

PONTIFÍCIA UNIVERSIDADE CATÓLICA DE GOIÁS
ESCOLA POLITÉCNICA E DE ARTES / ENGENHARIA DE CONTROLE E AUTOMAÇÃO
Trabalho Final de Curso

André Victor de Oliveira Barros dos Santos

**INTEGRAÇÃO DE IA E NB-IOT NO MONITORAMENTO DE
EQUIPAMENTOS INDUSTRIAIS PARA CONTROLE DE QUALIDADE**

Trabalho Final de Curso como parte dos requisitos para
obtenção do título de bacharel em Engenharia de
Controle e Automação apresentado à Pontifícia
Universidade Católica de Goiás.

BANCA EXAMINADORA:

Prof. Dr. Marcos Antônio de Sousa	– Orientador. POLI-PUC Goiás.
Prof. Dr. Antônio Marcos Melo Medeiros	– Avaliador. POLI-PUC Goiás.
Prof. Dr. Marcos Lajovic Carneiro	– Avaliador. POLI-PUC Goiás.

Goiânia, 3 de dezembro de 2025.

Integração de IA e NB-IoT no Monitoramento de Equipamentos Industriais para Controle de Qualidade

ANDRÉ VICTOR DE OLIVEIRA BARROS DOS SANTOS, MARCOS A. SOUSA

Abstract— This work presents the development of an innovative predictive monitoring system for industrial inkjet printers, specifically designed to detect printing failures like "nozzle missing" in large-format K-JET machines (3.2 meters). The system leverages Artificial Intelligence (AI) in an Intelbras IM1 camera to identify anomalies, generating real-time UDP data. A robust NB-IoT (Narrowband-IoT) device, the HTM32, captures these network data and transmits critical alerts via a 5G cellular network to the Telegram messaging application. This approach ensures continuous and reliable communication, even if local internet infrastructure fails. The objective is to enhance preventive maintenance strategies, minimize material waste, and reduce downtime in industrial printing environments. Two main development lines are explored: first, the implementation of an efficient and highly accurate AI-based vision system for anomaly detection; second, the integration of NB-IoT on a 5G network for resilient real-time alerting. The performance of the developed system is validated through tests replicating real operational scenarios, from data acquisition at the printer to the mobile user interface for notifications. The results achieved, demonstrating the system's effectiveness in providing rapid and reliable alerts, are presented and discussed.

Keywords – *Industrial IoT, NB-IoT, 5G, Artificial Intelligence, Predictive Maintenance, Inkjet Printing, Real-time Monitoring.*

Resumo – Este trabalho apresenta o desenvolvimento de um sistema de monitoramento preditivo para impressoras industriais de jato de tinta, especificamente projetado para detectar falhas de impressão como "nozzle missing" em máquinas K-JET de grande formato (3.2 metros). O sistema emprega Inteligência Artificial (IA) em uma câmera Intelbras IM1 para identificar anomalias, gerando dados UDP em tempo real. Um dispositivo NB-IoT (Narrowband-IoT), o HTM32, captura esses dados da rede e transmite alertas críticos via rede celular 5G para o aplicativo de mensagens Telegram. Essa abordagem garante comunicação contínua e confiável, mesmo em caso de falha na infraestrutura de internet local. O objetivo é aprimorar as estratégias de manutenção preventiva, minimizando o desperdício de material e reduzindo o tempo de inatividade em ambientes de impressão industrial. Duas linhas principais de desenvolvimento são exploradas: primeiro, a implementação de um sistema de visão baseado em IA eficiente e de alta precisão para detecção de anomalias; segundo a integração do NB-IoT em uma rede 5G para alertas resilientes em tempo real. O desempenho do sistema desenvolvido é validado através de testes que reproduzem cenários operacionais reais, desde a aquisição de dados na impressora até a interface móvel do usuário para notificações. Os resultados alcançados, demonstrando a eficácia do sistema em fornecer alertas rápidos e confiáveis, são apresentados e discutidos.

Palavras-chave – *IoT Industrial, NB-IoT, 5G, Inteligência Artificial, Manutenção Preditiva, Impressão Jato de Tinta, Monitoramento em Tempo Real.*

I. INTRODUÇÃO

O cenário industrial contemporâneo é caracterizado por uma busca incessante por otimização de processos, redução de custos e aumento da produtividade. Nesse contexto, o avanço tecnológico, especialmente nas áreas de Internet das Coisas (IoT) e Inteligência Artificial (IA), tem pavimentado o caminho para a emergência da Indústria 4.0 [1]. Esta nova era industrial se baseia na interconexão de sistemas ciberfísicos, análise de dados em tempo real e automação inteligente, transformando fundamentalmente as operações de manufatura [2].

A IoT, em particular, tem se estabelecido como um pilar central da transformação digital, permitindo a conexão de objetos físicos à internet para coleta e troca de dados. Essa capacidade de interligar dispositivos físicos e virtuais em uma rede global possibilita que "coisas" adquiram, compartilhem e armazenem informações sobre o ambiente ao seu redor, gerando volumes massivos de dados que, quando processados e analisados, se convertem em inteligência e serviços de valor inestimável [3]. A aplicação da IoT estende-se por diversos setores, desde a automação residencial até a agricultura, saúde e, crucialmente, o ambiente industrial [4].

No âmbito da indústria, a adoção de soluções IoT para monitoramento e controle preditivo está em ascensão. Em processos produtivos que dependem de equipamentos complexos e de alto desempenho, como as impressoras industriais de jato de tinta de grande formato (e.g., K-JET 3204), a operação contínua e sem falhas é um requisito crítico. Essas máquinas, que representam um investimento substancial e são vitais para a produção em larga escala, exigem monitoramento constante devido à sua complexidade operacional e ao alto custo associado a interrupções ou desperdícios de material [5]. Anomalias, como a ausência de bicos injetores ("nozzle missing"), podem levar a perdas financeiras significativas, retrabalho custoso e paralisações inesperadas na linha de produção [6]. Métodos de monitoramento convencionais frequentemente se mostram insuficientes para garantir a agilidade e a precisão necessárias na detecção e correção desses problemas.

Para garantir a confiabilidade em ambientes industriais, onde a estabilidade da conectividade é fundamental, a escolha da tecnologia de comunicação é crucial. Tecnologias como o NB-IoT (Narrowband-IoT), parte integrante dos padrões de comunicação celular 5G, destacam-se pela sua robustez e confiabilidade. Diferentemente de redes Wi-Fi ou Ethernet que podem ser suscetíveis a interrupções por falhas de infraestrutura local, o NB-IoT opera sobre a infraestrutura celular, utilizando chips SIM para garantir uma comunicação

estável e segura, mesmo em cenários de falha da internet local, assegurando a transmissão contínua de dados críticos [7, 8]. Esta resiliência é um fator decisivo para aplicações de monitoramento onde a ininterruptão da comunicação é vital para a manutenção preditiva, a segurança operacional e o controle de qualidade.

Diante desse desafio, este Trabalho de Conclusão de Curso propõe o desenvolvimento de um sistema inovador de monitoramento preditivo para impressoras industriais. A solução integra visão computacional com Inteligência Artificial para a detecção automática de falhas, especificamente a ausência de bicos injetores, utilizando uma câmera Intelbras IM1. As informações de falha, geradas como dados UDP na rede local, são capturadas por um dispositivo NB-IoT (HTM32). Este, por sua vez, aproveita a robustez e a abrangência da conectividade 5G (via NB-IoT) para transmitir alertas em tempo real para o aplicativo Telegram, notificando os operadores instantaneamente sobre a necessidade de intervenção.

O principal objetivo deste trabalho é demonstrar a viabilidade e eficácia da integração de tecnologias de IA em câmeras de segurança com a confiabilidade e eficiência do NB-IoT operando sobre uma infraestrutura 5G, para criar um sistema de alerta rápido e robusto que otimize a manutenção preventiva em ambientes industriais de alto valor e, consequentemente, assegure o controle de qualidade do processo. A justificativa para este estudo reside na capacidade de mitigar os impactos negativos de falhas de impressão, reduzindo o desperdício de material, aumentando a disponibilidade dos equipamentos e, consequentemente, elevando a eficiência operacional, mesmo em condições adversas de conectividade local. Quanto mais rápida for a identificação da falha e mais garantida a comunicação do alerta, mais ágil e eficaz será a ação corretiva, minimizando perdas e garantindo a continuidade da produção.

Este documento está estruturado nas seguintes seções: a seção II apresenta a fundamentação teórica, abordando conceitos essenciais da Indústria 4.0, Internet das Coisas (com foco em NB-IoT e 5G), Inteligência Artificial aplicada à visão computacional e trabalhos correlatos. A seção III descreve a metodologia de projeto, detalhando os componentes de hardware (câmera Intelbras IM1 e dispositivo HTM32) e software utilizados na implementação do sistema. A seção IV expõe os resultados obtidos nos testes e na validação do protótipo, acompanhados de suas discussões. Finalmente, a seção V apresenta as conclusões gerais do trabalho, destacando as contribuições e propondo futuras linhas de pesquisa.

