



Immersive Technologies for Augmented Visual Control in Smart Manufacturing

MARIANA RODRIGUES ALMEIDA

Setembro de 2025

Immersive Technologies for Augmented Visual Control in Smart Manufacturing

Mariana Almeida

**Dissertação para obtenção do Grau de Mestre em
Engenharia Informática, Área de Especialização em
Engenharia de Dados**

**Orientador: Paula Viana
Supervisor: Pedro Carvalho**

Declaração de Integridade

Declaro ter conduzido este trabalho académico com integridade.

Não plagiei ou apliquei qualquer forma de uso indevido de informações ou falsificação de resultados ao longo do processo que levou à sua elaboração.

Portanto, o trabalho apresentado neste documento é original e de minha autoria, não tendo sido utilizado anteriormente para nenhum outro fim.

Declaro ainda que tenho pleno conhecimento do Código de Conduta Ética do P.PORTO.

ISEP, Porto, 27 de setembro de 2025

Resumo

O avanço tecnológico tem vindo a transformar os processos industriais, abrindo caminho para a automatização de tarefas críticas, como a inspeção visual no controlo de qualidade. Este trabalho insere-se neste contexto, ao explorar o potencial da Realidade Aumentada (RA) como ferramenta de apoio à deteção de defeitos em embalagens plásticas, contribuindo para práticas de sustentabilidade industrial. Desenvolvido em colaboração com a empresa Neutroplast, no âmbito do projeto Sustainable Plastics, o protótipo proposto integra alertas multimodais e modos de validação ajustáveis, com vista à automatização do processo de inspeção. A aplicação foi concebida para ser utilizada com óculos de RA, promovendo uma interação intuitiva e adaptada ao contexto industrial. A avaliação envolveu vinte e um participantes e combinou métodos quantitativos e qualitativos. Os resultados revelam uma aceitação positiva, com destaque para a clareza dos alertas e a simplicidade da interação. Foram igualmente apontadas limitações ao nível do conforto visual e da consistência da experiência entre diferentes formas de interação. No conjunto, os dados confirmam o potencial da abordagem proposta e evidenciam oportunidades de melhoria para futuras versões do protótipo.

Palavras-chave: Realidade Aumentada, Óculos de RA, Inspeção Visual, Controlo de Qualidade, Automação de Processos, Sustentabilidade Industrial

Abstract

Technological advances have been transforming industrial processes, paving the way for the automation of critical tasks such as visual inspection in quality control. This work fits into this context by exploring the potential of Augmented Reality (AR) as a tool to support the detection of defects in plastic packaging, contributing to industrial sustainability practices. Developed in collaboration with the company Neutroplast, as part of the Sustainable Plastics project, the proposed prototype integrates multimodal alerts and adjustable validation modes, with a view to automating the inspection process. The application was designed to be used with AR glasses, promoting intuitive interaction adapted to the industrial context. The evaluation involved twenty-one participants and combined quantitative and qualitative methods. The results reveal positive acceptance, with emphasis on the clarity of the alerts and the simplicity of interaction. Limitations were also noted in terms of visual comfort and consistency of experience between different forms of interaction. Overall, the data confirm the potential of the proposed approach and highlight opportunities for improvement in future versions of the prototype.

Conteúdo

Lista de Figuras	xi
Lista de Tabelas	xiii
Lista de Algoritmos	xiii
Lista de Código	xiii
Lista de Acrónimos	xv
1 Introdução	1
1.1 Contexto e Motivação	1
1.2 Objetivos	2
1.3 Considerações Éticas	3
1.4 Estrutura do Documento	3
2 Estado da Arte	5
2.1 Realidade Aumentada	5
2.1.1 Tracking	6
2.1.2 Displays	9
2.1.3 Motores de Desenvolvimento de Jogos	12
2.1.4 Software Development Kit	13
2.1.5 Visualização e Interação	13
2.1.6 Avaliação de Sistemas de RA	14
2.1.7 Desafios Técnicos na Aplicação de RA	15
2.2 Trabalhos Relacionados	15
2.2.1 Medicina	15
2.2.2 Turismo	18
2.2.3 Arquitetura e Construção	19
2.2.4 Indústria	21
2.2.5 Síntese Comparativa	23
3 Desenho	25
3.1 Cenário Narrativo de Utilização	25
3.2 Requisitos Funcionais	26
3.3 Requisitos Não Funcionais	27
3.4 Esquema dos Dados Recebidos	27
3.5 Fluxos de Decisão	27
3.5.1 Emissão de Alerta	28
3.5.2 Modos de Inspeção	28

