \dots, \frac{\partial f}{\partial x_n} \right) \quad (2.1)$$

onde  $\nabla f(x)$  denota o gradiente da função  $f(x)$ , indicando a direção e a taxa de variação mais rápida da função. Já  $\frac{\partial f}{\partial x_i}$  representa as derivadas parciais de  $f(x)$  em relação às variáveis  $x_1, x_2, \dots, x_n$ . Por fim,  $x_1, x_2, \dots, x_n$  correspondem às variáveis independentes



que formam o vetor  $x$  em um espaço de dimensão  $n$ .

No contexto do aprendizado de máquina, o vetor  $x$  corresponde aos parâmetros do modelo, geralmente representados por  $\theta$ . Assim, o gradiente pode ser escrito como:

$$\nabla f(\theta) = \left( \frac{\partial f}{\partial \theta_1}, \frac{\partial f}{\partial \theta_2}, \dots, \frac{\partial f}{\partial \theta_n} \right) \quad (2.2)$$

Esse vetor de gradiente fornece as direções e os ajustes necessários nos parâmetros  $\theta$  para otimizar o modelo. A capacidade do gradiente em se adaptar ao comportamento dos dados, ajustando-se a diferentes condições, como a variação dos tipos de hidrômetros ou da iluminação na captura das imagens, é uma das razões pelas quais o modelo pode generalizar tão bem e oferecer bons resultados mesmo em cenários desafiadores.

A otimização por gradiente descendente é o método que usa essa informação para atualizar os parâmetros do modelo de forma iterativa. A fórmula básica para a atualização dos parâmetros é dada pela equação 2.3:

$$\theta = \theta - \eta \nabla f(\theta) \quad (2.3)$$

onde  $\theta$  denota os parâmetros do modelo, como, por exemplo, os pesos de uma rede neural,  $\eta$  é a taxa de aprendizado, que controla o tamanho do passo dado em cada iteração, e  $\nabla f(\theta)$  é o gradiente da função de custo  $f(\theta)$  em relação aos parâmetros  $\theta$ .

Esse processo iterativo permite que o modelo se ajuste continuamente aos dados, sendo capaz de se adaptar a diferentes condições, como tipos variados de hidrômetros ou ruídos nas imagens, otimizando sua performance ao longo do tempo, como é discutido por (GOODFELLOW; BENGIO; COURVILLE, 2016).

É importante destacar que existem várias variações do algoritmo de gradiente descendente, como o gradiente descendente estocástico (SGD) e o gradiente descendente por mini-batch, que são utilizados para otimizar o processo, especialmente quando se lida com grandes volumes de dados ou dados com variações significativas.

Assim, a otimização por gradiente descendente constitui o fundamento matemático que possibilita o treinamento eficaz de modelos. Com essa base estabelecida, passamos à descrição da aprendizagem supervisionada, que utiliza esse mecanismo para atualizar os parâmetros do modelo com base em dados rotulados.

A aprendizagem supervisionada é uma abordagem da aprendizagem de máquina em que um modelo é treinado com um conjunto de dados rotulados, ou seja, pares de entradas (como imagens) e saídas esperadas (como os dígitos correspondentes). Durante o treinamento, o modelo ajusta seus parâmetros internos para minimizar a diferença entre suas previsões e os rótulos fornecidos, utilizando técnicas como opti-

zação por gradiente descendente. Essa abordagem é amplamente utilizada em tarefas de visão computacional, como a classificação de imagens e o reconhecimento de padrões (BISHOP, 2006).

No contexto da leitura de hidrômetros, a aprendizagem supervisionada é aplicada para treinar modelos que identifiquem e classifiquem dígitos em imagens dos mostradores. Um exemplo clássico é o uso de redes neurais convolucionais (CNNs), que são particularmente eficazes para processar dados visuais. Essas redes aprendem a extrair características relevantes, como contornos e formas dos dígitos, diretamente das imagens, eliminando a necessidade de engenharia manual de características. Datasets como o MNIST (LECUN et al., 1998) e o SVHN (NETZER et al., 2011), que contêm imagens de dígitos manuscritos e números em contextos reais, respectivamente, são frequentemente usados como referência para desenvolver modelos robustos, adaptáveis a diferentes condições de captura.

### 2.3 Identificação de Objetos

A visão computacional é um ramo da ciência da computação que busca capacitar máquinas a interpretar e compreender informações visuais, como imagens e vídeos, de maneira semelhante à visão humana. Essa área combina técnicas de processamento de imagens, reconhecimento de padrões e aprendizagem de máquina para realizar tarefas como detecção de objetos, segmentação de imagens e classificação. No contexto da leitura de hidrômetros, a visão computacional é essencial para capturar imagens dos mostradores, processá-las e extrair os valores numéricos exibidos (SZELISKI, 2010).

As principais etapas de um sistema de visão computacional incluem a aquisição de imagens, o pré-processamento, a segmentação e a classificação. O pré-processamento envolve técnicas como conversão para escala de cinza, redimensionamento e remoção de ruídos, que preparam a imagem para análise. A segmentação isola regiões de interesse, como os dígitos no mostrador, enquanto a classificação associa essas regiões a valores numéricos específicos. Essas etapas, quando combinadas com algoritmos de aprendizagem de máquina, permitem que sistemas de visão computacional alcancem alta precisão, mesmo em condições desafiadoras, como variações de iluminação ou ângulos de captura desfavoráveis.

A identificação de objetos é uma tarefa central da visão computacional, que consiste em detectar e localizar elementos específicos em uma imagem ou vídeo. Essa tarefa pode variar desde a detecção de objetos genéricos, como carros ou pessoas, até a identificação de elementos específicos, como dígitos em um hidrômetro. A segmentação de imagens é uma subárea da identificação de objetos que foca em isolar regiões de interesse com alta precisão, frequentemente usando arquiteturas de redes neurais espe-

cializadas, como a U-Net (RONNEBERGER; FISCHER; BROX, 2015).

A U-Net é uma rede neural convolucional projetada para segmentação, com uma estrutura que combina caminhos de contração (para capturar contexto) e expansão (para recuperar detalhes espaciais). Essa arquitetura é particularmente eficaz em tarefas que exigem segmentação precisa, mesmo com datasets de tamanho limitado. Na leitura de hidrômetros, a identificação de objetos é usada para isolar os dígitos do mostrador, permitindo que o sistema foque na análise das regiões relevantes e ignore elementos irrelevantes, como o fundo ou partes do dispositivo.

## 2.4 Leitura de Dígitos

A leitura de dígitos é uma aplicação específica da visão computacional que combina segmentação e classificação para interpretar números exibidos em dispositivos como hidrômetros, medidores de energia ou relógios analógicos.

Para lidar com essa tarefa, redes neurais convolucionais (CNNs) têm sido amplamente utilizadas, devido à sua capacidade de extrair e aprender representações visuais complexas diretamente das imagens (KRIZHEVSKY; SUTSKEVER; HINTON, 2012). Entre as arquiteturas mais eficazes estão as redes do tipo ResNet (Residual Networks), que utilizam blocos residuais para mitigar o problema do desaparecimento do gradiente em redes profundas, facilitando o aprendizado de funções mais complexas (HE et al., 2016).

A estrutura residual permite que a rede aprenda mapeamentos de identidade, acelerando o treinamento e melhorando o desempenho. Essa abordagem tem se mostrado especialmente útil em tarefas sujeitas a ruídos visuais e variações contextuais. Em ambientes com maior complexidade visual, como os encontrados em imagens reais, arquiteturas residuais tendem a apresentar melhor robustez.

## 2.5 Métricas de Avaliação

A avaliação do desempenho de sistemas de visão computacional e aprendizado de máquina é fundamental para mensurar sua eficácia e identificar oportunidades de aprimoramento. Diferentes métricas são aplicadas conforme a tarefa realizada.

No caso da classificação de dígitos, onde frequentemente há desequilíbrio entre as classes, utilizamos métricas que melhor refletem essa característica. A base para essas métricas é a **matriz de confusão**, apresentada na Tabela 1, que resume o desempenho do modelo de classificação ao comparar os rótulos reais com os rótulos preditos.

**Tabela 1 – Matriz de confusão para um problema de classificação binária.**

	<b>Classe Predita: Positiva</b>	<b>Classe Predita: Negativa</b>
<b>Classe Real: Positiva</b>	Verdadeiro Positivo (VP)	Falso Negativo (FN)
<b>Classe Real: Negativa</b>	Falso Positivo (FP)	Verdadeiro Negativo (VN)

A partir da matriz de confusão, podemos definir diversas métricas de avaliação. Uma das mais utilizadas é a acurácia, que representa a proporção de previsões corretas (tanto positivas quanto negativas) em relação ao total de amostras avaliadas. Ela é especialmente útil como métrica geral de desempenho, desde que as classes estejam razoavelmente balanceadas (TERVEN et al., 2025). A Equação 2.4 apresenta sua definição formal:

$$\text{Acurácia} = \frac{VP + VN}{VP + VN + FP + FN} \quad (2.4)$$

onde  $VP$  (Verdadeiros Positivos) representa as amostras positivas corretamente classificadas,  $VN$  (Verdadeiros Negativos) são as amostras negativas corretamente classificadas,  $FP$  (Falsos Positivos) correspondem às amostras negativas que foram incorretamente classificadas como positivas, e  $FN$  (Falsos Negativos) são as amostras positivas que foram incorretamente classificadas como negativas.

Essa métrica fornece uma visão geral do desempenho do modelo de classificação, mas não orienta diretamente o ajuste dos parâmetros durante o treinamento. Para isso, utiliza-se a função de perda (loss), que quantifica o erro do modelo em relação aos rótulos corretos. Em problemas de classificação multiclasse, os rótulos geralmente são representados utilizando a codificação one-hot, técnica que permite aplicar corretamente funções de perda como a categorical crossentropy (GOODFELLOW; BENGIO; COURVILLE, 2016), a codificação one-hot é uma técnica comum para representar variáveis categóricas de forma binária em problemas de aprendizado de máquina. Nessa abordagem, cada classe é representada por um vetor em que a posição correspondente à classe é marcada com 1, enquanto todas as outras posições são 0, por exemplo, em um problema com 3 classes (A, B, C), a codificação one-hot seria: Classe A: [1, 0, 0], Classe B: [0, 1, 0] e Classe C: [0, 0, 1], representação é útil porque permite que o modelo trabalhe diretamente com as classes, sem precisar lidar com valores numéricos contínuos, o que poderia ser interpretado como uma ordem ou distância entre as classes.