II. REVISÃO BIBLIOGRÁFICA

A Indústria 4.0 representa a quarta grande revolução industrial, caracterizada pela fusão de tecnologias digitais, físicas e biológicas. Esse paradigma promove a criação de "fábricas inteligentes", onde máquinas, sistemas e produtos são interconectados e comunicam-se entre si, permitindo a automação flexível, a tomada de decisão descentralizada e a personalização em massa [1]. O objetivo central é otimizar processos produtivos, aumentar a eficiência e reduzir custos, ao integrar tecnologias como IoT, Big Data, Inteligência Artificial, Manufatura Aditiva e sistemas ciberfísicos [2]. Na manufatura inteligente, a capacidade de coletar e analisar dados em tempo real é crucial para monitorar a saúde dos

equipamentos, prever falhas e garantir a qualidade do produto.

A. Internet das Coisas (IoT)

A Internet das Coisas (IoT) pode ser definida como uma extensão da Internet atual que confere a objetos do cotidiano a capacidade de comunicação e interação com a rede mundial de computadores. Essa conexão permite o controle remoto dos objetos e, mais significativamente, os transforma em provedores de serviços, ao coletarem e trocarem dados sobre o ambiente em que estão inseridos [3]. Os sistemas IoT são geralmente compostos por sensores que coletam dados do mundo físico, módulos de comunicação que transmitem esses dados, e plataformas de nuvem para armazenamento, processamento e análise. No contexto industrial, IoT possibilita o monitoramento remoto, a coleta massiva de dados e a automação de processos, sendo um habilitador fundamental para a Indústria 4.0 [4]. No presente trabalho, a câmera atua como sensor de aquisição das imagens, o HTM32 como dispositivo de comunicação, e as notificações enviadas ao usuário — via Telegram e *buzzer* — representam o "atuador" do sistema, uma vez que são o mecanismo responsável por acionar a resposta do operador diante das falhas detectadas.

B. NB-IoT e Conectividade 5G para IoT Industrial

Para a conectividade em ambientes industriais, a escolha da tecnologia de comunicação é de suma importância, priorizando a robustez, a confiabilidade e a abrangência. O *Narrowband-IoT* (NB-IoT), uma tecnologia de rede de baixa potência e longa distância (LPWAN) padronizada pelo 3GPP (*3rd Generation Partnership Project*), que faz parte do ecossistema do 5G, emerge como uma solução ideal para o monitoramento industrial [7]. O NB-IoT utiliza a infraestrutura celular existente, operando em bandas licenciadas, o que lhe confere maior segurança, cobertura e penetração em ambientes desafiadores, como grandes fábricas com muitas barreiras físicas.

A principal vantagem do NB-IoT para este trabalho reside em sua robustez e independência da internet local. Ao utilizar chips M2M (Machine-to-Machine) de operadoras como a TIM, VIVO, CLARO entre outras. O dispositivo HTM32 mantém uma conexão de comunicação estável e dedicada à rede celular 5G, garantindo a transmissão contínua de alertas críticos. Isso contrasta com tecnologias como Wi-Fi ou Ethernet, que podem ser comprometidas por falhas na infraestrutura de rede local (como roteadores ou servidores de internet), uma vulnerabilidade inaceitável para aplicações de manutenção preditiva onde a interrupção da comunicação pode resultar em perdas de alto valor. A integração do NB-IoT no espectro do 5G otimiza ainda mais essa comunicação, oferecendo maior capacidade, menor latência e maior densidade de dispositivos conectados, características essenciais para a expansão da IoT Industrial [8].

C. Inteligência Artificial e Visão Computacional para Detecção de Falhas

A Inteligência Artificial (IA) e, em particular, a Visão Computacional, têm se mostrado ferramentas poderosas para o monitoramento de processos industriais, permitindo a análise automatizada de imagens e vídeos para detecção de anomalias [9]. No contexto deste trabalho, a IA não é embarcada diretamente na câmera, mas é executada em um computador externo, onde um script Python acessa o *stream*

RTSP (*Real Time Streaming Protocol*) da câmera Intelbras IM1.

O sistema utiliza o algoritmo YOLOv5 (You Only Look Once versão 5), uma arquitetura de rede neural convolucional para detecção de objetos em tempo real. O YOLOv5 foi escolhido por sua eficiência, velocidade e precisão, características essenciais para um monitoramento contínuo em ambientes industriais, além de ser uma referência consolidada para detecção de falhas específicas, como o "nozzle missing" em impressão [10, 11].

A detecção de "nozzle missing" é crítica em impressoras de jato de tinta industrial, pois indica que um ou mais bicos estão sujos ou entupidos. Essa falha, se não for imediatamente corrigida, compromete a qualidade da impressão e leva à perda de material de alto valor. A capacidade da IA de identificar essa anomalia de forma autônoma e em tempo real é fundamental para acionar a manutenção corretiva imediata e garantir a qualidade do produto ofertado.

D. Impressão Industrial de Jato de Tinta e a Necessidade de Monitoramento

As impressoras industriais de jato de tinta de grande formato, como as máquinas K-JET de 3.2 metros, são equipamentos complexos e de alto custo, essenciais em diversas cadeias produtivas. O processo de deposição de tinta por jatos microscópicos exige precisão extrema e a funcionalidade impecável de todos os bicos injetores. Contudo, esses bicos são suscetíveis a entupimentos ou sujeiras, resultando na falha de "nozzle missing".

A ocorrência de "nozzle missing" impacta diretamente a qualidade do material impresso, levando à produção de itens defeituosos que precisam ser descartados. Em indústrias que lidam com materiais de alto valor, o desperdício resultante de uma única falha de impressão pode ser financeiramente significativo. Além do custo de material, há o custo de tinta, o tempo de máquina parado para limpeza e o retrabalho. O operador, por vezes, devido à falta de atenção ou necessidade de atuar em outras frentes, pode não perceber a falha de imediato, agravando as perdas. Dessa forma, um sistema de monitoramento constante e automatizado é imperativo para mitigar esses riscos, garantir a qualidade do produto e otimizar a operação dessas impressoras [12]. A manutenção imediata, geralmente manual, acionada pelo sistema é vital para evitar perdas maiores [13].

E. Protocolos de Comunicação

A comunicação eficiente entre os componentes do sistema é fundamental. Neste projeto, três protocolos principais foram empregados:

- *UDP (User Datagram Protocol)*: é um protocolo da camada de transporte definido pelo modelo OSI, amplamente utilizado em aplicações que demandam baixa latência e alta velocidade de transmissão, mesmo que isso implique em menor confiabilidade. Diferente do TCP (*Transmission Control Protocol*), o UDP é um protocolo "sem conexão" (*connectionless*), ou seja, não realiza o processo de *handshake* nem o controle de congestionamento ou confirmação de entrega. Cada pacote, chamado de datagrama, é enviado de forma independente, sem verificação de ordem ou retransmissão automática de pacotes perdidos.

Essa característica o torna ideal para transmissão contínua e leve de pequenas mensagens, como pacotes de alerta em sistemas de monitoramento. No presente projeto, o script Python responsável pela inferência da IA envia mensagens

via UDP contendo o estado da detecção ("nozzle missing" ou normal) para o dispositivo embarcado HTM32, através da rede local sem fio (Wi-Fi 2.4 GHz).

É importante ressaltar que, por operar exclusivamente na rede local, o protocolo UDP não depende de acesso à Internet para funcionar. Essa independência garante que o sistema continue detectando e transmitindo alertas internamente mesmo em situações de falha na conexão externa, assegurando resiliência operacional. O único requisito é que os dispositivos estejam conectados ao mesmo ponto de acesso Wi-Fi local, o que torna o processo de comunicação simples e robusto para ambientes industriais isolados da rede global. A estrutura do pacote UDP é composta por um cabeçalho simples de 8 bytes (com portas de origem, destino, comprimento e *checksum*), o que reduz significativamente o overhead e garante tempos de resposta inferiores a 1 ms em redes locais. Embora o protocolo não garanta a entrega, o risco de perda é minimizado no ambiente controlado da rede interna, e o HTM32 aplica filtros de consistência e validação dos dados recebidos — utilizando média móvel e confirmação cruzada de eventos — para evitar notificações falsas. Assim, o UDP atua como ponte de comunicação de alta velocidade e baixa dependência externa entre a camada de detecção (IA) e a camada de comunicação NB-IoT, priorizando tempo de resposta e disponibilidade local em detrimento da confiabilidade absoluta.