4	Implementação	31
4.1	Arquitetura Geral	31
4.2	Dados de entrada	32
4.3	Servidor	34
4.4	Aplicação de Realidade Aumentada	36
4.4.1	Ferramentas de suporte	36
4.4.2	Implementação da aplicação	37
4.4.2.1	Alertas	38
4.4.2.2	Modos de Inspeção	41
4.4.2.3	Interação	45
4.4.2.4	Configuração e Personalização	46
5	Avaliação	49
5.1	Metodologia	49
5.2	Questionário de Avaliação	50
5.3	Análise dos Resultados	50
5.3.1	Caracterização da amostra	51
5.3.2	Resultados Quantitativos	51
5.3.3	Resultados Qualitativos	54
5.3.4	Comparação entre Perfis de Participantes	54
5.4	Discussão	56
6	Conclusões e Trabalho Futuro	59
6.1	Conclusões	59
6.2	Trabalho Futuro	60
	Bibliografia	61
	Apêndice A Documentos Éticos e Legais	67
	Apêndice B Criação de um Dataset	71
B.1	Constituição do <i>Dataset</i>	71
B.1.1	Descrição dos Defeitos	71
B.1.2	Metodologia de Aquisição	73
B.1.2.1	<i>Aquisição de Imagens</i>	73
B.1.3	Estrutura do <i>Dataset</i>	74
B.2	Preparação do <i>Dataset</i>	75
	Apêndice C Questionário	77

Lista de Figuras

1.1	Fluxo global do sistema de inspeção	2
2.1	<i>Continuum</i> da Realidade–Virtualidade (Adaptada de [10])	6
2.2	Métodos de <i>tracking</i> em RA (Adaptada de [11])	7
2.3	Exemplo de <i>marker-based tracking</i> (Adaptada de [13])	8
2.4	Exemplos de <i>markerless tracking</i> (Adaptada de [7])	8
2.5	Arquiteturas de <i>tracking Outside-In</i> (à esquerda) e <i>Inside-Out</i> (à direita) (Adaptada de [14])	9
2.6	Representação simplificada de <i>video see-through display</i> (esquerda) e <i>optical see-through display</i> (direita) (Adaptada de [16])	10
2.7	Sistema desenvolvido por Wen et al. (Adaptada de [46])	16
2.8	Sistema desenvolvido por Fick et al. numa plataforma web (em cima) e num dispositivo de RA (em baixo) (Adaptada de [47])	16
2.9	Sistema desenvolvido por Ivanov et al. (Adaptada de [48])	17
2.10	Sistema desenvolvido por Al Janabi et al. (Adaptada de [49])	17
2.11	Sistema Archeoguide desenvolvido por Vlahakis et al. (Adaptada de [50])	18
2.12	Sistema ToARist desenvolvido por Williams et al. (Adaptada de [52])	19
2.13	Sistema desenvolvido por Chen et al., com <i>interface</i> aumentada no local (à esquerda) e remota (à direita) (Adaptada de [53])	20
2.14	Arquitetura da aplicação desenvolvida por Pan e Isnaeni (Adaptada de [54])	20
2.15	Sistema desenvolvido por Fiorentino et al. (Adaptada de [56])	21
2.16	Sistema desenvolvido por Marino et al. (Adaptada de [57])	22
2.17	Sistema desenvolvido por Seeliger et al. (Adaptada de [2])	22
2.18	Infraestrutura proposta por Alves et al. (Adaptada de [58])	23
2.19	Sistema desenvolvido por Szajna et al. (Adaptada de [59])	23
3.1	Fluxo global de utilização do protótipo	26
3.2	Árvore de decisão para emissão de alertas	28
3.3	Árvore de decisão para os modos de inspeção	29
4.1	Arquitetura geral do sistema	31
4.2	Conjunto de dados de referência	33
4.3	Exemplos de imagens adquiridas, identificadas de a) a j)	33
4.4	Objeto <code>MQTTClient</code> no <i>Inspector</i> do <i>Unity</i> , parâmetros <code>brokerIp</code> e <code>brokerPortValue</code> visíveis para configuração.	35
4.5	Óculos <i>Meta Quest 3S</i> (Adaptada de [62])	36
4.6	Relação entre os principais <i>GameObjects</i> da cena <i>Unity</i> e os respetivos <i>scripts</i> de implementação	37
4.7	<i>Inspector</i> para configuração do <code>AudioSource</code> para alerta sonoro	38
4.8	<i>Interfaces</i> relativas ao alerta visual	40
4.9	<i>Interface</i> de configuração do modo de inspeção	42
4.10	<i>Interface</i> do modo de inspeção simples	42