A função de perda categorical crossentropy é uma função de erro muito utilizada para problemas de classificação múltipla, especialmente quando as saídas são representadas com codificação one-hot. Ela calcula a diferença entre a distribuição de probabilidades predita pelo modelo e a distribuição verdadeira representada pelos rótulos codificados one-hot (TERVEN et al., 2025). A fórmula matemática dessa função é

dada pela equação 2.5:

$$\text{Loss} = - \sum_{i=1}^C y_i \cdot \log(p_i) \quad (2.5)$$

onde  $C$  é o número de classes,  $y_i$  é o valor da codificação one-hot para a classe  $i$  em que 1 indica a classe correta e 0 as outras e  $p_i$  é a probabilidade predita pelo modelo para a classe  $i$ .

Em outras palavras, a categorical crossentropy mede o quão bem as probabilidades preditas correspondem às classes reais. Quanto maior a probabilidade atribuída à classe correta, menor será o valor da perda. A função de perda encoraja o modelo a aprender a atribuir altas probabilidades à classe correta e baixas probabilidades às classes incorretas.

Se o modelo predizer corretamente uma classe ou seja,  $p_i$  para a classe verdadeira é próximo de 1, a perda será pequena, e se o modelo errar ou seja,  $p_i$  para a classe verdadeira é próximo de 0, a perda será grande.

As métricas a seguir são utilizadas para avaliar a qualidade das previsões em tarefas de segmentação, como o isolamento de dígitos em imagens de hidrômetros utilizando o modelo U-Net. Cada métrica é apresentada com sua respectiva equação e uma descrição detalhada das variáveis envolvidas.

A acurácia de pixel mede a proporção de pixels corretamente classificados em relação ao total de pixels da imagem (TERVEN et al., 2025), conforme definido na Equação 2.6:

$$\text{Acurácia} = \frac{1}{H \times W} \sum_{i=1}^H \sum_{j=1}^W \mathbf{1}(\hat{y}_{ij} = y_{ij}) \quad (2.6)$$

onde  $H \times W$  é a dimensão da imagem, com  $H$  sendo a altura e  $W$  a largura em pixels;  $y_{ij} \in \{0, 1\}$  é o valor real do pixel na posição  $(i, j)$  da máscara verdadeira, onde 1 indica que o pixel pertence à região de interesse e 0 indica que não pertence;  $\hat{y}_{ij} \in \{0, 1\}$  é o valor previsto do pixel na posição  $(i, j)$  da máscara predita, onde 1 indica que o pixel foi classificado como pertencente à região de interesse e 0 indica que não foi;  $\mathbf{1}(\hat{y}_{ij} = y_{ij})$  é a função indicadora que retorna 1 se o pixel na posição  $(i, j)$  foi corretamente classificado (*i.e.*,  $\hat{y}_{ij} = y_{ij}$ ) e 0 caso contrário.

O coeficiente de Dice mede a similaridade entre as máscaras predita e verdadeira, enfatizando a interseção entre elas (TERVEN et al., 2025), conforme apresentado na Equação 2.7:

$$\text{Dice} = \frac{2 \times |Y \cap \hat{Y}|}{|Y| + |\hat{Y}|} = \frac{2 \sum_{i=1}^H \sum_{j=1}^W y_{ij} \hat{y}_{ij}}{\sum_{i=1}^H \sum_{j=1}^W y_{ij} + \sum_{i=1}^H \sum_{j=1}^W \hat{y}_{ij}} \quad (2.7)$$

onde  $Y$  é a máscara verdadeira e  $\hat{Y}$  é a máscara predita;  $|Y \cap \hat{Y}|$  é a cardinalidade da interseção entre as máscaras, correspondendo à soma dos pixels corretamente classificados como positivos;  $|Y| + |\hat{Y}|$  é a soma das cardinalidades das máscaras verdadeira e predita, representando o total de pixels positivos nas duas máscaras.

A Intersection over Union (IoU), também chamada de índice de Jaccard, calcula a razão entre a interseção e a união das máscaras verdadeira e predita (TERVEN et al., 2025), conforme descrito na Equação 2.8:

$$\text{IoU} = \frac{|Y \cap \hat{Y}|}{|Y \cup \hat{Y}|} = \frac{\sum_{i=1}^H \sum_{j=1}^W y_{ij} \hat{y}_{ij}}{\sum_{i=1}^H \sum_{j=1}^W y_{ij} + \sum_{i=1}^H \sum_{j=1}^W \hat{y}_{ij} - \sum_{i=1}^H \sum_{j=1}^W y_{ij} \hat{y}_{ij}} \quad (2.8)$$

onde  $|Y \cap \hat{Y}|$  é a cardinalidade da interseção entre as máscaras verdadeira e predita, representando os pixels corretamente classificados como positivos, e  $|Y \cup \hat{Y}|$  é a cardinalidade da união das máscaras, correspondendo ao total de pixels positivos presentes em pelo menos uma das máscaras.

A Precision avalia a proporção de predições positivas corretas em relação ao total de predições positivas feitas pelo modelo, sendo útil para minimizar falsos positivos (TERVEN et al., 2025), conforme apresentado na Equação 2.9:

$$\text{Precision} = \frac{\sum_{i=1}^H \sum_{j=1}^W y_{ij} \cdot \hat{y}_{ij}}{\sum_{i=1}^H \sum_{j=1}^W \hat{y}_{ij} + \epsilon} \quad (2.9)$$

onde  $\epsilon$  é um pequeno valor constante, tipicamente  $10^{-7}$ , adicionado ao denominador para evitar divisão por zero. Esta métrica representa a proporção de pixels preditos como positivos que realmente pertencem à região de interesse.

A métrica de recall, ou sensibilidade, mede a proporção de pixels positivos corretamente identificados pelo modelo, sendo sensível a falsos negativos (TERVEN et al., 2025), conforme definido na Equação 2.10:

$$\text{Recall} = \frac{\sum_{i=1}^H \sum_{j=1}^W y_{ij} \cdot \hat{y}_{ij}}{\sum_{i=1}^H \sum_{j=1}^W y_{ij} + \epsilon} \quad (2.10)$$

O F1-score combina precisão e recall em uma única métrica, calculando a média harmônica entre elas, sendo útil para avaliar o equilíbrio entre falsos positivos e falsos negativos (TERVEN et al., 2025), conforme descrito na Equação 2.11:

$$\text{F1-score} = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall} + \epsilon} \quad (2.11)$$

onde Precision e Recall são as métricas definidas nas Equações 2.9 e 2.10, respectivamente.

A função de perda binary crossentropy é utilizada durante o treinamento para otimizar os pesos da rede, medindo a divergência entre os valores reais e as probabilidades preditas (TERVEN et al., 2025), conforme apresentado na Equação 2.12:

$$\mathcal{L} = -\frac{1}{H \times W} \sum_{i=1}^H \sum_{j=1}^W [y_{ij} \cdot \log(p_{ij}) + (1 - y_{ij}) \cdot \log(1 - p_{ij})] \quad (2.12)$$

onde  $p_{ij} \in [0, 1]$  é a probabilidade prevista pelo modelo para o pixel  $(i, j)$ , representando a confiança de que ele pertence à região de interesse.

### 3 Trabalhos Correlatos

Pesquisas recentes têm se concentrado na automação da leitura de hidrômetros e medidores de energia por meio de redes neurais e técnicas de visão computacional, com o objetivo de substituir o processo manual tradicional por soluções mais precisas e eficientes.

Slyusar et al. (2022) compararam diferentes arquiteturas de redes neurais para segmentação de mostradores analógicos de medidores, alcançando uma precisão de até 85,20% com uma versão simplificada da arquitetura U-Net. O foco principal do estudo foi a segmentação eficiente das regiões de interesse, destacando o potencial das redes neurais convolucionais em lidar com estruturas analógicas complexas.

De forma complementar, Saetan e Dittakan (2024) desenvolveram um sistema baseado em Redes Neurais Convolucionais (CNNs) para leitura automática de hidrômetros tailandeses a partir de imagens capturadas por smartphones. O modelo proposto obteve 99,7% de precisão na localização dos grupos de dígitos e 96,6% na leitura dos números. Ao contrário de Slyusar et al. (2022), que se concentraram na segmentação, Saetan e Dittakan (2024) abordaram o processo completo de localização e leitura dos números, demonstrando a viabilidade de um sistema robusto para múltiplos modelos de hidrômetros.

Salomon, Laroca e Menotti (2020) utilizaram arquiteturas de detecção de objetos como Faster R-CNN e YOLOv3 para a tarefa de localização e reconhecimento de dígitos em imagens de medidores. Apesar do bom desempenho em condições normais, os autores enfrentaram dificuldades em cenários adversos de iluminação e qualidade de imagem, obtendo uma taxa de acerto de 75,25% na leitura completa. Ainda assim, a abordagem alcançou um F1-score de 100%, evidenciando seu potencial em ambientes controlados.

Imran et al. (2023) aplicaram a arquitetura YOLOv3 em um conjunto de 10.000 imagens de medidores de energia elétrica. Com o uso de técnicas de aumento de dados (data augmentation), o modelo atingiu uma precisão média de 98%. A principal contribuição do estudo foi a automação da leitura de números de série, reduzindo significativamente o esforço humano. No entanto, os autores destacam a necessidade de expandir o conjunto de dados e otimizar o modelo para diferentes contextos. Futuras



direções incluem a integração com dispositivos de Internet das Coisas (IoT) e o desenvolvimento de uma interface gráfica (GUI) para aplicação em tempo real.

Souza (2022) propuseram uma abordagem baseada em OCR com deep learning para a leitura de hidrômetros em condomínios residenciais. Inicialmente, testaram soluções prontas como Amazon Textract e Tesseract, mas os resultados foram insatisfatórios. Para contornar essa limitação, criaram um conjunto de dados personalizado com imagens reais de dois edifícios em Recife. A equipe treinou 22 variações do Faster R-CNN e selecionou o modelo com maior precisão média (mAP), que alcançou mais de 90% de acurácia na leitura dos cinco primeiros dígitos. Essa abordagem se mostrou promissora para melhorar a confiabilidade dos dados em sistemas baseados em IoT.