A camada superior da comunicação IoT do sistema é responsável pela entrega das notificações ao usuário, etapa essencial para o fechamento do ciclo de monitoramento preditivo. Para isso, foi utilizada a *-API do Telegram*: um conjunto de *endpoints RESTful* oferecidos pelo aplicativo de mensagens, que permite a integração de bots com sistemas externos. Por meio dessa API, o dispositivo HTM32 realiza requisições HTTP seguras (HTTPS) ao servidor do Telegram, utilizando o método *sendMessage* para enviar alertas automáticos ao chat do operador. Cada alerta contém informações como tipo de falha, horário de detecção e status do equipamento, podendo ser configurado para incluir imagens ou relatórios mais detalhados.

A escolha da API do Telegram deve-se a vários fatores:

- Gratuidade e disponibilidade global, sem necessidade de servidores intermediários.
- Baixa latência nas notificações, tipicamente inferior a 300 ms.
- Compatibilidade com comandos interativos, permitindo que o operador responda ou reconheça alertas diretamente no aplicativo.
- Criptografia ponta a ponta, assegurando confidencialidade na transmissão das mensagens.

Além das interfaces digitais tradicionais, o sistema desenvolvido faz uso do -protocolo RTSP (*Real Time Streaming Protocol*) para a transmissão contínua do fluxo de vídeo capturado pela câmera Intelbras IM1. O RTSP é um protocolo amplamente empregado em aplicações de vigilância e monitoramento em tempo real, permitindo que clientes de rede — como o script de visão computacional — acessem o *stream* de vídeo diretamente pela rede local. Por meio de um endereço RTSP autenticado, a câmera transmite quadros em alta definição para o módulo de processamento, garantindo baixa latência e sincronização contínua das imagens. Essa arquitetura torna possível a análise em tempo real dos frames pela Inteligência Artificial, viabilizando a detecção imediata de falhas de impressão como o nozzle missing sem a necessidade de armazenamento local de vídeo.

F. Sistema de Monitoramento Preditivo via IA e NB-IoT

O sistema completo de monitoramento proposto neste trabalho foi estruturado em módulos interconectados, de forma a garantir a integração entre captura de imagem, processamento inteligente, comunicação NB-IoT e notificação ao operador. A Figura 1 ilustra a arquitetura modular do sistema, destacando o fluxo de dados desde a aquisição da imagem até o envio do alerta ao usuário e o acionamento do alarme sonoro.

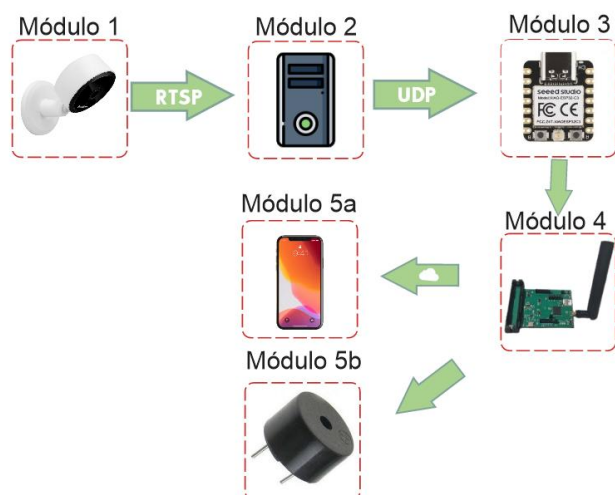


Figura 01. Arquitetura modular do sistema de monitoramento preditivo via IA e NB-IoT.

A implementação foi organizada em **cinco módulos principais (1 a 5)**, descritos a seguir.

-Módulo 1 – Aquisição de Imagem (Câmera Intelbras IM1): O primeiro módulo é responsável pela captura contínua do processo de impressão por meio da câmera Intelbras IM1, que fornece vídeo em tempo real utilizando o protocolo RTSP. Essa câmera foi escolhida por sua compatibilidade com sistemas de monitoramento industrial, robustez de conexão e facilidade de integração com aplicações de visão computacional. A câmera é posicionada de modo a enquadrar a área de impressão e transmitir as imagens através da rede local (LAN) para o servidor de processamento. O uso do RTSP assegura estabilidade na transmissão e permite que o módulo seguinte acesse diretamente o fluxo de vídeo para análise instantânea.

-Módulo 2 – Processamento de Imagem e Detecção de Falhas: No segundo módulo ocorre o processamento de imagem e a inferência da Inteligência Artificial. Um servidor executa um script Python que consome o fluxo RTSP proveniente da câmera e aplica o modelo de detecção YOLOv5, treinado especificamente para identificar falhas de impressão (*nozzle missing*). Quando a IA reconhece uma anomalia, o sistema gera um pacote de dados contendo informações do evento — como o tipo de falha, o instante da detecção e o status do processo — e o envia via protocolo UDP para o próximo módulo. O uso de UDP foi definido por sua baixa latência e simplicidade de implementação, características que o tornam ideal para a transmissão de mensagens curtas e imediatas, onde a prioridade é a velocidade de comunicação.

-Módulo 3 – Interface UDP-UART (ESP32-C3): O terceiro módulo é composto por um microcontrolador ESP32-C3, responsável por receber os pacotes UDP enviados pelo script de Inteligência Artificial e retransmiti-los via comunicação serial UART para o módulo NB-IoT HMT32. O ESP32-C3

atua apenas como uma ponte de dados entre a rede local Wi-Fi e o sistema celular, sem realizar processamento adicional. Essa função garante a transferência rápida e estável das informações de detecção, assegurando que os alertas gerados pela IA cheguem corretamente ao HMT32 para posterior envio ao operador.

-Módulo 4 – Comunicação NB-IoT e Envio de Alertas: O quarto módulo é composto pelo dispositivo NB-IoT HMT32, responsável por capturar os pacotes UDP e transmiti-los pela rede celular 5G através de sua interface NB-IoT. Esse módulo atua como um gateway inteligente, traduzindo a comunicação local em uma mensagem de telemetria móvel robusta. A transmissão é realizada utilizando um chip M2M com conectividade dedicada, garantindo que os alertas cheguem ao servidor remoto e ao aplicativo de mensagens Telegram mesmo em situações de falha da internet local. Assim, a infraestrutura NB-IoT assegura confiabilidade e resiliência de comunicação, pontos críticos para aplicações industriais que demandam disponibilidade contínua.

- Módulo 5 – Notificação e Atuação: O quinto módulo é subdividido em duas partes complementares.

-Módulo 5a – Notificação Mobile: Responsável por encaminhar a mensagem de alerta ao operador por meio do aplicativo Telegram, contendo informações sobre o tipo de falha e o equipamento afetado. Essa comunicação instantânea permite uma resposta ágil e a tomada de decisão imediata, reduzindo significativamente o tempo de parada da impressora.

-Módulo 5b – Atuação Local: Consiste em um *buzzer* conectado ao HMT32 que é acionado assim que o evento de falha é identificado e confirmado. O alerta sonoro fornece uma indicação local imediata, mesmo em situações em que o operador não esteja com o smartphone em mãos, garantindo redundância no processo de aviso.

- Integração entre os Módulos: O fluxo de dados entre os módulos ocorre de forma sequencial e sincronizada: a câmera (Módulo 1) captura as imagens e as envia via RTSP para o processador (Módulo 2), onde a IA analisa e envia as detecções via UDP para o dispositivo NB-IoT (Módulo 3). Este, por sua vez, encaminha as notificações simultaneamente ao smartphone (Módulo 4a) e ao *buzzer* (Módulo 4b). Essa arquitetura modular oferece escalabilidade, permitindo que novos sensores, câmeras ou atuadores sejam adicionados futuramente sem a necessidade de reestruturação do sistema. Além disso, o uso de protocolos leves (RTSP e UDP) combinado à conectividade 5G NB-IoT proporciona um sistema eficiente, confiável e com baixa latência, adequado a aplicações industriais em tempo real.

A adoção desse modelo modular facilita o diagnóstico e manutenção, além de favorecer o reaproveitamento de cada subsistema em outras aplicações de monitoramento industrial. O conjunto formado pela IA embarcada, transmissão RTSP, comunicação UDP e NB-IoT cria uma solução integrada que alia velocidade, precisão e robustez — três pilares fundamentais para sistemas de manutenção preditiva baseados em IoT.

G. Plataformas Embarcadas: O Dispositivo HMT32

Sistemas embarcados são plataformas de desenvolvimento compostas por um componente de processamento (microcontrolador ou microprocessador) acoplado a um circuito impresso, projetados para executar

funções específicas, controladas por um software (firmware) otimizado [14]. Para aplicações de IoT industrial, a escolha de uma plataforma embarcada robusta e com capacidade de comunicação adequada é crucial.