4.11	<i>Interface</i> do modo de inspeção complexo	43
4.12	<i>Interface</i> com sobreposição de defeitos: caixas coloridas por tipo e etiquetas com nome e confiança	45
4.13	Exemplo de suporte a interação por mãos e controladores	46
4.14	<i>Interface</i> com diferentes níveis de opacidade	47
4.15	Exemplo de <i>zoom</i> na imagem de inspeção	48
5.1	Distribuição da experiência prévia com óculos de RA	51
5.2	Resultados da média e desvio padrão das questões de usabilidade	52
5.3	Resultados da média e desvio padrão das questões de conforto	53
5.4	Resultados da média e desvio padrão das questões de utilidade	53
5.5	Distribuição das preferências dos participantes	54
5.6	Comparação das médias das respostas por experiência prévia em RA	55
5.7	Comparação dos desvios padrão das respostas por experiência prévia em RA	56
A.1	Documento Política de Privacidade	69
A.2	Documento Termos e Condições	70
B.1	Catálogo com defeitos visuais	72
B.2	Tipos de embalagens de plástico consideradas	74
B.3	Exemplos de defeitos considerados	75
B.4	Exemplos de embalagens com as <i>bounding boxes</i>	76
C.1	Questionário de avaliação	81

Lista de Tabelas

2.1 Síntese comparativa dos trabalhos relacionados	24
--	----

Lista de Acrónimos

BIM	<i>Building Information Modeling.</i>
CVAT	<i>Computer Vision Annotation Tool.</i>
GIMP	<i>GNU Image Manipulation Program.</i>
GPS	<i>Global Positioning System.</i>
HMDs	<i>Head-Mounted Displays.</i>
HWDs	<i>Head-Worn Displays.</i>
IDE	<i>Integrated Development Environment.</i>
MEC	<i>Mobile Edge Computing.</i>
MQTT	<i>Message Queuing Telemetry Transport.</i>
POI	<i>Points of Interest.</i>
PSSUQ	<i>Post-Study System Usability Questionnaire.</i>
QR	<i>Quit Response.</i>
RA	Realidade Aumentada.
RM	Realidade Mista.
RV	Realidade Virtual.
SDK	<i>Software Development Kit.</i>
SLAM	<i>Simultaneous Localization and Mapping.</i>
TXT	<i>Text File.</i>
UPM	<i>Unity Package Manager.</i>
VA	Virtualidade Aumentada.
VIO	<i>Visual Inertial Odometry.</i>

Capítulo 1

Introdução

Este capítulo apresenta o contexto e a motivação que sustentam este trabalho, centrado na exploração do potencial da Realidade Aumentada como tecnologia inovadora para apoiar a automatização da inspeção visual no processo de controlo de qualidade em ambientes fabris. São ainda definidos os principais objetivos da dissertação, abordadas as respetivas considerações éticas, e apresentada a estrutura geral do documento.

1.1 Contexto e Motivação

A qualidade dos produtos é um requisito central em qualquer linha de produção industrial, sendo a inspeção visual manual uma das práticas mais utilizadas devido à sua simplicidade e flexibilidade. No entanto, este método depende fortemente da perceção e do julgamento humano, o que o torna vulnerável a fatores técnicos, psicofísicos, organizacionais, ambientais e sociais. Estas limitações podem comprometer a consistência e a eficácia das decisões, sobretudo em contextos de elevada cadência produtiva ou quando os defeitos surgem em áreas de difícil deteção [1, 2].