Radyhin (2023) realizou a segmentação de dígitos em medidores de água utilizando a arquitetura U-Net com backbone EfficientNetB0. As imagens e máscaras foram redimensionadas para 256×256 pixels e submetidas a técnicas de aumento de dados. O treinamento foi realizado por 50 épocas com a função de perda FocalLoss e otimizador Adam. O desempenho do modelo foi avaliado utilizando o coeficiente de Dice, alcançando um valor de 0,63, o que demonstra um resultado promissor para aplicações em ambientes reais.

Por fim, um estudo recente de Fei et al. (2023) propôs um método aprimorado para segmentação da área de leitura de hidrômetros digitais com o uso da arquitetura Double-RSU U<sup>2</sup>-Net. Essa versão estendida do U<sup>2</sup>-Net original aumenta a profundidade da rede, resultando em melhorias nas métricas MIoU, MAE e F1-Score em relação a abordagens tradicionais. O modelo demonstrou robustez frente a desafios como variabilidade nos formatos dos hidrômetros, interferência de caracteres fora da área de leitura e desequilíbrio de classes. Em testes comparativos sem aumento de dados, o U<sup>2</sup>-Net alcançou F1-Score de 0,93, enquanto a versão Double-RSU U<sup>2</sup>-Net atingiu 0,93. Com técnicas de data augmentation, esse valor subiu para 0,94, evidenciando a eficácia da proposta.

Diversos estudos vêm explorando abordagens semelhantes. Limeira et al. Limeira (2021) aplicaram a arquitetura U-Net para segmentação de mapas históricos, demonstrando sua eficácia na separação de regiões complexas em imagens, o que valida sua aplicação na segmentação de dígitos em hidrômetros. Para o reconhecimento de números, o uso de redes neurais convolucionais (CNNs) é amplamente consolidado, sendo o dataset MNIST uma referência clássica para o desenvolvimento e avaliação de modelos de classificação de dígitos manuscritos (LECUN et al., 1998). Em cenários mais realistas, como o dataset SVHN (NETZER et al., 2011), desafios como variação de iluminação e sobreposição de caracteres também são enfrentados, e estudos mostraram que modelos baseados em aprendizado profundo são capazes de superar essas

dificuldades.

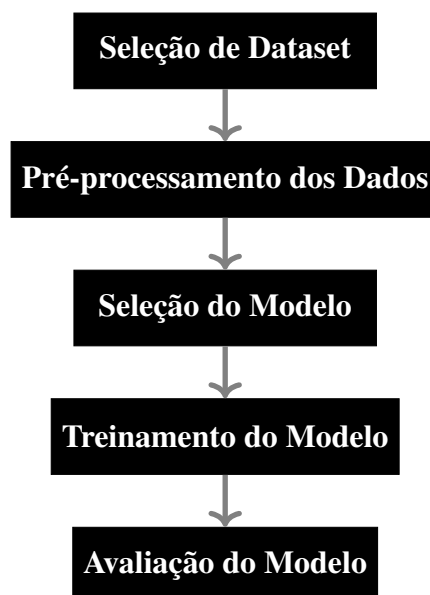
Estudo realizado por Souza (2022) aplicou a arquitetura Faster R-CNN para detectar e reconhecer dígitos em hidrômetros. Os resultados obtidos reforçam a relevância do problema tratado neste trabalho e demonstram o potencial das técnicas profundas em contextos reais. Essas abordagens não apenas aumentam a eficiência operacional, mas também possibilitam a integração com sistemas inteligentes de gestão de recursos hídricos, viabilizando leituras remotas, em tempo real e com menor margem de erro.

Além disso, ao oferecer dados confiáveis de consumo, essas soluções tecnológicas beneficiam tanto os usuários quanto os prestadores de serviço, possibilitando melhor planejamento, controle e conservação dos recursos hídricos. O uso de técnicas como segmentação semântica e redes convolucionais para classificação viabiliza a construção de sistemas robustos mesmo em condições adversas de iluminação, posicionamento ou obstrução parcial do visor dos hidrômetros.

## 4 Metodologia

Este capítulo descreve os procedimentos adotados para desenvolver o sistema de leitura automática de hidrômetros, com foco na segmentação da região de interesse (ROI) e no reconhecimento de dígitos. A metodologia foi dividida em cinco etapas principais: Seleção de Dataset, Pré-processamento dos Dados, Seleção do Modelo, Treinamento do Modelo e Avaliação do Modelo. Cada etapa é detalhada em subseções, e um diagrama textual ilustra o fluxo do processo.

Na Figura 1 é apresentado um diagrama com a sequência das etapas da metodologia:



**Figura 1 – Fluxograma das etapas da metodologia para leitura automática de hidrômetros.**

O processo começa com a escolha do dataset, seguido pelo pré-processamento das imagens para segmentação e classificação. Depois, os modelos U-Net para segmentação e CNN para classificação são selecionados, treinados e avaliados, formando um pipeline integrado para a leitura automática de hidrômetros.

## 4.1 Seleção de Datasets

Os datasets utilizados neste trabalho foram o disponibilizado pelo Yandex Toloka (KUCEV, 2020) e o dataset complementar proposto por Yang et al. (YANG et al., 2019). O primeiro contém 1.244 imagens RGB de hidrômetros com máscaras de segmentação pré-rotuladas, adequadas para tarefas de segmentação da região de interesse (ROI) e reconhecimento de dígitos. O segundo inclui 2.000 imagens de hidrômetros, utilizado para complementar a tarefa de classificação de dígitos. Ambos os datasets foram escolhidos por sua acessibilidade pública, alta qualidade das anotações e diversidade em condições de iluminação, ângulos e tipos de mostradores, garantindo representatividade de cenários reais e contribuindo para a robustez dos experimentos.

O dataset disponibilizado pelo (KUCEV, 2020) foi alocado para a tarefa de segmentação, utilizando suas 1.244 imagens com máscaras pré-rotuladas para identificar a região de interesse ROI nos hidrômetros. Para a tarefa de classificação, ambos os datasets foram combinados, totalizando inicialmente 3.244 imagens, sendo 1.244 do (KUCEV, 2020) e 2.000 do dataset proposto por (YANG et al., 2019), destinadas ao reconhecimento de dígitos. A preparação dos dados para essas tarefas, incluindo a extração de dígitos individuais, é descrita na seção de pré-processamento subsequente.

## 4.2 Pré-processamento dos Dados

O pré-processamento foi dividido em duas etapas distintas, uma para a tarefa de segmentação e outra para a tarefa de classificação, para preparar as imagens de forma adequada para os respectivos modelos.

O pré-processamento para a tarefa de segmentação foi projetado para preparar as imagens RGB e suas respectivas máscaras binárias do dataset disponibilizado pelo Yandex Toloka (KUCEV, 2020) para o modelo U-Net, garantindo compatibilidade com a arquitetura, eficiência computacional no ambiente do Google Colab e robustez no treinamento. As etapas incluíram padronização das imagens e máscaras com redimensionamento e normalização otimizados, binarização das máscaras para clara representação da região de interesse e divisão estratificada dos dados para avaliação equilibrada do modelo. Essas ações reduziram a complexidade computacional e asseguraram um conjunto de dados adequado para a segmentação.

As 1.244 imagens RGB e suas máscaras binárias do dataset disponibilizado pelo Yandex Toloka (KUCEV, 2020) foram redimensionadas para  $224 \times 224$  pixels e normalizadas para o intervalo  $[0, 1]$ . As máscaras foram binarizadas considerando valores do canal 0 maiores que 0,1 como 1 e os demais como 0. O conjunto foi dividido em treino (987 imagens), validação (247 imagens) e teste fixo (10 imagens), com embar-

lhamento aleatório para garantir a reprodutibilidade.

O pré-processamento para a tarefa de classificação foi projetado para preparar as imagens do dataset disponibilizado pelo Yandex Toloka (KUCEV, 2020) e do dataset complementar proposto por Yang e colaboradores (YANG et al., 2019) para a classificação de dígitos. As etapas incluíram a extração da região de interesse (ROI) usando máscaras binárias, recorte e alinhamento das ROIs, divisão em dígitos individuais, redimensionamento, conversão para tons de cinza e divisão estratificada dos dados. Essas ações garantiram a padronização das imagens, corrigiram variações de orientação, eliminaram amostras inadequadas e reduziram a complexidade computacional, criando um conjunto robusto e equilibrado para o treinamento do modelo de classificação.

Como o modelo de segmentação ainda não estava finalizado durante a etapa de pré-processamento, foi utilizado um algoritmo alternativo baseado nas máscaras binárias do dataset disponibilizado pelo Yandex Toloka (KUCEV, 2020). Nessas máscaras, a região do hidrômetro aparece em branco sobre um fundo preto. O algoritmo identificou os pixels brancos e calculou os limites da área de interesse, definindo uma caixa delimitadora retangular bounding box a partir das coordenadas mínimas e máximas. Para imagens inclinadas, foi implementada a opção de gerar caixas delimitadoras rotacionadas. O valor numérico do hidrômetro foi extraído diretamente do nome do arquivo da imagem por meio de expressões regulares.

As coordenadas das bounding boxes foram então utilizadas para recortar as regiões de interesse (ROIs) nas imagens RGB originais. Cada (ROI) foi alinhada horizontalmente, redimensionada para  $224 \times 64$  pixels e armazenada diretamente na memória RAM do Google Colab, sem a necessidade de salvar arquivos intermediários. Durante a inspeção visual, constatou-se que algumas imagens estavam invertidas (de cabeça para baixo), sendo corrigidas manualmente por meio de rotação de 180 graus. Amostras com menos de oito dígitos visíveis foram descartadas, resultando em 1.215 imagens válidas do dataset principal.