Neste trabalho, o dispositivo HTM32 (MCP HTNBAT32L-XXX) foi selecionado como a plataforma embarcada principal. Sua escolha se deu pela robustez inerente à conectividade 5G (NB-IoT) e por possuir um processador ARM Cortex M3 (operando a até 204MHz), adequado para o processamento de dados UDP e a gestão da comunicação celular, com 4MB de NOR flash e 272KB de SRAM. O HTM32 integra o modem Qualcomm QCX-212 LTE IoT, suportando o 3GPP Release 14 Cat. NB2 IoT, o que garante compatibilidade com as redes celulares M2M [15].

Suas características de hardware incluem interfaces digitais como I2C, UART, SPI e GPIOs (com PWM e Timer), além de um ADC de 12 bits, que oferecem flexibilidade para futuras expansões. A faixa de alimentação (2.5 - 4.3V) e a temperatura de operação (-20 a +70 °C) o tornam adequado para ambientes industriais. A presença de recursos de segurança como criptografia de hardware (AES e SHA), flash encryption, secure boot e suporte a SSL/TLS/DTLS reforça sua adequação para aplicações críticas. Além disso, o suporte a FreeRTOS, controle via comandos AT (3GPP TS27.005, 27.007), e a capacidade de lidar com IPv4, IPv6, UDP, TCP, MQTT e HTTPS, conferem grande versatilidade ao dispositivo para comunicação [15]. A disponibilidade do chip M2M da TIM integrado à placa simplifica a implementação da conectividade NB-IoT 5G.

H. Trabalhos Relacionados

A literatura apresenta diversos estudos que exploraram a aplicação de visão computacional e IoT no monitoramento industrial. A seguir, destacam-se alguns trabalhos correlatos ao presente projeto:

Trabalho 1 [10]: Choi, E. et al. apresentaram um estudo sobre detecção de gotículas de tinta em jato, utilizando *Deep Learning* para caracterização do jato e sincronização *multijet*. Embora o foco seja na otimização do processo de jato de tinta em si, o uso de algoritmos baseados em YOLOv5 para análise visual de falhas de precisão corrobora a escolha do YOLOv5 para o presente trabalho, evidenciando sua adequação para a inspeção de componentes críticos em sistemas de impressão.

Trabalho 2 [11]: Liu, J. et al. propuseram um método eficiente de detecção de defeitos de impressão baseado em YOLOv5-DCN-LSK. Este trabalho demonstra a versatilidade e a eficácia das variantes do YOLOv5 na identificação de uma ampla gama de defeitos em processos de impressão. A análise comparativa de diversas versões do YOLO (até a v9) que este estudo oferece reforça a decisão de utilizar o YOLOv5 pela sua performance equilibrada em velocidade e precisão.

Trabalho 3 [13]: Bischoff et al. desenvolveram um método para monitorar a degradação da qualidade de impressão em sistemas industriais de jato de tinta por meio da análise das imagens impressas. Utilizando uma impressora DOD em um ensaio de longo prazo, o estudo identifica tendências de deterioração, como variações no tamanho e posição dos pontos, associadas a falhas de bicos (*nozzle clogging*). Essa abordagem reforça a importância de sistemas de detecção rápida dessas anomalias para acionar manutenção preditiva e evitar perdas na produção, alinhando-se diretamente com o objetivo deste trabalho.

O diferencial do trabalho desenvolvido neste artigo reside na integração de uma câmera de IA comercial (Intelbras IM1)

com um sistema de comunicação NB-IoT 5G robusto para alertas em tempo real via Telegram, focado especificamente na detecção de "nozzle missing" em impressoras industriais de grande formato K-JET. Enquanto os trabalhos correlatos focam na detecção de defeitos ou otimização do jato de tinta, ou em sistemas de monitoramento mais genéricos, o presente projeto oferece uma solução completa que engloba a aquisição inteligente da falha, a comunicação ultra confiável em ambientes industriais críticos (graças ao NB-IoT 5G), e um sistema de alerta imediato e acessível ao operador, mitigando o risco de falhas por desatenção ou interrupções na rede local.

III. MATERIAIS E MÉTODOS

Visando a resolução de problemas relacionados à detecção de falhas de impressão em tempo real e à comunicação confiável em ambientes industriais, foi desenvolvido um estudo para a criação de um sistema de monitoramento preditivo via IA e NB-IoT, considerando uma escala de aplicação prática em impressoras industriais de grande formato.

O projeto tem como propósito o desenvolvimento de uma solução integrada que una a visão computacional baseada em Inteligência Artificial e a comunicação NB-IoT sobre rede 5G, de forma a permitir o monitoramento contínuo e o envio de alertas automáticos ao operador sempre que uma anomalia de impressão (como o "nozzle missing") for detectada.

A. Definição do Hardware de Monitoramento

A etapa inicial do desenvolvimento do sistema consistiu na definição do hardware de monitoramento, responsável pela captura das imagens do processo de impressão e pela geração dos dados visuais utilizados pela Inteligência Artificial. Essa fase teve como objetivo principal estabelecer um fluxo estruturado de decisões para a seleção, configuração e validação do dispositivo mais adequado ao contexto industrial do projeto.

O processo foi conduzido de forma sistemática, o fluxo completo dessas etapas está representado na Figura 02, que descreve graficamente o processo de definição e montagem do hardware adotado no sistema de monitoramento preditivo via IA e NB-IoT.

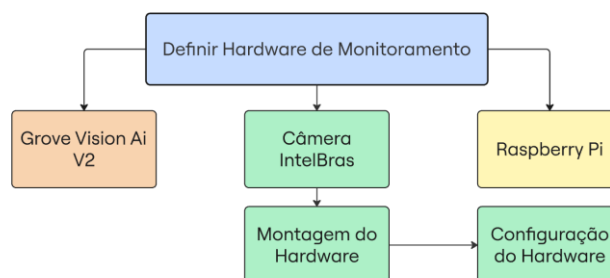


Figura 02. Fluxograma do processo de definição e montagem do hardware de monitoramento.

Esse fluxo permitiu estruturar de forma clara o processo de desenvolvimento físico do sistema, garantindo que cada decisão de hardware fosse acompanhada de critérios técnicos bem definidos. A partir dessa análise, quatro alternativas foram consideradas para o módulo de captura de imagem: Grove Vision AI V2, Raspberry Pi 4 e Câmera Intelbras IM1.

A Tabela 01 apresenta o comparativo técnico entre essas opções, com base em parâmetros essenciais de desempenho, custo e integração.

Tabela 01. Comparativo entre plataformas de monitoramento.

Nome	Câmera Intelbras	Raspberry Pi 3	Grove Vision	Esp32Cam
Modelo	IM1	Modelo B	AI V2	ESP-32
Custo (R\$)	200	500	430	30
Lâtenzia	Baixa	Baixa	Muito Baixa	Alta
Integração	Simple	Média	Complexa	Difícil
Inferência(FPS)	30	60	33	10
Tempo de Envio	Imediato	Imediato	21 dias	21 dias
Interf. Desenvol.	Python VScode	Linux OS Python	Edge Impulse	Arduino IDE+EI

Fonte: elaborada pelo autor com base em IntelBras, Raspberry, HacksterIO,Espressif.

Após a análise comparativa e os testes iniciais de funcionamento, a Câmera Intelbras IM1 foi selecionada como o hardware principal do sistema. Essa decisão baseou-se na sua confiabilidade operacional, compatibilidade com o protocolo RTSP, disponibilidade nacional imediata e facilidade de integração com o código Python responsável pela inferência da IA.

O uso do protocolo RTSP garante a baixa latência e comunicação estável com o módulo de processamento de IA. Esses fatores tornam a IM1 a opção ideal para monitoramento contínuo de impressoras industriais, viabilizando a detecção automática e em tempo real de falhas de impressão, como o “*nozzle missing*”.

- *Montagem e Configuração Prática:* Com base no fluxograma estabelecido, procedeu-se à montagem física do sistema. A Câmera Intelbras IM1 foi fixada mecanicamente na estrutura da impressora industrial por meio de parafusos e suporte metálico ajustável, de modo a manter o campo de visão estável sobre a área de impressão. Essa fixação garante que o ângulo de captura permaneça constante durante a operação contínua da máquina, evitando ruídos visuais no processo de detecção da IA.



Figura 03. Câmera Intelbras IM1 montada na Impressora K-JET 3204.

Após a instalação física, foi realizada a configuração lógica do dispositivo, composta pelas seguintes etapas:

1. Conexão da câmera à rede Wi-Fi 2.4 GHz, mantendo compatibilidade com o roteador local e o computador responsável pelo processamento de IA.
2. Atribuição de endereço IP fixo à câmera, garantindo sua identificação estável dentro da rede local;
3. Definição da porta RTSP (*Real Time Streaming Protocol*) para acesso direto ao fluxo de vídeo;

4. Criação de credenciais de autenticação (usuário e senha) para o dispositivo, assegurando o controle de acesso;

Concluídas essas configurações, foi executado um teste de conectividade e streaming através de um script em Python, utilizando a biblioteca OpenCV[20]. O teste consistiu na abertura do canal RTSP e na exibição dos quadros de vídeo em tempo real, confirmando a estabilidade da comunicação e a integridade do fluxo de dados.