Neste cenário, a Indústria 4.0 assume um papel relevante ao promover a digitalização e a integração de tecnologias inteligentes nos processos produtivos, aumentando a eficiência, flexibilidade e adaptabilidade. Entre estas tecnologias, a Realidade Aumentada (RA) destaca-se pela capacidade de sobrepor informação relevante diretamente no campo de visão do operador, de forma contextual e em tempo real, o que reduz a carga cognitiva e apoia decisões mais rápidas e precisas [3, 4].

Para além destes contributos, a RA abre caminho a benefícios adicionais, como a possibilidade de realizar inspeções à distância, a inclusão de operadores com mobilidade reduzida e uma gestão mais flexível do tempo de trabalho. O recurso a dispositivos de RA permite apresentar a informação diretamente diante do utilizador, mantendo-o focado na tarefa e libertando-o de suportes físicos adicionais. Num plano mais estratégico, a introdução de tecnologias imersivas em contexto industrial poderá ainda atrair novos perfis profissionais, mais familiarizados com ambientes digitais e *interfaces* avançadas.

A relevância da RA aumenta quando combinada com modelos de visão computacional capazes de detetar e classificar defeitos de forma automática, fornecendo informação adicional que pode ser sobreposta no campo de visão do utilizador [4]. Esta integração permite disponibilizar informações visuais enriquecidas, que facilitam a deteção de defeitos e elevam a qualidade do controlo final.

Neste contexto, o presente trabalho tem como objetivo explorar a aplicação da RA no apoio à inspeção de defeitos em embalagens plásticas. O protótipo foi desenvolvido no âmbito do

projeto *Sustainable Plastics*, uma Agenda Mobilizadora para os Plásticos Sustentáveis em Portugal, cujo objetivo é alavancar a transição do setor dos plásticos para uma economia circular, promovendo a sustentabilidade, a eficiência na utilização de recursos, a redução das emissões de gases com efeito de estufa e a criação de emprego [5]. A implementação do protótipo contou com a colaboração da *Neutroplast*, especialista na produção de embalagens primárias para os setores farmacêutico, dermocosmético e alimentar, onde a deteção rigorosa de defeitos é fundamental para garantir a qualidade e conformidade dos produtos [6]. A solução proposta pretende assim demonstrar o valor da RA como ferramenta de apoio à inspeção, aumentando a fiabilidade do controlo de qualidade e reforçando a confiança nas decisões do operador.

Para melhor enquadrar o papel do protótipo no processo de inspeção, a Figura 1.1 apresenta o fluxo global do sistema. A sequência inicia-se na linha de montagem, com a captura da imagem da embalagem, posteriormente analisada por um módulo de visão computacional responsável pela identificação de potenciais defeitos. A informação relativa à localização e ao tipo de defeito é então enviada, em conjunto com a imagem, para o módulo de RA. É neste último bloco que se centra o presente trabalho, disponibilizando ao operador a informação de forma contextual e interativa para apoiar a decisão durante a inspeção.