Cada (ROI) contendo oito dígitos foi segmentada em oito partes iguais, utilizando caixas delimitadoras fixas, gerando imagens individuais de dígitos e seus respectivos rótulos. Essas imagens foram redimensionadas para  $28 \times 28$  pixels, tamanho amplamente utilizado em tarefas de classificação de dígitos, como no dataset MNIST. O dataset complementar proposto por Yang e colaboradores (YANG et al., 2019) foi processado da mesma maneira, com a diferença de que cada imagem continha cinco dígitos. Ao final do processo, obteve-se um total de 17.066 imagens de dígitos, sendo 9.248 provenientes do dataset principal e 7.818 do complementar.

As imagens foram organizadas em três conjuntos: treinamento, validação e teste. O conjunto de treinamento incluiu 697 amostras por classe (dígitos de 0 a 9),

enquanto os conjuntos de validação e teste receberam 150 amostras por classe. A divisão foi feita de forma estratificada para garantir o equilíbrio entre as classes, com embaralhamento aleatório controlado pelo parâmetro `random_state = 42`, assegurando a reprodutibilidade dos experimentos.

Todas as imagens de dígitos foram convertidas de RGB para tons de cinza utilizando a fórmula de luminosidade ponderada, dado o cálculo da luminância (Y) a partir dos componentes de cor RGB (vermelho, verde e azul), com pesos específicos para cada componente. O verde (G) tem o maior peso, refletindo a maior sensibilidade do olho humano a essa cor. Essa fórmula é amplamente utilizada em sistemas de vídeo, como no modelo YUV, no qual a luminância (Y) é separada da croma-luminância. Essa separação explora a maior sensibilidade do olho humano ao canal verde em comparação ao azul, permitindo tanto uma compressão mais eficiente quanto a preservação das informações visuais essenciais, como brilho e estrutura dos dígitos (WYSZECKI; STILES, 1982):

$$Y = 0,2989 \cdot R + 0,5870 \cdot G + 0,1140 \cdot B \quad (4.1)$$

Por fim, os rótulos correspondentes dígitos de 0 a 9 foram transformados em vetores utilizando a codificação one-hot encoding, adequada para tarefas de classificação multiclasse com dez categorias.

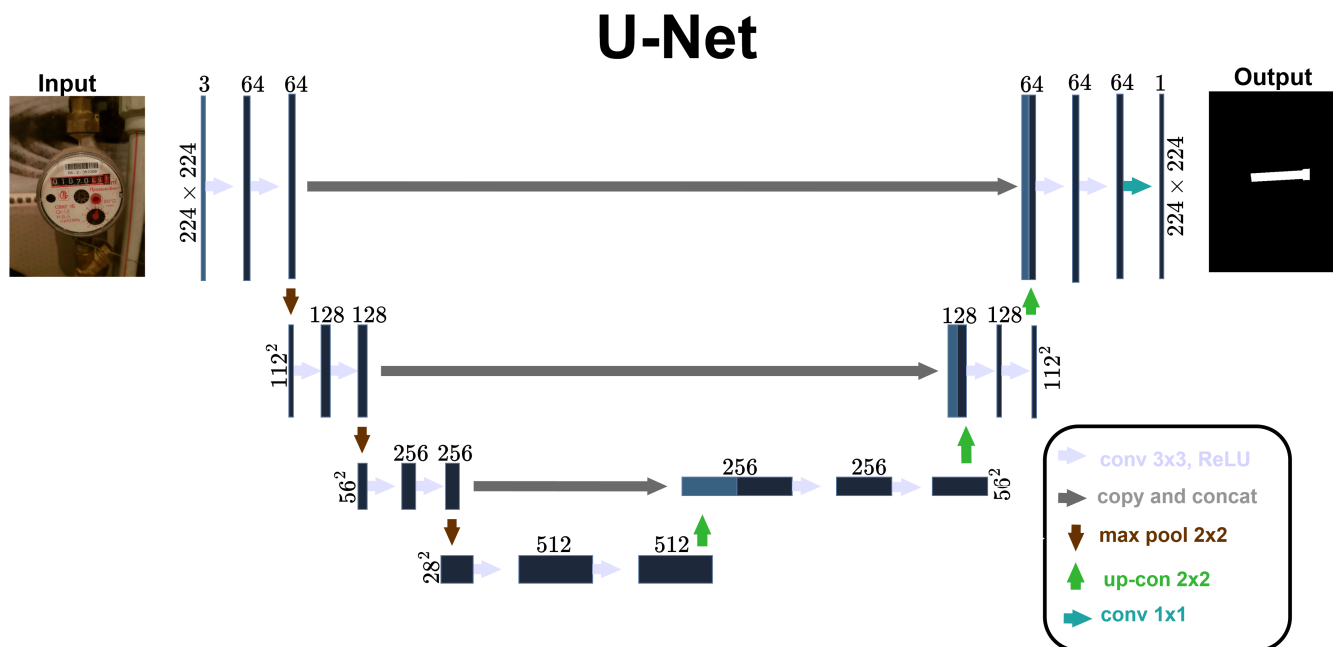
### 4.3 Seleção do Modelo

A seleção dos modelos foi realizada para atender às duas tarefas principais do sistema de leitura automática de hidrômetros: segmentação da região de interesse (ROI) e classificação de dígitos. Para a segmentação, buscou-se um modelo capaz de identificar com precisão a área do mostrador do hidrômetro em imagens RGB, mesmo sob variações de iluminação, ângulos e ruídos. Para a classificação, o modelo deveria reconhecer dígitos individuais com alta acurácia, considerando o formato padronizado das imagens processadas  $28 \times 28$  pixels em tons de cinza. A escolha dos modelos foi baseada em sua eficácia comprovada em tarefas de visão computacional, compatibilidade com os recursos computacionais disponíveis no ambiente Google Colab e simplicidade de implementação, garantindo um equilíbrio entre desempenho e viabilidade prática.

Para a tarefa de segmentação, foi selecionada a arquitetura U-Net, amplamente utilizada em problemas de segmentação de imagens, especialmente em aplicações que requerem alta precisão na delimitação de regiões de interesse. A U-Net é caracterizada por sua estrutura em forma de “U”, que combina camadas de contração downsampling para capturar contexto global com camadas de expansão upsampling para recuperar detalhes espaciais, sendo ideal para processar as máscaras binárias fornecidas pelo dataset

disponibilizado pelo Yandex Toloka (KUCEV, 2020). Sua escolha foi motivada pela robustez em cenários com conjuntos de dados de tamanho moderado (1.244 imagens) e pela capacidade de generalização em condições variadas de captura de imagens.

A Figura 2 apresenta uma ilustração da arquitetura U-Net utilizada no processo de segmentação:



**Figura 2 – Arquitetura da U-Net para segmentação de imagens.**

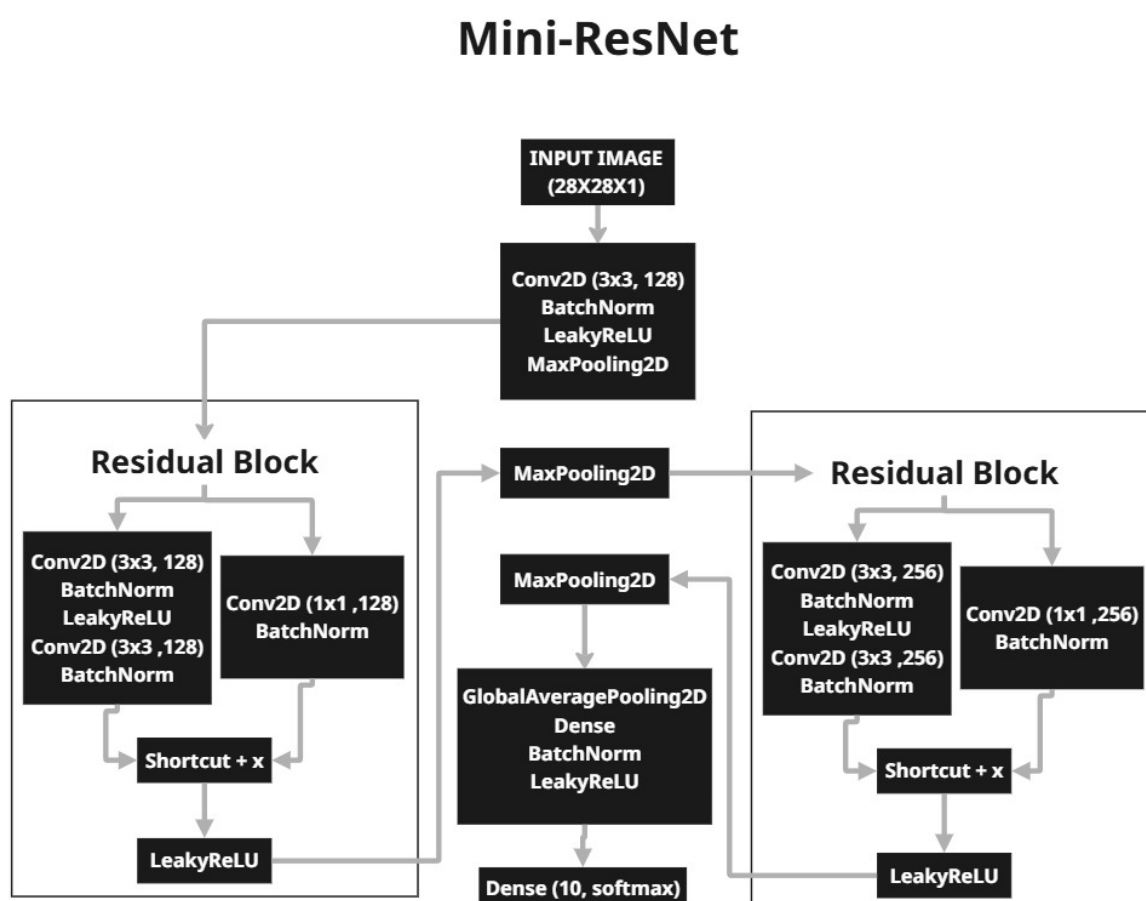
Na etapa de contração, a imagem de entrada passa por blocos formados por convoluções  $3 \times 3$  com ativação ReLU, seguidas de operações de *max pooling*  $2 \times 2$ , que reduzem progressivamente as dimensões espaciais de  $224 \times 224$  até  $28 \times 28$  ao mesmo tempo em que aumentam a profundidade dos mapas de características de 64 até 512 filtros. Esse processo possibilita a captura de representações mais abstratas e globais da imagem.