B. Definição da Inteligência Artificial para Monitoramento

Para a implementação do sistema de monitoramento preditivo da impressora industrial de jato de tinta, utilizou-se uma abordagem modular, integrando captura de imagem, processamento por IA, comunicação NB-IoT e notificação ao operador. O fluxo de funcionamento do sistema está ilustrado na Figura 04, que apresenta tanto a opção de criar um modelo próprio quanto a utilização de um modelo pronto (YOLOv5) para detecção de falhas.

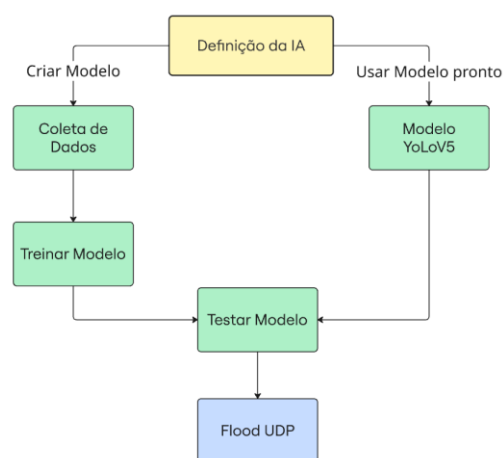


Figura 04. Fluxograma da IA preditivo de falhas de impressão.

No método implementado, a aquisição das amostras ocorre diretamente na saída da impressão, utilizando uma câmera Intelbras posicionada de forma fixa para capturar continuamente a imagem da linha impressa. Cada imagem capturada é encaminhada a um script Python responsável por executar a inferência utilizando o modelo YOLO previamente treinado. Como se trata de um modelo leve, otimizado apenas para a fase de inferência, sua execução local torna-se viável e não afeta o desempenho da produção, garantindo processamento em tempo real.

Após o carregamento da imagem, o script realiza o pré-processamento e envia o quadro ao modelo, que detecta regiões correspondentes ao padrão de impressão e identifica a presença de falhas associadas ao fenômeno *nozzle missing*. Para permitir uma análise mais objetiva do estado de cada bico, foi definido um índice percentual de entupimento, variando de **0% (sem obstrução)** a **100% (entupimento total)**. Esse índice é discretizado em intervalos de **10%**, de forma a simplificar a interpretação dos resultados e aproximar o comportamento real observado no processo industrial.

Com base nesses valores, o operador pode configurar um **limiar de acionamento**, permitindo determinar em qual nível de entupimento o sistema deve gerar uma notificação. Assim, o próprio método possibilita adaptar o sistema às exigências específicas de diferentes linhas de impressão ou condições de operação, mantendo a coerência com a necessidade de intervenções rápidas no ambiente industrial.

Como a comunicação com o módulo ESP32-C3 ocorre via UDP, em rede local Wi-Fi na banda de 2,4 GHz, toda a operação de inferência precisa ser realizada localmente. Esse requisito elimina qualquer dependência de serviços em nuvem, reduz a latência e assegura que o sistema permaneça funcional mesmo em ambientes industriais sem acesso à Internet. Dessa forma, o processamento local, aliado ao protocolo UDP, garante que a detecção seja feita de maneira contínua, rápida e integrada ao fluxo produtivo.

C. Integração e Programação dos Módulos ESP32-C3 e HTM32

Nesta etapa, foi realizada a integração entre os módulos ESP32-C3 e HTM32 (HTNB32L), conforme o fluxo ilustrado na Figura 05. O objetivo foi estabelecer uma comunicação confiável entre a rede local (Wi-Fi 2.4 GHz) e o canal NB-IoT 5G, garantindo que as mensagens geradas pela Inteligência Artificial fossem recebidas e encaminhadas ao operador em tempo real.

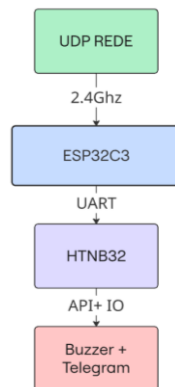


Figura 05. Fluxograma da integração entre os módulos ESP32-C3 e HTM32 (HTNB32L).

1. Programação do ESP32-C3

O microcontrolador ESP32-C3 foi programado utilizando o ambiente Visual Studio Code (VSCode) com a extensão PlatformIO[18], que fornece um ecossistema completo para desenvolvimento embarcado. Essa plataforma foi escolhida por permitir a integração direta com bibliotecas da Espressif[19] e oferecer suporte à compilação, upload e monitoramento serial de forma unificada.

O firmware foi desenvolvido em linguagem C++, utilizando as bibliotecas nativas do Arduino Core for ESP32, configuradas para operação em rede Wi-Fi 2.4 GHz. A principal função do código é receber, via UDP, os pacotes transmitidos pelo script Python de inferência da IA e retransmiti-los pela interface UART, através do pino GPIO10 (TX), para o módulo HTM32.

A comunicação UDP foi implementada por meio das classes WiFiUDP e WiFi.h, que permitem operação em modo cliente dentro da mesma sub-rede local da câmera Intelbras IM1. Como o ESP32-C3 não realiza nenhum processamento adicional sobre os dados, ele atua como uma ponte de comunicação rápida e eficiente, convertendo os pacotes da rede local em dados seriais assíncronos. Essa abordagem reduz a carga computacional do sistema NB-IoT e assegura baixa latência na transmissão.

O funcionamento geral do algoritmo proposto pode ser visualizado no Pseudocódigo 01, que descreve a lógica principal utilizada para identificar a condição de *nozzle missing* e acionar os módulos de comunicação.

Pseudocódigo 01 – UDP to UART

Passo01. Conectar no Wi-Fi e Inicializa pino D10 como Saída

Conecta no SSID e password

Passo02. Começa a fazer leitura UDP Broadcast no Port 5005

Passo03. Apenas pega mensagens com início "#TCC:"

Passo04. Envia mensagem captada ao Pino D10 no baudrate 115200

(Passo01). Conectar ao Wi-Fi e inicializar o pino D10 como saída

O microcontrolador ESP32-C3 é configurado para se conectar à rede Wi-Fi local utilizando o SSID e a senha definidos no código-fonte[19]. Em seguida, o pino D10 é inicializado como saída UART (TX), estabelecendo o canal de transmissão serial para o módulo HTM32.

(Passo02). Iniciar escuta de mensagens UDP Broadcast na porta 5005.

Após a conexão Wi-Fi, o módulo configura um socket UDP e começa a escutar continuamente na porta 5005. Essa porta é usada para receber as mensagens enviadas pelo script Python responsável pela inferência da Inteligência Artificial.

(Passo03). Filtrar mensagens recebidas com prefixo "#TCC:." cada pacote UDP recebido é analisado. Somente as mensagens que iniciam com o identificador "#TCC:." são consideradas válidas, garantindo que apenas os dados relevantes da aplicação sejam processados e retransmitidos.

(Passo04). Enviar a mensagem captada via UART (pino D10, baudrate 115200) As mensagens filtradas são transmitidas diretamente ao módulo HTM32 pela interface UART configurada no pino D10, utilizando taxa de transmissão de 115200 bps. Essa etapa garante o envio rápido e assíncrono das informações detectadas pela IA ao canal NB-IoT, permitindo a notificação imediata ao operador.

2. Programação do HTM32 (HTNB32L)

O módulo HTM32 foi configurado e programado com base no SDK oficial HTNB32L-XXX, descrito no manual *Getting Started* fornecido pela HT Micron [14]. O desenvolvimento foi conduzido inteiramente no VSCode, com o suporte de extensões específicas recomendadas pela fabricante — como C/C++, Cortex-Debug, Makefile Tools, RTOS Views e Serial Monitor — além da instalação do compilador GNU ARM Embedded Toolchain (gcc-arm-none-eabi).

O SDK disponibiliza um ambiente completo baseado em FreeRTOS, permitindo que o firmware seja embarcado diretamente no microcontrolador principal (modo OpenCPU), dispensando um microcontrolador externo. O projeto foi estruturado com base no exemplo de comunicação MQTT_Example, adaptado para operar com UART e HTTP API.

A aplicação desenvolvida no HTM32 executa continuamente a leitura da UART, utilizando a biblioteca PySerial durante as fases de teste e depuração, e converte as mensagens recebidas em comandos de alto nível. Assim que o módulo identifica uma mensagem válida, ele verifica o Limiar de ativação e pode executar ou não duas tarefas principais:

1. Envio de alerta via API do Telegram, utilizando o protocolo HTTPS para transmitir mensagens automáticas ao operador por meio do bot configurado;

2. Atuação local via GPIO, acionando um *buzzer* conectado a uma das saídas digitais para sinalização sonora imediata.

O processo de compilação e gravação do firmware foi realizado diretamente pelo VSCode, utilizando as tarefas pré-configuradas do SDK (`make -j4 gccall`).