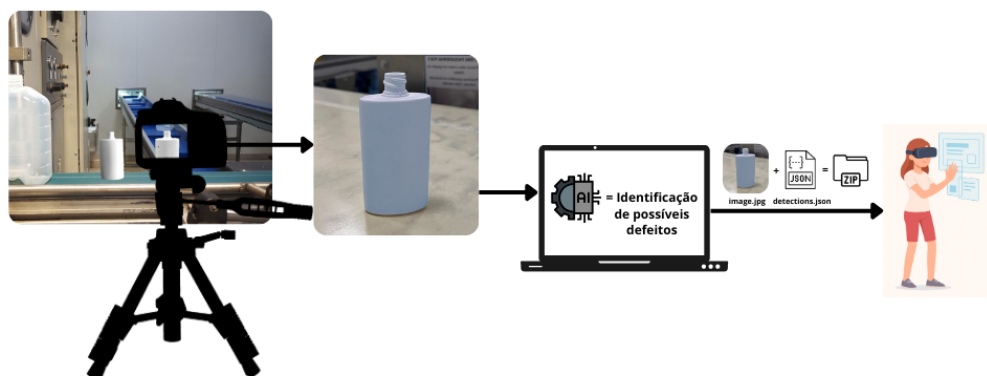


Figura 1.1: Fluxo global do sistema de inspeção

1.2 Objetivos

O principal objetivo deste trabalho é explorar o potencial da RA através do desenvolvimento de um protótipo para apoiar a inspeção visual de embalagens plásticas. Pretende-se avaliar de que forma uma aplicação em RA pode apoiar os operadores na validação de defeitos, aumentando a clareza da informação e oferecendo diferentes formas de interação. Para a implementação recorreu-se aos óculos *Meta Quest 3S*, utilizados como exemplo de dispositivo para validar o protótipo. Para concretizar este objetivo geral, foram definidos quatro objetivos específicos:

- **Explorar modalidades de alerta**, avaliando diferentes formas de notificação (sonora, visual e adaptativa ao ambiente), de modo a identificar qual se revela mais eficaz para captar a atenção do operador em tempo útil;
- **Explorar estratégias de apresentação da informação**, centradas na forma como os defeitos são disponibilizados ao operador, através de três modos de inspeção distintos: validação global de todos os defeitos em simultâneo, análise defeito a defeito e seleção

adaptativa em função da quantidade de anomalias. O objetivo é garantir clareza na visualização e adequação ao contexto de utilização;

- **Explorar mecanismos de adaptação da interface**, incluindo funcionalidades de ajuste de parâmetros como zoom, opacidade e intensidade dos alertas visuais, de forma a permitir que a aplicação se ajuste às preferências do operador e a diferentes condições de utilização;
- **Explorar métodos de interação multimodal**, considerando interações baseadas em mãos virtuais e controladores físicos, de forma a proporcionar uma experiência de utilização intuitiva e ajustável às condições de operação.

Ainda como parte integrante do trabalho, pretende-se fazer a avaliação do protótipo com um conjunto de participantes, através de uma sessão experimental seguida de um questionário, permitindo a recolha de evidências sobre o seu desempenho.

1.3 Considerações Éticas

Durante o desenvolvimento deste trabalho foram tidos em conta aspetos éticos relacionados com a utilização da aplicação e com a recolha de dados no âmbito da sua avaliação. Embora a aplicação de RA não recolha nem armazene dados pessoais, a avaliação com participantes exigiu a criação de mecanismos que assegurassem a proteção da privacidade e a transparência no tratamento da informação. Para tal, foram elaborados dois documentos de suporte:

- **Política de Privacidade da Avaliação:** descreve de forma clara os dados recolhidos através do questionário, a sua utilização exclusiva para fins académicos, o período de retenção, as medidas de segurança aplicadas e os direitos dos participantes relativamente às suas respostas;
- **Termos e Condições de Utilização:** estabelecem as regras associadas à utilização do protótipo, incluindo as responsabilidades do utilizador, os direitos de propriedade intelectual e as limitações de responsabilidade, garantindo a transparência e a proteção de todas as partes envolvidas.

Ambos os documentos foram disponibilizados aos participantes na resposta ao questionário, assegurando que a sua participação foi informada, voluntária e consentida. Os documentos completos encontram-se disponíveis no Anexo A.

1.4 Estrutura do Documento

Este documento encontra-se organizado em seis capítulos. No primeiro é feita a introdução ao tema, apresentando a motivação, objetivos, considerações éticas e a organização do trabalho. O segundo capítulo aborda o estado da arte, enquadrando conceitos fundamentais da RA e aplicações em diversos contextos. O terceiro apresenta o desenho do protótipo, incluindo requisitos, fluxos de decisão e modelo conceptual. No quarto capítulo é descrita a implementação do protótipo, com a respetiva arquitetura e principais componentes. O quinto apresenta a avaliação realizada, cobrindo metodologia, questionário e análise dos resultados. Por fim, o sexto capítulo sintetiza as conclusões e aponta direções para trabalhos futuros.