Na etapa de expansão, realiza-se a operação inversa por meio de up-convolutions  $2 \times 2$ , que aumentam a dimensão espacial dos mapas de ativação. A cada etapa de expansão, as ativações são concatenadas com os mapas correspondentes da etapa de contração por meio de conexões de atalho, de modo a recuperar detalhes espaciais perdidos durante o downsampling. Em seguida, aplicam-se convoluções  $3 \times 3$  com ativação ReLU, refinando as representações combinadas.

Por fim, uma convolução  $1 \times 1$  reduz a dimensionalidade para um único canal de saída, resultando em uma máscara binária que indica a região segmentada de interesse no hidrômetro. Essa estrutura permite à U-Net capturar tanto o contexto global da imagem quanto preservar detalhes locais essenciais para a segmentação precisa.

Para a tarefa de classificação, optou por modelo de CNN convencional para a tarefa de classificação de dígitos. No entanto, os resultados obtidos não foram satisfatórios: a acurácia apresentava uma evolução lenta e a função de perda (loss) não convergia de forma consistente. Esses sintomas indicavam possíveis problemas de otimização, como o desaparecimento do gradiente (vanishing gradient), especialmente nas camadas mais profundas. Para contornar essa limitação, foi adotada uma arquitetura inspirada na ResNet (Mini-ResNet), que utiliza conexões residuais. Esse tipo de arquitetura ajuda a preservar os gradientes durante o treinamento, o que contribui para uma melhor propagação da informação e uma convergência mais estável. Além disso, considerando que o conjunto de dados apresentava ruídos visuais relevantes nas imagens, a Mini-ResNet se mostrou mais robusta, favorecendo a extração de características discriminativas mesmo em condições adversas.

A Figura 3 apresenta uma ilustração da arquitetura Mini-ResNet utilizada no processo de Classificação:



**Figura 3 – Arquitetura da Mini-ResNet para classificação de imagens.**



O processo inicia-se com a entrada de uma imagem com dimensão  $28 \times 28 \times 1$ , representando uma imagem em tons de cinza. Esta imagem é primeiramente processada por uma camada convolucional bidimensional Conv2D com filtros de tamanho  $3 \times 3$  e 128 canais de saída. Essa etapa tem como objetivo extrair características espaciais locais da imagem, como bordas e texturas. Em seguida, aplica-se uma camada de Batch Normalization, que promove a normalização das ativações por mini-batch, contribuindo para a estabilidade e eficiência do treinamento. Posteriormente, é utilizada a função de ativação LeakyReLU, que introduz uma pequena inclinação nos valores negativos, evitando que unidades neuronais fiquem inativas. A primeira fase termina com a aplicação de um MaxPooling2D, o qual reduz a dimensão espacial da imagem ao selecionar os valores máximos em regiões específicas, geralmente de tamanho  $2 \times 2$ .

Na sequência, a arquitetura incorpora o primeiro bloco residual, um dos principais diferenciais da rede. Esse bloco é composto por duas ramificações internas: a principal e a de atalho. Na trilha principal, são aplicadas duas camadas Conv2D  $3 \times 3$ , 128, cada uma seguida por BatchNorm e a primeira delas com ativação LeakyReLU. Simultaneamente, a trilha de atalho processa o mesmo sinal de entrada com uma camada Conv2D  $1 \times 1$ , 128 e BatchNorm, ajustando os canais para garantir compatibilidade dimensional com a trilha principal. As duas saídas são somadas  $\text{shortcut} + x$ , permitindo que a informação original flua diretamente para camadas mais profundas, facilitando o aprendizado em redes mais complexas. Após a soma, aplica-se novamente a função de ativação LeakyReLU.

Finalizado o primeiro bloco residual, a saída é submetida a uma nova camada de MaxPooling2D, que reduz ainda mais a dimensão espacial. Esta etapa antecede a entrada no segundo bloco residual, que segue a mesma estrutura do primeiro, porém com um aumento no número de filtros para 256 canais. A trilha principal contém duas camadas Conv2D  $3 \times 3$ , 256, seguidas de BatchNorm e ativação LeakyReLU na primeira delas. Paralelamente, a trilha de atalho realiza uma Conv2D  $1 \times 1$ , 256 com BatchNorm. As saídas de ambas são somadas e novamente aplicadas à função LeakyReLU, mantendo a consistência da arquitetura residual.

Após esse segundo bloco residual, aplica-se mais uma camada de MaxPooling2D, seguida por uma operação de GlobalAveragePooling2D, que transforma cada mapa de ativação em um único valor por canal, por meio da média global de seus valores espaciais. Em seguida, essa saída é processada por uma camada densa Dense fully connected, que consiste em uma camada de neurônios onde cada unidade recebe entrada de todos os neurônios da camada anterior, permitindo a combinação linear de todas as características extraídas previamente. Esses neurônios anteriores correspondem exatamente aos valores médios de cada canal produzidos pela camada GlobalAveragePool-

ling2D, resumindo as informações espaciais dos mapas de ativação. A camada Dense é acompanhada por normalização em batch e ativação LeakyReLU.

Por fim, a arquitetura é finalizada com uma camada densa com 10 unidades e ativação softmax Dense 10 softmax, responsável pela predição da classe da imagem de entrada. As entradas dessa última Dense são os neurônios produzidos pela Dense anterior, permitindo que a rede combine todas as características extraídas para realizar a classificação final. Cada uma das 10 unidades corresponde a uma classe possível, e a função softmax transforma os valores da camada anterior em probabilidades, assegurando que a soma total das saídas seja igual a 1, e atribuindo maior probabilidade à classe mais compatível com a entrada.

#### 4.4 Treinamento do Modelo

O treinamento dos modelos foi conduzido com o objetivo de otimizar o desempenho do sistema de leitura automática de hidrômetros nas tarefas de segmentação da região de interesse (ROI) e classificação de dígitos. Todas as etapas foram realizadas no ambiente Google Colab, utilizando a GPU Tesla T4 para acelerar o processamento, com Python (3.11.13), TensorFlow (2.18.0) e Keras (3.8.0), visando garantir eficiência computacional e robustez em cenários reais. Os hiperparâmetros foram definidos com base em experimentos preliminares, considerando o desempenho dos modelos e as limitações de recursos do ambiente. A semente aleatória `random_state = 42` foi utilizada em todas as divisões de dados e inicializações de pesos, assegurando a reprodutibilidade dos resultados.

Para a tarefa de segmentação, o modelo U-Net foi treinado por 50 épocas, com um tamanho de lote de oito imagens. A função de perda utilizada foi a entropia cruzada binária, adequada para problemas de segmentação com máscaras binárias. O otimizador Adam foi configurado com uma taxa de aprendizado inicial de 0,00001, escolhida para garantir uma convergência estável em um dataset de tamanho reduzido (1.244 imagens). O conjunto de dados foi dividido em treinamento (987 imagens), validação (247 imagens) e teste (10 imagens), com embaralhamento aleatório. Para melhorar a capacidade de generalização, foram aplicadas técnicas de aumento de dados, incluindo rotações, deslocamentos horizontais e verticais, além de variações de brilho.

Para a tarefa de classificação, a Mini-ResNet foi treinada por 22 épocas, com um tamanho de lote de quatro imagens. A função de perda escolhida foi a entropia cruzada categórica, adequada para problemas de classificação multiclasse, dígitos de 0 a 9. O otimizador Adam foi utilizado com a taxa de aprendizado padrão (0,001). A pequena dimensão do lote foi adotada para melhorar a exploração do espaço de parâmetros, enquanto o aumento de dados foi aplicado por meio de um gerador datagen,

incluindo rotações, deslocamentos e variações de intensidade. O conjunto foi composto por 17.066 imagens de dígitos, divididas em treinamento até 697 amostras por classe, validação 150 amostras por classe e teste 150 amostras por classe.

#### 4.5 Avaliação do Modelo

A avaliação dos modelos será conduzida para verificar o desempenho do sistema de leitura automática de hidrômetros nas tarefas de segmentação da região de interesse (ROI) e classificação de dígitos, com ênfase na robustez frente a variações de iluminação, ângulos e ruídos. Serão empregadas métricas específicas para cada tarefa, aplicadas aos conjuntos de teste separados, no ambiente Google Colab, visando uma análise quantitativa e qualitativa robusta do desempenho dos modelos.

Para a tarefa de segmentação, o modelo U-Net será avaliado no conjunto de teste composto por 10 imagens do dataset disponibilizado pelo Yandex Toloka (KU-CEV, 2020). As métricas empregadas para a avaliação quantitativa incluem:

- **Acurácia de pixel**, que calcula a proporção de pixels corretamente classificados como pertencentes ou não à região de interesse (ROI).
- **Função de perda (loss)** baseada na entropia cruzada binária, que quantifica o erro do modelo na predição das máscaras binárias.
- **Interseção sobre União (IoU)**, que mede a sobreposição entre as máscaras preditas e as de referência, fornecendo uma avaliação precisa da qualidade da segmentação.
- **Coefficiente de Dice**, que avalia a similaridade entre as máscaras predita e real, enfatizando a interseção das regiões segmentadas.
- **Recall**, que mede a capacidade do modelo de identificar corretamente os pixels positivos (verdadeiros positivos em relação aos reais positivos).
- **F1 Score**, que representa a média harmônica entre precisão e recall, oferecendo uma visão balanceada do desempenho do modelo, especialmente útil em cenários com classes desbalanceadas.

Além da avaliação quantitativa, foi realizada uma inspeção visual das máscaras preditas para analisar qualitativamente o desempenho do modelo em cenários desafiadores, como variações de iluminação, ruído e formas irregulares.

Para a tarefa de classificação, a rede neural convolucional (CNN) foi avaliada no conjunto de teste composto por 150 imagens correspondentes aos dígitos de 0 a 9. As métricas empregadas para a avaliação quantitativa incluem:

- **Acurácia**, que mede a proporção de dígitos corretamente classificados pelo modelo.

- **Função de perda (loss)** baseada na entropia cruzada categórica, que quantifica o erro do modelo na predição das classes dos dígitos.

## 5 Resultados

Este capítulo apresenta os resultados obtidos com os modelos de segmentação e classificação aplicados à leitura automática de hidrômetros. São expostos os valores das principais métricas de avaliação utilizadas, além de gráficos que ilustram o desempenho durante o treinamento e exemplos visuais das predições realizadas.