TARGET=qcx212_0h00), e o carregamento foi efetuado pela interface UART, aproveitando o *bootloader* interno do módulo. O ambiente de depuração foi configurado com o SEGGER J-Link, conforme as instruções do fabricante, permitindo testes em tempo real e validação da comunicação serial entre os módulos.

Essa integração consolidou uma arquitetura híbrida e robusta, em que o ESP32-C3 se encarrega da comunicação local em alta velocidade via Wi-Fi/UDP, enquanto o HTM32 é responsável pela camada de conectividade celular NB-IoT e notificação em nuvem, garantindo operação contínua mesmo em situações de falha na infraestrutura de internet local.

Para complementar a descrição dos procedimentos adotados, o Pseudocódigo 02 apresenta de forma estruturada o funcionamento do módulo de inteligência artificial atuando como elemento de decisão no sistema. Nele são detalhadas as etapas de captura da imagem, processamento pelo modelo de inferência, cálculo do índice percentual de entupimento e comparação com o limiar definido pelo operador. A partir dessa lógica, o sistema passa a atuar automaticamente sempre que o valor identificado ultrapassar o limite estabelecido, enviando um comando de alerta ao módulo ESP32-C3 via UDP.

Pseudocódigo 02 – IA para Atuador

Passo01. Conectar na APN chip e configurar Pinos
 Passo02. Ler entrada UART
 Passo03. Processar de acordo com limiar nozzle missing
 Passo04. Ativar ou não atuadores de acordo com limiar de ativação
 Passo04. Looping

(Passo01). Inicializar o módulo e configurar UART
 O microcontrolador HTM32 é inicializado em modo *OpenCPU* com o ambiente FreeRTOS. A UART é configurada para comunicação serial com o ESP32-C3, utilizando baud rate de 115200 bps, para receber as mensagens transmitidas pelo pino TX (D10) do módulo Wi-Fi, Conectar APN do chip e testar sinal

(Passo02). Ler continuamente os dados recebidos na UART
 O firmware entra em um laço de leitura contínua da UART. Cada pacote recebido é armazenado temporariamente em buffer de memória e analisado para identificar comandos ou mensagens provenientes da ponte UDP [17].

(Passo03). Filtrar mensagens válidas com prefixo “#TCC:”
 Somente as mensagens iniciadas com o identificador “#TCC:” são consideradas válidas. Esse filtro evita a interpretação incorreta de ruídos ou dados não pertencentes ao sistema de monitoramento.

(Passo04). Processar e transmitir alerta via API do Telegram [16]
 Ao reconhecer uma mensagem válida, o módulo HTM32 executa duas ações:

- (a) envia uma notificação automática ao operador por meio de requisição HTTPS para a API do Telegram, utilizando o *bot* configurado;
- (b) aciona um *buzzer* local via GPIO, sinalizando o evento de forma sonora e imediata.

(Passo05). Repetir continuamente o processo
 O sistema retorna ao estado de espera de novas mensagens na UART, garantindo a operação contínua e autônoma mesmo em caso de falhas na conexão local de internet. A implementação completa do sistema está disponível no repositório GitHub do autor [22].

IV.RESULTADOS E DISCUSSÕES

Após os testes iniciais utilizando o modelo pré-treinado YOLOv5, observou-se que o desempenho na detecção de falhas específicas de impressão, como o *nozzle missing*, não atingiu o nível de precisão esperado para aplicação em ambiente industrial. Diante disso, optou-se pelo desenvolvimento de um modelo próprio de Inteligência Artificial, adaptado às particularidades visuais do processo de impressão analisado.

A. Coleta das Imagens e Rotulagem (Labeling)

Para a etapa de treinamento da Inteligência Artificial, foi desenvolvido um processo sistemático de captura, rotulagem e organização das imagens, garantindo a construção de um dataset compatível com o modelo YOLOv5. As imagens foram obtidas diretamente da câmera Intelbras IM1 durante a operação real da impressora industrial, resultando em um conjunto de 465 imagens contendo diferentes graus de falhas de jato (*nozzle missing*), variando entre 0% e 100%.

A rotulagem manual foi realizada utilizando a plataforma *MakeSense.ai*, onde cada falha foi marcada visualmente. Inicialmente, os rótulos foram exportados no formato COCO JSON, permitindo a verificação completa das *bounding boxes*, suas dimensões e posições absolutas (*x, y, width, height*). Após validação, todos os rótulos foram convertidos automaticamente para o formato YOLO TXT, onde cada imagem passa a ter um arquivo correspondente contendo suas classes e coordenadas normalizadas (padrão YOLO: *class x_center y_center width height*). Esse formato é obrigatório para o treinamento do YOLOv5.

A Figura 06 apresenta um exemplo da interface de rotulagem no *MakeSense.ai*.

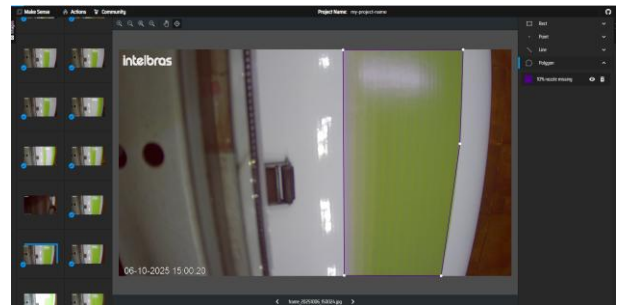


Figura 06. *Labeling* manual na plataforma *MakeSense.ai*, Exemplo de Nozzle com entupimento de 10%.

Foram definidas **11 classes**, representando cada nível de severidade de *nozzle missing*, indo de “0% *nozzle missing*” até “100% *nozzle missing*” de 10% em 10%. Essa abordagem permitiu que o modelo aprendesse não apenas a presença da falha, mas também a intensidade do defeito, tornando o sistema mais útil para diagnósticos e tomadas de decisão no controle de qualidade em ambiente industrial.

Essa preparação criteriosa do dataset foi crucial para garantir que o modelo pudesse generalizar corretamente e compreender a progressão real das falhas durante o processo de impressão.

B. Resultados da Detecção por Inteligência Artificial

O modelo treinado com 150 épocas foi então testado em condições reais. Em materiais semelhantes aos utilizados no treinamento, a precisão média obtida foi de 67% de 10 testes. Contudo, ao testar materiais de cores significativamente

diferentes — como vermelho, laranja e amarelo — a precisão caiu para cerca de 35%.

Essa diferença se deve à limitação do dataset, composto por poucas cores e pouca variação de iluminação. Observou-se que cores mais claras apresentaram maior dificuldade para identificação das falhas, pois a câmera IM1 não consegue capturar com clareza os riscos sutis do *nozzle missing* nessas tonalidades. Já nas cores escuras, a diferença de contraste facilita a detecção, resultando em melhor desempenho.

A Figura 07 apresenta o exemplo de detecção real com a *bounding box* aplicada pelo modelo YOLOv5.

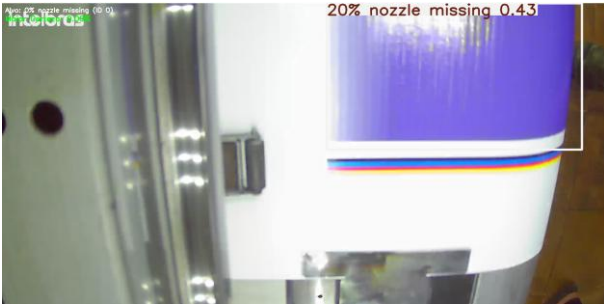


Figura 07. Detecção em Tempo Real visualizada pela Câmera 20% de nozzle missing.

Além disso, o tempo total necessário para processamento da inferência foi de aproximadamente 150 ms, atendendo aos requisitos de baixa latência exigidos no ambiente industrial.

C. Resultados da Comunicação via UDP

A comunicação entre o script Python (IA) e o módulo HTM32 ocorreu através do protocolo UDP, escolhido pela sua baixa latência. O tempo médio entre o envio do pacote pelo computador e o recebimento no HTM32 permaneceu em torno de 500 ms.

Embora ocorreram perdas eventuais de pacotes — característica típica do UDP — nenhuma falha operacional foi registrada. Isso foi possível graças ao mecanismo de reenvio contínuo, no qual o script transmitia o estado da falha repetidamente enquanto a IA confirmasse o *nozzle missing*. Para evitar excesso de alertas, foi implementado um sistema de *cooldown*, limitado a 5 notificações em um intervalo de 10 segundos. Essa abordagem equilibrou rapidez, eficiência e confiabilidade, permitindo comunicação eficaz mesmo em ambiente fabril sujeito a interferências.