Capítulo 2

Estado da Arte

Este capítulo apresenta os conceitos fundamentais da RA, abordando os princípios que a suportam e os principais componentes necessários ao seu funcionamento. São também analisados trabalhos prévios que demonstram a aplicação desta tecnologia em diferentes domínios, como turismo, educação, saúde e indústria.

2.1 Realidade Aumentada

O termo Realidade Aumentada (RA) remonta a 1963, quando Ivan Sutherland [7] desenvolveu o *Sketchpad*, reconhecido como o primeiro programa gráfico interativo e frequentemente referido como o ponto de partida para as experiências de RA. Desde então, o conceito foi definido de diversas formas, sendo a definição mais amplamente aceite a proposta por Azuma [8], que descreve a RA como uma tecnologia capaz de enriquecer a percepção do mundo real ao sobrepor informações digitais a este. Segundo Azuma, os sistemas de RA apresentam três características fundamentais [8]:

- **Combinação de elementos reais e virtuais:** combinação de elementos do mundo real com objetos e informações virtuais, criando um ambiente misto no qual os utilizadores podem interagir com ambos os mundos em simultâneo;
- **Interatividade em tempo real:** resposta imediata às ações do utilizador, fornecendo *feedback* e orientação imediata;
- **Registo tridimensional:** alinhamento preciso dos elementos virtuais com o mundo real, utilizando técnicas de *tracking* que garantem o posicionamento correto dos elementos virtuais no ambiente físico.

Em 1994, Paul Ingram e Fumio Kishino [9] introduziram o conceito de *continuum* da realidade-virtualidade (Figura 2.1), que define as diferentes classes de tecnologias imersivas e as posiciona num espetro contínuo. Neste espetro, os ambientes totalmente reais encontram-se no extremo esquerdo e os ambientes completamente virtuais no extremo direito.

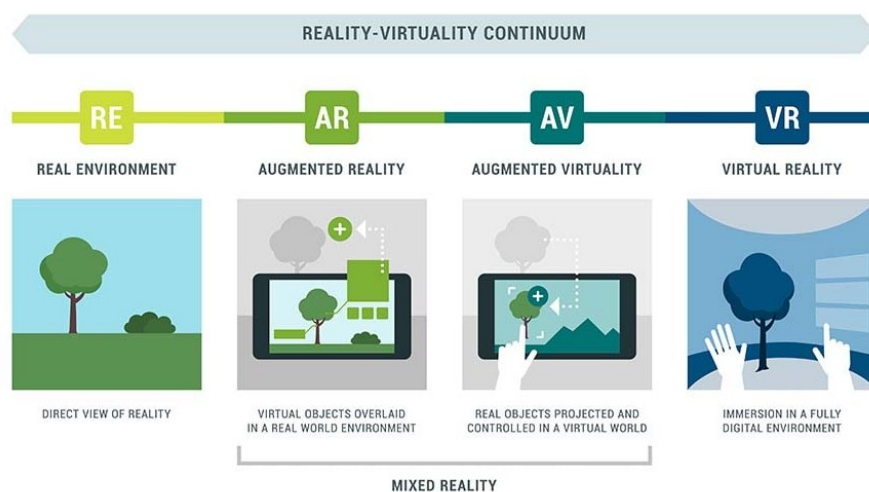


Figura 2.1: *Continuum* da Realidade–Virtualidade (Adaptada de [10])

De forma simplificada, o *continuum* pode ser dividido em três zonas principais [9]:

- **Extremo esquerdo:** ambientes reais, observados diretamente ou através de vídeo convencional;
- **Extremo direito:** Realidade Virtual (RV), composta por ambientes gerados por simulação digital, que proporcionam uma imersão total num espaço virtual, completamente isolado do mundo físico;
- **Zona intermédia:** ambientes de Realidade Mista (RM), que englobam todas as combinações possíveis de elementos reais e virtuais num único ambiente, incluindo:
 - Realidade Aumentada (RA): mais próxima do extremo real, enriquecendo o ambiente físico com elementos digitais;
 - Virtualidade Aumentada (VA): mais próxima do extremo virtual, incorporando elementos do mundo real em ambientes predominantemente digitais.