### 5.1 Desempenho do Modelo de Segmentação

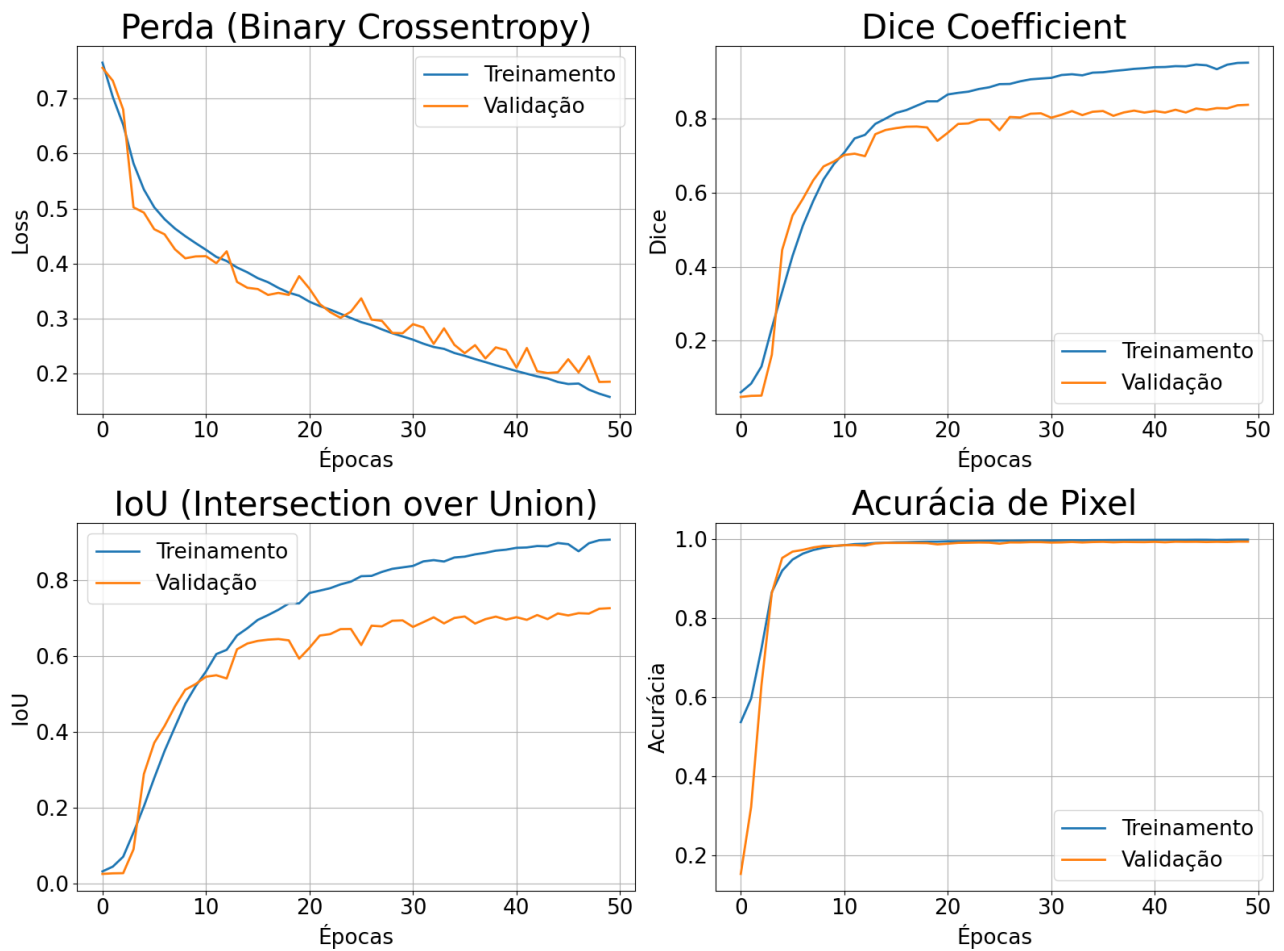
O modelo U-Net foi avaliado utilizando as métricas de perda (Loss), coeficiente de Dice (Dice Coefficient), interseção sobre união (IoU), acurácia de pixel (Pixel Accuracy), revocação (Recall) e pontuação F1 (F1 Score) no conjunto de teste.

A Tabela 2 apresenta os resultados obtidos, demonstrando que o modelo alcançou uma perda (Loss) de 0,18, um coeficiente de Dice de 0,84, um valor de IoU igual a 0,73, uma acurácia de pixel de 0,99, um valor de Recall de 0,81 e uma pontuação F1 (F1 Score) de 0,84. Esses resultados indicam um bom desempenho do modelo U-Net na tarefa de segmentação de mostradores.

Métricas	Conjunto de Teste
Loss (Binary Crossentropy) ↓	0,18
Coeficiente de Dice ↑	0,84
IoU (Interseção sobre União) ↑	0,73
Pixel Accuracy ↑	0,99
Recall ↑	0,81
F1 Score ↑	0,84

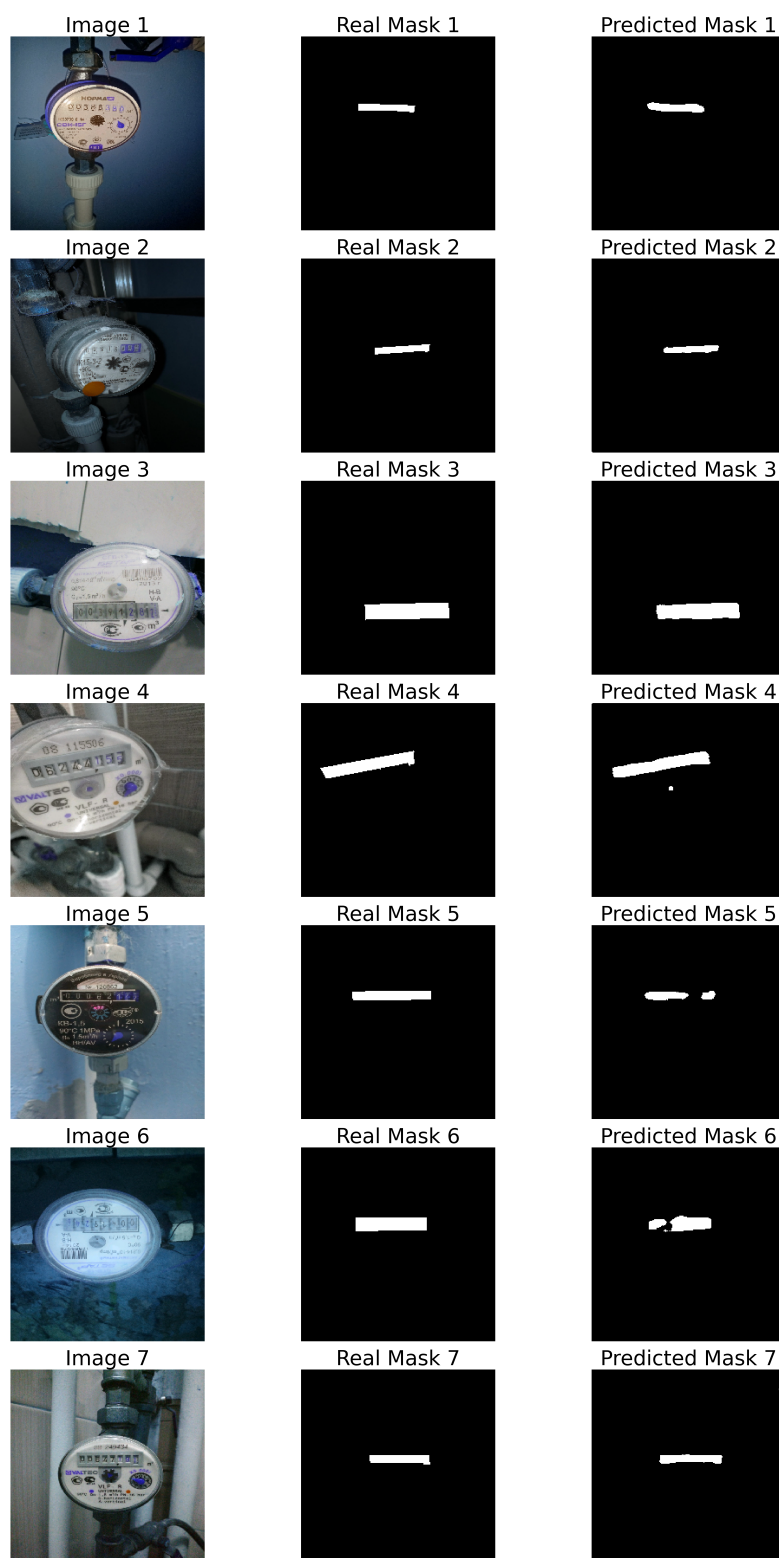
**Tabela 2 – Métricas de desempenho do modelo U-Net no conjunto de teste para segmentação de mostradores.**

A Figura 4 apresenta a evolução das principais métricas do modelo U-Net, incluindo perda (Loss), coeficiente de Dice, interseção sobre união (IoU) e acurácia de pixel.



**Figura 4 – Curvas de aprendizado do modelo U-Net para as principais métricas nos conjuntos de treino e validação ao longo de 50 épocas.**

A Figura 5 ilustra os resultados obtidos com o modelo U-Net no processo de segmentação. A imagem original de entrada, a máscara real (ground truth) e a máscara gerada pelo modelo.