D. Desempenho do Módulo NB-IoT e Envio ao Telegram

O envio da notificação final ao operador foi realizado pelo módulo celular NB-IoT da HTMicron, compatível exclusivamente com chips CAT-NB2. Essa limitação exigiu a utilização de SIM cards fornecidos pela Datatem, compatíveis com a tecnologia NB-IoT pura.

Os testes demonstraram que o tempo total entre o envio do alerta pelo HTM32 e a chegada da mensagem no Telegram variou entre 0,8 s e 1,2 s, com variação típica de redes celulares (~20%). Os valores registrados foram:

- 1,224 s
- 0,886 s
- 1,566 s
- 1,032 s
- 1,102 s

Importante destacar que o sistema funcionou perfeitamente mesmo com o Wi-Fi da fábrica desligado, comprovando a resiliência da solução e a independência da infraestrutura

local. A Figura 08 mostra a tela do smartphone exibindo uma das notificações enviadas via NB-IoT.

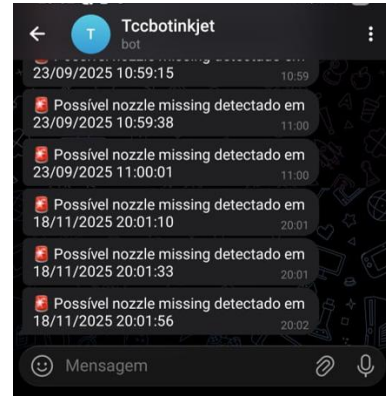


Figura 08. Notificações via bot Telegram no Smartphone.

E. Hardware e Integração Física do Protótipo

A validação física do protótipo foi conduzida através da montagem e integração dos módulos responsáveis pela recepção dos pacotes UDP, processamento local e transmissão via NB-IoT. O conjunto de hardware empregado consistiu no módulo HTMicron (HTM32) com modem NB-IoT integrado (compatível com SIM CAT-NB2) e uma MCU ESP32-C3 utilizada para receber os pacotes UDP do computador (via Wi-Fi local) e encaminhar os comandos ao modem.

A linha UART entre os dispositivos foi configurada para comunicação serial entre a ESP32-C3 e o HTM32, permitindo que os estados detectados pela IA (enviados por UDP ao ESP32-C3) fossem rapidamente repassados ao modem NB-IoT para posterior envio ao servidor/Telegram.

O pino D10 da ESP32-C3 foi conectado ao GPIO6 do HTMicron — essa conexão é usada para o envio do pacote de evento do microcontrolador para o modem HTMicron.

O GPIO5 do HTMicron foi configurado para acionar um *buzzer* local (alerta sonoro). Quando o HTM32 confirma o evento crítico, ele ativa o GPIO5, que por sua vez dispara o *buzzer*, sinalizando imediatamente o operador na planta.

A Figura 09 mostra a foto do protótipo com a ESP32-C3 montada conectada a placa do HTMicron, a conexão UART, a antena NB-IoT e Conexão ao Buzzer.

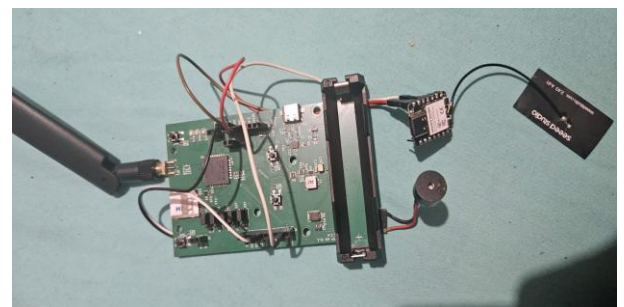


Figura 09. Montagem do protótipo: ESP32-C3 montada conectada ao kit HTMicron, conexões UART e antena NB-IoT.

Testes de integração elétrica e de comunicação

Durante os testes práticos verificou-se: a estabilidade da interface UART entre ESP32-C3 e HTM32 sob carga de envio repetido (reenvio UDP contínuo); o funcionamento do acionamento do *buzzer* via GPIO5 em sincronia com a notificação enviada ao Telegram; e operação contínua do modem NB-IoT com os SIMs CAT-NB2 (Datatem), inclusive com o Wi-Fi local desligado.

Esses testes confirmaram que a montagem física e as conexões adotadas suportam o funcionamento contínuo exigido em ambiente industrial, além de permitir manutenções e substituições modulares dos componentes sem necessidade de retrabalho significativo.

A Figura 10 ilustra o diagrama de ligação entre a ESP32-C3 e o módulo HTM32 utilizado no sistema. O pino D10 da ESP32-C3 foi conectado ao GPIO6 do HTM32 para o recebimento das mensagens enviadas via UDP, enquanto o GPIO4 foi configurado para o acionamento do buzzer. Esse conjunto de conexões compõe a interface embarcada responsável pela interpretação dos alertas e atuação do sistema.

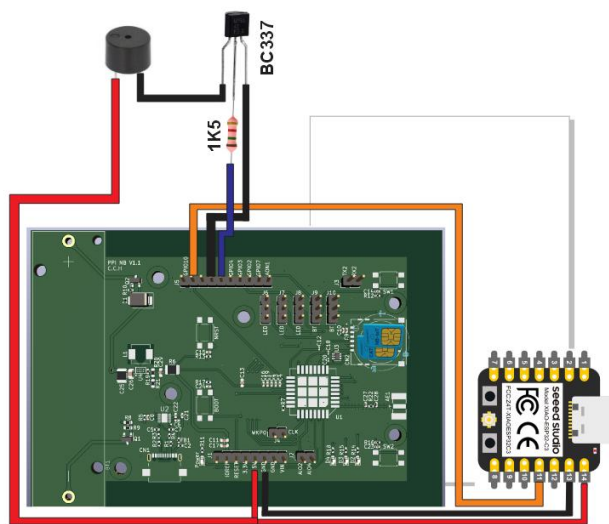


Figura 10. Diagrama esquemático das principais conexões do protótipo.

V.CONCLUSÃO

O presente trabalho apresentou o desenvolvimento de um sistema de monitoramento preditivo voltado para impressoras industriais de grande formato, capaz de detectar automaticamente falhas do tipo *nozzle missing* e transmitir alertas críticos em tempo real por meio da tecnologia NB-IoT. A solução integrou visão computacional baseada em Inteligência Artificial, comunicação em rede local via UDP e transferência de dados resiliente pela infraestrutura celular 5G, demonstrando ser uma alternativa robusta ao monitoramento tradicional dessas máquinas.

A abordagem adotada permitiu comprovar a viabilidade técnica de se aplicar IA para identificação de falhas em impressões reais utilizando uma câmera Intelbras IM1. O modelo YOLOv5 treinado apresentou desempenho satisfatório em cenários reais, obtendo taxa média de acerto próxima de 67% para cores semelhantes às presentes no dataset. Observou-se, porém, que a generalização para cores mais claras ou significativamente diferentes das usadas no treinamento resultou em queda de desempenho, atingindo aproximadamente 35%. Esses resultados evidenciam tanto o potencial quanto as limitações do sistema, reforçando a necessidade futura de ampliar o conjunto de imagens e diversificar o dataset para aumentar a robustez do modelo.

O uso do protocolo UDP demonstrou-se adequado para aplicações de baixa latência, permitindo que o dispositivo NB-IoT HTM32 recebesse os pacotes em cerca de 200 ms, mesmo com pequenas perdas inerentes ao protocolo. A transmissão dos alertas via NB-IoT em rede 5G se mostrou altamente confiável, garantindo notificações mesmo em

cenários de queda da internet local — uma limitação crítica em ambientes industriais. O tempo total médio entre a detecção do defeito pela IA e o recebimento do alerta no Telegram foi de aproximadamente 1 segundo, variando dentro de uma faixa aceitável para aplicações de manutenção preventiva. Essa latência reduzida possibilita intervenções rápidas, contribuindo para evitar desperdícios de materiais e minimizar o tempo de máquina parada.

A arquitetura proposta demonstrou, portanto, que a combinação de visão computacional, comunicação em rede local e conectividade NB-IoT forma uma solução eficiente, resiliente e de baixo custo para o monitoramento preditivo de impressoras industriais. Além disso, a implementação prática mostrou que dispositivos como o HTM32, aliados a microcontroladores embarcados como a ESP32-C3, são capazes de oferecer uma infraestrutura simples, modular e confiável para aplicações industriais reais.