2.1.1 Tracking

O *tracking*, em aplicações de RA, corresponde ao processo de determinar continuamente a posição e orientação do utilizador ou de objetos relevantes, no espaço físico. Este processo é essencial para que os elementos virtuais sejam corretamente posicionados e alinhados com o ambiente real, permitindo uma experiência imersiva e iterativa [7]. Nesta secção analisam-se os principais métodos de *tracking* aplicados em RA, conforme ilustrado na Figura 2.2.

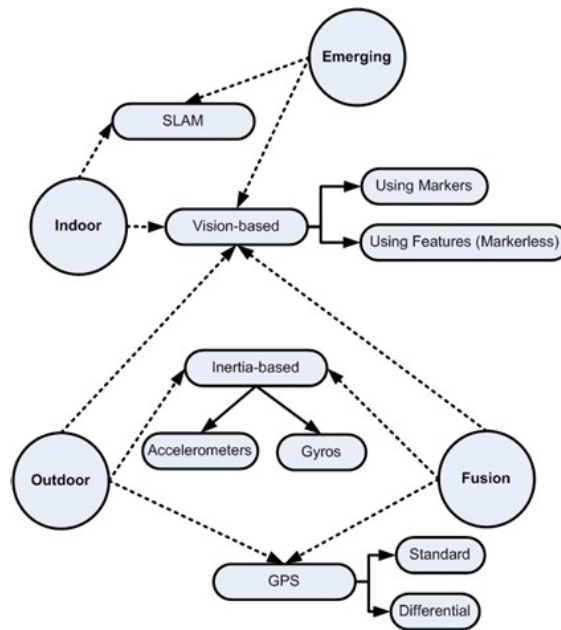


Figura 2.2: Métodos de *tracking* em RA (Adaptada de [11])

Nos métodos *indoor*, o ambiente tende a ser mais estruturado e previsível, com dimensões fixas e movimentos limitados por parte do utilizador. Nestas situações, é comum o uso de marcadores fiduciais, bem como técnicas baseadas em visão computacional com câmaras, códigos *Quit Response* (QR) ou sensores de movimento. Este tipo de configuração, associado a um ambiente controlado, contribui para níveis mais elevados de precisão no *tracking*. Por outro lado, em ambientes *outdoor*, o espaço é mais amplo e menos previsível, sendo geralmente inviável preparar o ambiente de forma antecipada para o *tracking*. Nestes casos, tecnologias como *Global Positioning System* (GPS) e sensores inerciais são frequentemente utilizadas para estimar a localização e orientação do utilizador. Estes ambientes apresentam desafios adicionais, como variações de iluminação natural e a necessidade de detetar marcadores naturais, em vez de depender dos artificiais [11].

Os sistemas de *tracking* podem ser classificados segundo a dependência ou não de marcadores artificiais visuais e é neste sentido que se distinguem duas abordagens principais: os métodos *marker-based* e os métodos *markerless* [7, 11, 12]. O *Tracking marker-based* (Figura 2.3) baseia-se na deteção de marcadores artificiais (fiduciais), como códigos QR, padrões *ARToolKit* ou LEDs infravermelhos. Estes marcadores são identificados por câmaras e utilizados para calcular a pose do dispositivo em relação ao mundo real. Este método é amplamente utilizado em ambientes controlados, dada a sua precisão e fiabilidade, mas exige preparação prévia e visibilidade constante dos marcadores [11].



Figura 2.3: Exemplo de *marker-based tracking* (Adaptada de [13])

Por outro lado, os métodos *markerless* (Figura 2.4) dispensam marcadores físicos e utilizam informações extraídas diretamente do ambiente. Podem assumir diferentes formas, conforme a fonte de dados:

- *Sensor-based*, que utiliza sensores físicos como acelerômetros, giroscópios, magnetômetros e GPS. Estes sensores oferecem altas taxas de atualização e resistência a movimentos abruptos, mas também sofrem de problemas de acumulação de erro ao longo do tempo [11];
- *Vision-based*, que recorre a câmaras para extrair características visuais do ambiente (texturas, bordas, formas) e estimar a pose com base nessas informações. Este método é preciso, mas sensível a condições de iluminação, oclusões ou superfícies uniformes [7, 12];
- Híbrido, que combina múltiplas fontes como sensores inerciais e visão computacional para compensar as limitações de cada abordagem individual. Um exemplo é a integração de visão com *Simultaneous Localization and Mapping* (SLAM), que permite mapear o ambiente e localizar o utilizador em simultâneo, mesmo em cenários dinâmicos e não preparados [7, 12].