**Figura 5 – As imagens utilizadas abrangem diferentes condições de campo, como variações de iluminação, presença de reflexos no visor e obstruções parciais, representando uma diversidade de cenários reais.**

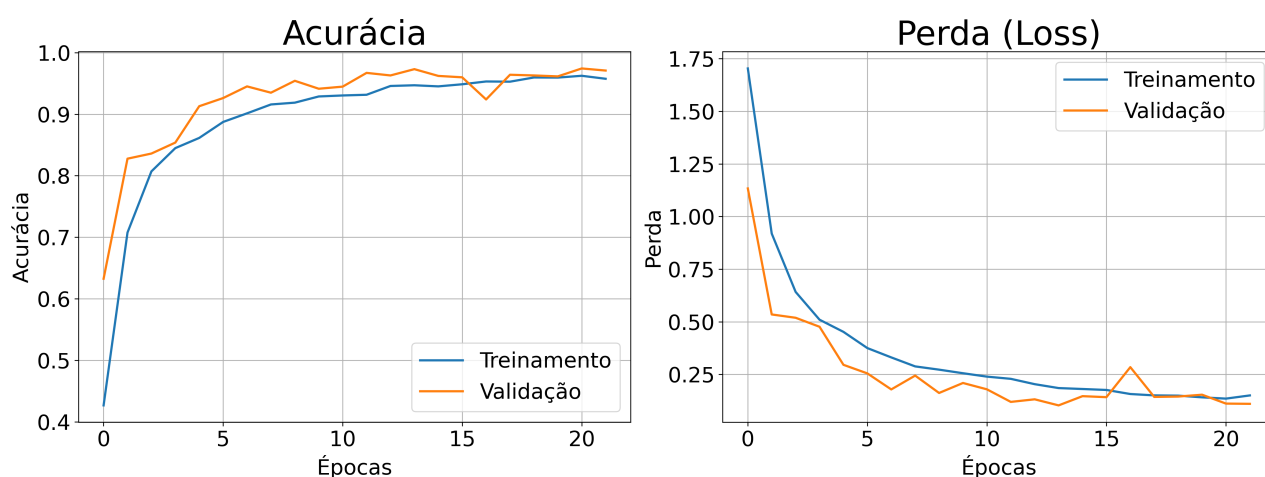
## 5.2 Desempenho do Modelo de Classificação

O desempenho do modelo Mini-ResNet para classificação de dígitos foi avaliado utilizando as métricas de acurácia e perda (Loss) nos conjuntos de treino, validação e teste, conforme apresentado na Tabela 3. Observa-se que, no conjunto de teste, o modelo obteve uma acurácia de 0,97 e uma perda de 0,14, evidenciando sua boa capacidade de generalização para dados não vistos durante o treinamento.

**Tabela 3 – Métricas de desempenho Mini-Resnet para classificação de dígitos nos conjuntos de treino, validação e teste.**

Métricas	Conjunto de Teste
Accuracy ↑	0,97
Loss ↓	0,14

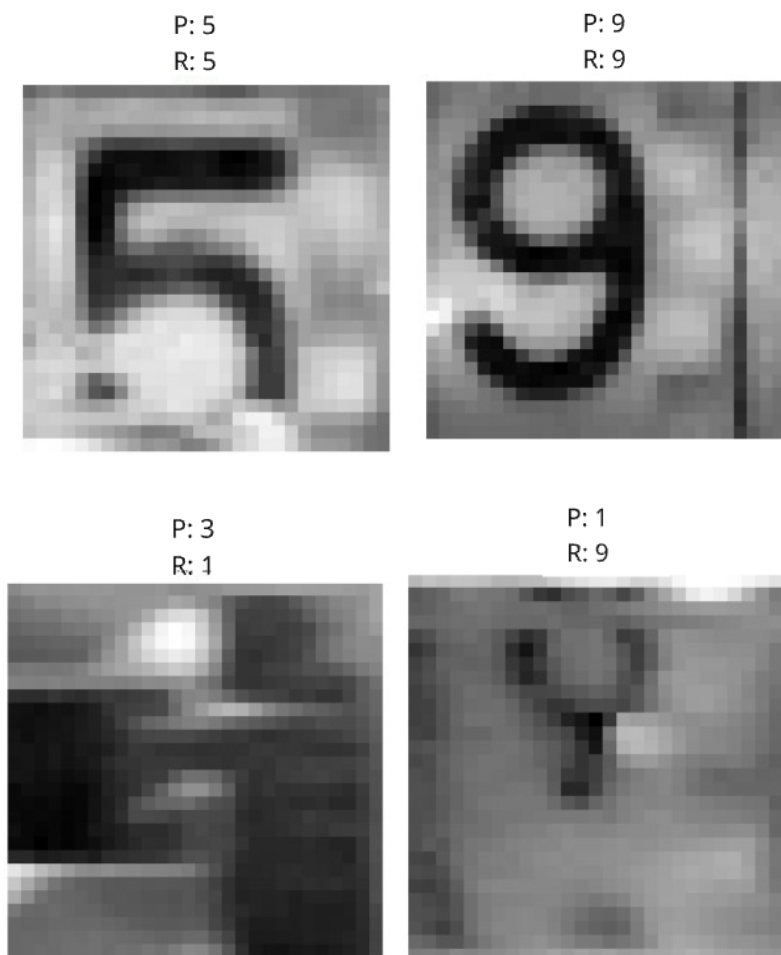
A Figura 6 mostra as curvas de aprendizado Mini-Resnet, com a evolução da acurácia e da perda ao longo de 22 épocas.



**Figura 6 – Curvas de aprendizado Mini-Resnet: Acurácia, Perda (Loss) nos conjuntos de treino e validação ao longo de 22 épocas.**

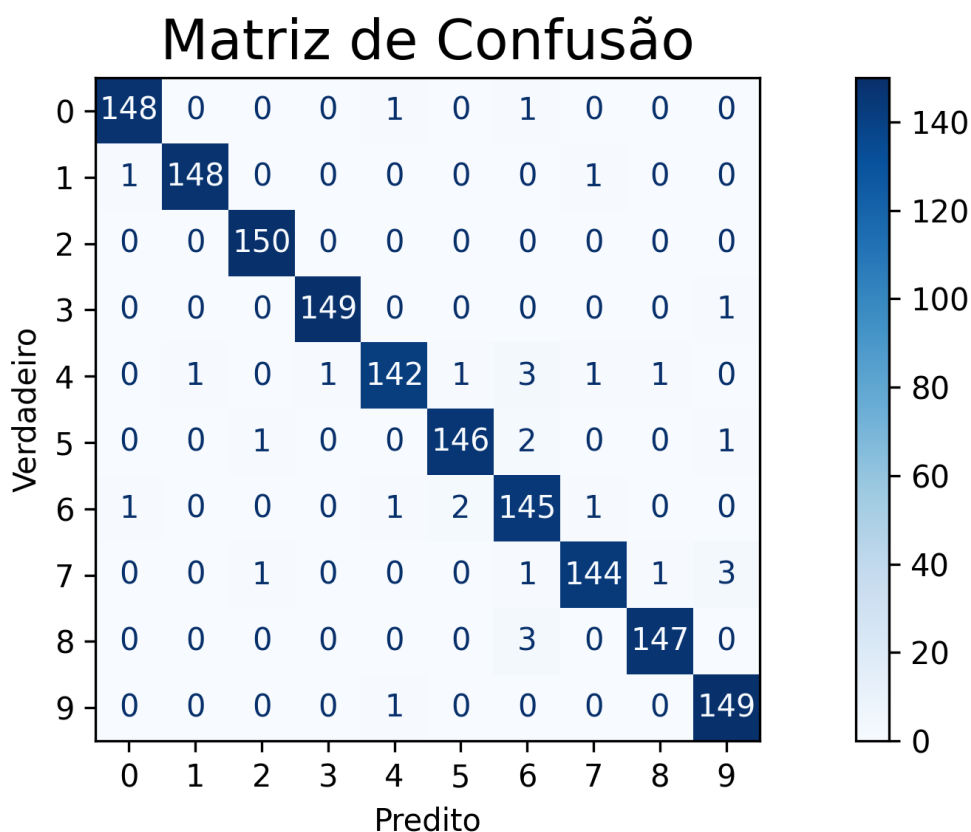
A Figura 7 apresenta alguns exemplos de imagens classificadas pelo modelo Mini-Resnet, juntamente com os respectivos rótulos reais e previsões geradas.





**Figura 7 – Exemplos de imagens classificadas pelo modelo Mini-ResNet, incluindo casos de acerto e erro.**

A Figura 8 apresenta a matriz de confusão obtida no conjunto de teste. Nela, é possível visualizar a quantidade de acertos e erros cometidos pelo modelo em cada classe.



**Figura 8 – Matriz de confusão do modelo Mini-ResNet no conjunto de teste. A maior parte das amostras foi corretamente classificada (valores na diagonal principal), com poucos erros, sendo o mais frequente 4 confundido com 3 (3 vezes).**

### 5.3 Comparação com Trabalhos Correlatos

Esta seção compara os resultados obtidos neste trabalho com os de estudos correlatos, evidenciando a competitividade da abordagem proposta. A utilização de arquiteturas leves, como uma Mini-ResNet para classificação e uma U-Net simplificada para segmentação, ambas treinadas do zero, demonstra desempenho robusto com menor custo computacional. Apesar das diferenças em objetivos, conjuntos de dados e metodologias, os resultados alcançados destacam a eficiência e a capacidade de generalização do modelo para aplicações práticas, especialmente em cenários com recursos limitados.

Na tarefa de segmentação da região de interesse, a U-Net desenvolvida neste trabalho alcançou resultados notáveis: acurácia de pixel de 99,27%, IoU de 0,73, Dice Coefficient de 0,84, Recall de 0,81 e F1 Score de 0,84. A Tabela 4 compara essas métricas com as de estudos correlatos.

**Tabela 4 – Comparação de métricas de segmentação da região de interesse**

Trabalho	Acurácia de Pixel (%) ↑	Dice ↑	F1 Score ↑
(SLYUSAR et al., 2022)	85,20	–	–
(RADYHIN, 2023)	–	0,63	–
(FEI et al., 2023)	–	–	0,94
<b>Este trabalho</b>	<b>99,27</b>	<b>0,82</b>	<b>0,84</b>

A análise comparativa revela a superioridade da abordagem proposta em relação a (SLYUSAR et al., 2022), que reportaram uma acurácia de pixel de 85,20%. Comparado ao modelo de (RADYHIN, 2023), baseado em U-Net com EfficientNetB0, nosso Dice Coefficient de 0,82 supera significativamente o valor de 0,63, apesar de utilizarmos uma arquitetura mais leve. Por outro lado, (FEI et al., 2023) obteve um F1 Score de 0,94, superior ao nosso, mas depende de uma arquitetura mais complexa e técnicas avançadas de aumento de dados. A U-Net leve proposta neste trabalho equilibra alta performance com simplicidade, destacando-se como uma solução eficiente.