Por fim, destaca-se que a solução desenvolvida não apenas cumpre seu objetivo principal, mas também abre caminho para aprimoramentos futuros. Entre eles, recomenda-se: expansão do dataset com imagens de diversas tonalidades e condições de iluminação; refinamento do modelo de IA para melhorar a acurácia em ambientes variáveis; implementação de mecanismos de ajuste dinâmico do limiar de detecção, visto que atualmente essa configuração só pode ser modificada diretamente no código do HTM32; e, adicionalmente, a possibilidade de integração com atuadores industriais, permitindo que o próprio sistema seja capaz de interromper automaticamente a impressão quando forem detectados níveis críticos de falha. Essas evoluções podem elevar ainda mais a eficácia da solução no controle de qualidade, consolidando seu uso em processos industriais de alta criticidade e fortalecendo práticas modernas de manutenção preditiva alinhadas aos princípios da Indústria 4.0 para o controle de qualidade.

VI.REFERENCIAS

- [1] FOLGADO, F. J. et al. Review of Industry 4.0 from the perspective of automation and supervision systems: definitions, architectures and recent trends. *Electronics*, v. 13, n. 4, art. 782, 2024. DOI: 10.3390/electronics13040782.
- [2] MANSOUR, M. et al. Internet of Things: a comprehensive overview on protocols, architectures, technologies, simulation tools, and future directions. *Energies*, v. 16, n. 8, 3465, 2023. DOI: 10.3390/en16083465.
- [3] WEI, J.; et al. Convergence of IoT and PLC in Industrial Automation: A Systematic Review of Emerging Trends, Technical Challenges and Prospects. 2025. Disponível em: Acesso em: 13 nov. 2025.
- [4] MOHAMED, A. B.; MIZAN, M. F. R.; HASAN, A. F. A Survey on Internet of Things: Architectures, Protocols, and Applications. In: 2018 4th International Conference on Advances in Electrical Engineering (ICAEE), Dhaka, 2018. Anais [...] IEEE, 2018.
- [5] WANG, L.; WANG, X.; WANG, L. Industrial Internet of Things for smart manufacturing: a review. *Journal of Manufacturing Systems*, v. 48, p. 3-16, jul. 2018.
- [6] TSUI, K. L.; CHEN, C.; ZHOU, G. Industrial big data analytics for predictive maintenance in smart manufacturing. *Journal of Intelligent Manufacturing*, v. 28, n. 7, p. 1693-1709, out. 2017.

- [7] ADEEL, M.; SULEMAN, M.; ALI, I. Narrowband Internet of Things (NB-IoT): A survey. *Journal of Network and Computer Applications*, v. 104, p. 23-39, fev. 2018.
- [8] MOHAMMADZADEH, M.; HUSSEIN, M. R. Narrowband Internet of Things (NB-IoT): A comprehensive survey. *Ad Hoc Networks*, v. 97, p. 102025, fev. 2020.
- [9] ZHANG, H.; LIU, Y.; WANG, X.; CHEN, J. *Anomaly Detection in Industrial Processes Using an Improved YOLO-Based Deep Learning Approach*. *Sensors*, v. 23, n. 12, p. 5438, 2023.
- [10] CHOI, E.; CHOI, S.; AN, K.; KANG, K. T. Deep Learning-Based Inkjet Droplet Detection for Jetting Characterizations and Multijet Synchronization. *Journal of Imaging Science and Technology*, v. 65, n. 4, p. 040401-1–040401-10, jul./ago. 2021.
- [11] LIU, J.; CAI, Z.; HE, K.; HUANG, C.; LIN, X.; LIU, Z.; LI, Z.; CHEN, M. An Efficient Printing Defect Detection Based on YOLOv5-DCN-LSK. *Electronics*, v. 12, n. 13, p. 2962, jul. 2023.
- [12] VAN PEIJPE, C.; GHANIPOOR, F.; DE LOORE, Y.; HACKING, P.; VAN DE WOUW, N.; ESFAHANI, P. M. Fault Isolation for the Ink Deposition Process in High-End Industrial Printers. In: 2020 IEEE CONFERENCE ON DECISION AND CONTROL (CDC), Jeju Island, Republic of Korea, 2020. **Anais [...]**. IEEE, 2020.
- [13] BISCHOFF, P.; ZEH, C.; SCHUSTER, C.; HÄRTLING, T.; KROH, C. *En route to automated maintenance of industrial printing systems: digital quantification of print-quality factors based on induced printing failure*. *Journal of Sensors and Sensor Systems*, v. 11, p. 277–287, 2022.
- [14] HT MICRON. *HTNB32L-XXX-SDK: Hardware User Manual and Datasheet*. Disponível em: <https://github.com/htmicron/HTNB32L-XXX-SDK/blob/main/docs/HTNB32L-XXX%20Hardware%20User%20Manual%20and%20Datasheet.pdf>. Acessado em 06 de Outubro de 2025.
- [15] SKRZYPEK, B. et al. *MakeSense.ai – Free Annotation Tool for Object Detection and Segmentation*. Disponível em: <https://www.makesense.ai>. Acesso em: out. 2025.
- [16] TELEGRAM. *Telegram Bot API Documentation*. Disponível em: <https://core.telegram.org/bots/api>. Acesso em out 2025.
- [17] TANENBAUM, A. S.; WETHERALL, D. *Computer Networks*. 6th ed. Pearson, 2020.
- [18] PLATFORMIO. *PlatformIO IDE for VSCode – Documentation*. 2024. Disponível em: <https://docs.platformio.org>. Acesso em out 2025.
- [19] ESPRESSIF SYSTEMS. *ESP32-C3 Datasheet & Arduino Core Documentation*. Espressif, 2023. Disponível em: <https://docs.espressif.com>. Acesso em out 2025.
- [20] BRADSKI, G.; KAEHLER, A. *Learning OpenCV 4: Computer Vision with Python and C++*. O'Reilly Media, 2020.
- [21] JOERGENSEN, G. et al. *YOLOv5 Documentation*. Ultralytics, 2023. Disponível em: <https://docs.ultralytics.com/pt/yolov5/>. Acesso em out 2025.
- [22] A. V. O. B. dos Santos, “inkjetErrorIdentifier – Source code repository,” GitHub, 2025. Disponível: <https://github.com/Andrewtk9/inkjetErrorIdentifier>. Acesso em Nov 2025.

Nota: o título original o Projeto Final de Curso (PFC) foi alterado

De:

“Aplicação de NB-IoT em monitoramento de equipamentos industriais: coleta de dados e integração com a nuvem para manutenção preventiva”

Para:

“INTEGRAÇÃO DE IA E NB-IOT NO MONITORAMENTO DE EQUIPAMENTOS INDUSTRIAIS PARA CONTROLE DE QUALIDADE”.



Pontifícia Universidade Católica de Goiás
Pontifical Catholic University of Goiás
Av. Universitária, 1009, Setor Universitário
Caixa Postal 86 - CEP 74.605-010
Goiânia - Goiás - Brasil

RESOLUÇÃO nº 038/2020 – CEPE

ANEXO I

APÊNDICE ao TCC

Termo de autorização de publicação de produção acadêmica

O estudante ANDRÉ VICTOR DE OLIVEIRA BARROS DOS SANTOS do Curso de ENGENHARIA DE CONTROLE E AUTOMAÇÃO matrícula 2021.1.0118.0003-5, telefone: 62-9-9652-6585 e-mail 20211011800035@pucgo.edu.br, na qualidade de titular dos direitos autorais, em consonância com a Lei nº 9.610/98 (Lei dos Direitos do Autor), autoriza a Pontifícia Universidade Católica de Goiás (PUC Goiás) a disponibilizar o Trabalho de Conclusão de Curso intitulado APLICAÇÃO DE NB-IOT EM MONITORAMENTO DE EQUIPAMENTOS INDUSTRIAIS: COLETA DE DADOS E INTEGRAÇÃO COM A NUVEM PARA MANUTENÇÃO PREVENTIVA, gratuitamente, sem ressarcimento dos direitos autorais, por 5 (cinco) anos, conforme permissões do documento, em meio eletrônico, na rede mundial de computadores, no formato especificado (Texto(PDF); Imagem (GIF ou JPEG); Som (WAVE, MPEG, AIFF, SND); Vídeo (MPEG, MWV, AVI, QT); outros, específicos da área; para fins de leitura e/ou impressão pela internet, a título de divulgação da produção científica gerada nos cursos de graduação da PUC Goiás.

Goiânia, 18 de setembro de 2025.

 Documento assinado digitalmente
ANDRÉ VICTOR DE OLIVEIRA BARROS DOS SANTOS
Data: 18/09/2025 17:24:39-0300
Verifique em: https://portal.br.gov.br

Assinatura do autor: _____

Nome completo do autor: ANDRÉ VICTOR DE OLIVEIRA BARROS DOS SANTOS

 Documento assinado digitalmente
MARCOS ANTÔNIO DE SOUSA
Data: 18/09/2025 16:54:39-0300
Verifique em: https://portal.br.gov.br

Assinatura do professor-orientador: _____

Nome completo do professor-orientador: MARCOS ANTÔNIO DE SOUSA

Escola Politécnica e de Artes - setembro de 2025