Os resultados apresentados demonstram que a abordagem proposta é altamente competitiva, mesmo frente a modelos mais elaborados. A combinação de desempenho robusto, simplicidade arquitetural e baixo custo computacional torna a solução ideal para aplicações embarcadas ou com restrições de recursos, oferecendo um equilíbrio excepcional entre eficácia e viabilidade prática.

Na tarefa de classificação de dígitos, este trabalho obteve uma acurácia de 0,97, superando ligeiramente os 0,96 reportados por Saetan e Dittakan (2024) com o modelo YOLOv5, como mostrado na Tabela 5.

**Tabela 5 – Comparação de acurácia na tarefa de classificação**

Trabalho	Arquitetura	Acurácia (%) ↑
(SAETAN; DITTAKAN, 2024)	YOLOv5	96,6
<b>Este trabalho</b>	Mini-ResNet (treinada do zero)	<b>96,7</b>

Embora alguns estudos correlatos alcancem acurácias próximas, recorrem frequentemente a arquiteturas mais complexas, pré-treinadas ou treinadas em conjuntos de dados maiores. Em contraste, a abordagem proposta neste trabalho destaca-se por atingir desempenho competitivo com uma Mini-ResNet simples, treinada do zero, o que reforça sua eficiência e adequação para aplicações com restrições computacionais.

## 6 Conclusão

Este trabalho desenvolveu um sistema automatizado para leitura de dígitos em hidrômetros, utilizando visão computacional e aprendizado de máquina. Estruturado em duas etapas, segmentação da área de interesse com a arquitetura U-Net e classificação de dígitos por uma rede neural convolucional, o sistema alcançou resultados expressivos. A segmentação obteve acurácia de 99%, Intersection over Union (IoU) de 0,73, Dice Coefficient de 0,84, Recall de 0,81 e F1 Score de 0,84, enquanto a classificação atingiu 97% de acurácia e loss de 0,09, beneficiada por um pré-processamento otimizado e um conjunto de dados mais robusto. Desafios como inclinações nas imagens, reflexos no visor plástico e ruídos por sombras ou linhas verticais foram observados, mas não comprometeram a eficácia geral do modelo.

A relevância do estudo reside na automação de um processo real, com potencial para reduzir erros humanos e custos operacionais de empresas de saneamento, além de melhorar a gestão de recursos hídricos em regiões dependentes de hidrômetros analógicos. Essa solução pode beneficiar consumidores ao promover maior precisão na medição e transparência nas contas, especialmente em países em desenvolvimento. Academicamente, o trabalho contribui para o avanço da visão computacional, demonstrando a aplicabilidade da U-Net em cenários com condições visuais adversas, um desafio recorrente na área.

Para trabalhos futuros, sugere-se mitigar os desafios observados com técnicas de normalização de imagens, como correção de perspectiva para inclinações e filtros para redução de reflexos. Arquiteturas mais robustas, como YOLO ou EfficientNet, podem melhorar o desempenho em condições complexas. Expandir o conjunto de dados com diferentes marcas de hidrômetros e cenários de iluminação é igualmente crucial. Por fim, propõe-se adaptar a metodologia para medidores de luz e gás e desenvolver um aplicativo móvel para leitura em tempo real, ampliando o impacto prático da solução.

## REFERÊNCIAS

ALPAYDIN, E. *Introduction to Machine Learning*. [S.l.]: MIT Press, 2020. ISBN: 978-0262043793.

Anyline GmbH. *Anyline OCR for Meter Reading*. 2023. <<https://anyline.com/products/ocr-meter-reading>>. Acesso em: 10 jun. 2025.

AZEEM, A. et al. Um sistema robusto de leitura automática de medidores baseado em mask-rcnn. In: *Conferência Internacional IEEE 2020 sobre Avanços em Engenharia Elétrica e Aplicações Computacionais (AEECA)*. [S.l.: s.n.], 2020. p. 209–213.

BANK, W. *Water Resources Management: Sector Results Profile*. World Bank Group, 2016. Disponível em: <<https://www.worldbank.org/en/results/2013/04/15/water-resources-management-results-profile>>.

BISHOP, C. M. *Pattern Recognition and Machine Learning*. New York: Springer, 2006. Disponível em: <<https://link.springer.com/book/9780387310732>>.

Clou Global. *Adoção de Medidores Inteligentes: Países Desenvolvidos x Países em Desenvolvimento*. 2025. Acesso em: 23 jun. 2025. Disponível em: <<https://clouglobal.com/pt/adocao-de-medidores-inteligentes-paises-desenvolvidos-x-paises-em-desenvolvimento/>>.

FEI, R. et al. An improved method for digital water meter reading area segmentation based on u 2-net. *Academic Journal of Computing & Information Science*, v. 6, n. 12, p. 33–44, 2023. Disponível em: <<https://doi.org/10.25236/AJCIS.2023.061204>>.

GOODFELLOW, I.; BENGIO, Y.; COURVILLE, A. *Deep Learning*. Cambridge: MIT Press, 2016. Acesso em: 19 set. 2025. Disponível em: <<https://mitpress.mit.edu/9780262035613/deep-learning/>>.

HE, K. et al. Deep residual learning for image recognition. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. [s.n.], 2016. p. 770–778. Disponível em: <[https://openaccess.thecvf.com/content\\_cvpr\\_2016/html/He\\_Deep\\_Residual\\_Learning\\_CVPR\\_2016\\_paper.html](https://openaccess.thecvf.com/content_cvpr_2016/html/He_Deep_Residual_Learning_CVPR_2016_paper.html)>.

IMRAN, M. et al. Image-based automatic energy meter reading using deep learning.

*Computers, Materials & Continua*, v. 74, n. 1, p. 203–216, 2023. Disponível em: <<https://doi.org/10.32604/cmc.2023.029834>>.

Inovamobil. *Soluções para Gestão de Água*. 2023. Disponível em: <https://inovamobil.com.br/blog/leitura-de-hidrometros-de-agua-processo-manual-vs-automatizado/>. Acesso em: 10 jun. 2025.

KRIZHEVSKY, A.; SUTSKEVER, I.; HINTON, G. E. Imagenet classification with deep convolutional neural networks. *Advances in Neural Information Processing Systems*, v. 25, p. 1097–1105, 2012. Disponível em: <<https://papers.nips.cc/paper/2012/file/c399862d3b9d6b76c8436e924a68c45b-Paper.pdf>>.

KUCEV, R. *Water Meters Dataset*. [S.l.]: Kaggle, 2020. <<https://www.kaggle.com/datasets/tapakah68/yandextoloka-water-meters-dataset>>. A dataset de 1244 imagens de hidrômetros com máscaras de segmentação e rótulos OCR, coletado via Yandex.Toloka.

LECUN, Y. et al. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, v. 86, n. 11, p. 2278–2324, 1998. Disponível em: <<http://yann.lecun.com/exdb/mnist/>>.

LIMEIRA, J. de A. *Aplicação da Arquitetura U-Net para Segmentação de Mapas*. 2021. <<https://repositorio.ufpb.br/jspui/bitstream/123456789/31596/1/Jackson%20de%20Araujo%20Limeira%20-%20TCC.pdf>>. Acesso em: 31 maio 2025.

NETZER, Y. et al. Reading digits in natural images with unsupervised feature learning. *NIPS Workshop on Deep Learning and Unsupervised Feature Learning*, 2011. Disponível em: <[http://ufldl.stanford.edu/housenumbers/nips2011\\_housenumbers.pdf](http://ufldl.stanford.edu/housenumbers/nips2011_housenumbers.pdf)>.

RADYHIN, A. *OCR-Water-Meter*. 2023. <<https://github.com/Andrii-Radyhin/OCR-Water-Meter>>. Acesso em: 13 jul. 2025.

RONNEBERGER, O.; FISCHER, P.; BROX, T. U-net: Convolutional networks for biomedical image segmentation. *Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, p. 234–241, 2015. Disponível em: <[https://doi.org/10.1007/978-3-319-24574-4\\_28](https://doi.org/10.1007/978-3-319-24574-4_28)>.

SAETAN, T.; DITTAKAN, N. Thailand water meter reading using convolutional neural networks from smartphone imagery. In: *2024 21st International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON)*. IEEE, 2024. Disponível em: <<https://ieeexplore.ieee.org/document/10480091>>.

SALOMON, G.; LAROCA, R.; MENOTTI, D. *Deep Learning for Image-based Automatic Dial Meter Reading: Dataset and Baselines*. ACM, 2020. 1809–1816 p.

Disponível em: <<https://ieeexplore.ieee.org/document/9207318>>.

SLYUSAR, V. et al. Segmentation of analogue meter readings using neural networks. In: CEUR WORKSHOP PROCEEDINGS. *Proceedings of the 44th International Workshop on Modern Machine Learning Technologies and Data Science (MoMLet+DS 2022)*. 2022. p. 165–175. Disponível em: <<https://ceur-ws.org/Vol-3312/paper14.pdf>>.

SOUZA, G. A. de. *Leitura Automática de Hidrômetros Usando Redes Neurais Convolucionais*. 2022. <<https://repositorio.ufpe.br/handle/123456789/52617>>. Trabalho de Conclusão de Curso (Graduação em Engenharia Elétrica) — Universidade Federal de Uberlândia, 2022. Acesso em: 31 maio 2025.

SZELISKI, R. *Computer Vision: Algorithms and Applications*. Springer, 2010. Disponível em: <<http://szeliski.org/Book>>.

TERVEN, J. et al. Uma pesquisa abrangente sobre funções de perda e métricas em aprendizado profundo. *Artificial Intelligence Review*, v. 58, p. 195, 2025. Aceito: 13 de março de 2025, Publicado: 11 de abril de 2025. Disponível em: <<https://doi.org/10.1007/s10462-025-11198-7>>.

WYSZECKI, G.; STILES, W. S. *Color Science: Concepts and Methods, Quantitative Data and Formulae*. Wiley, 1982. Acesso em: 23 ago. 2025. ISBN 0471021067. Disponível em: <<https://archive.org/details/colorscienceconc00unse>>.

YANG, F. et al. Fully convolutional sequence recognition network for water meter number reading. *IEEE Access*, 2019.