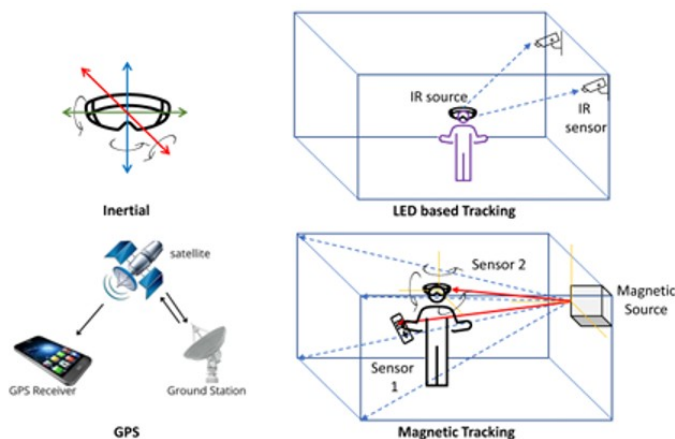


Figura 2.4: Exemplos de *markerless tracking* (Adaptada de [7])

Além do tipo de dados utilizado, outra distinção importante está relacionada com a posição dos sensores no sistema. Assim, os sistemas de *tracking* podem ser implementados em duas arquiteturas, como ilustrado na Figura 2.5 [11, 14]:

- *Outside-in*: os sensores estão instalados de forma fixa no ambiente ("outside"), como em paredes ou tetos, e detetam marcadores fiduciais que estão colocados normalmente no utilizador ("in"), permitindo calcular a sua posição e orientação. Esta configuração oferece elevada precisão, mas exige infraestrutura externa;
- *Inside-out*: o utilizador transporta os sensores ("inside"), normalmente integrados em dispositivos como *smartphones*, *tablets* ou óculos de RA, e reconhece marcadores distribuídos no ambiente ("out") para determinar a sua própria localização. Esta é uma abordagem comum em dispositivos móveis.

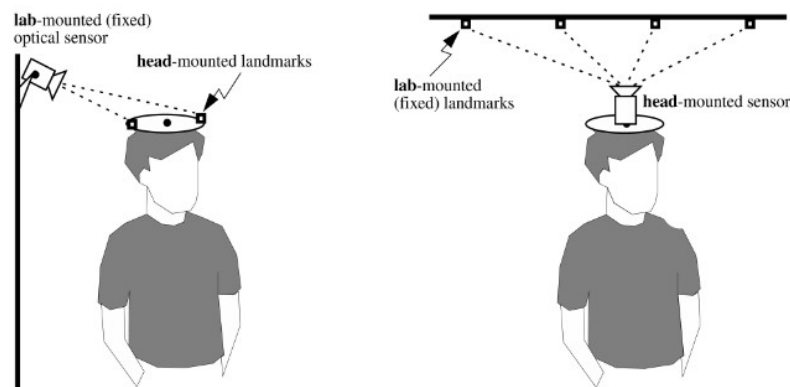


Figura 2.5: Arquiteturas de *tracking* *Outside-In* (à esquerda) e *Inside-Out* (à direita) (Adaptada de [14])

A escolha da abordagem de *tracking* depende do contexto de uso. Sistemas baseados em marcadores são eficazes em ambientes controlados, enquanto soluções *markerless*, especialmente as híbridas, oferecem maior adaptabilidade e escalabilidade. Em ambientes industriais ou não preparados, técnicas como SLAM combinadas com sensores inerciais revelam-se particularmente eficazes. Apesar dos avanços, os sistemas de *tracking* continuam a enfrentar desafios significativos. Entre eles incluem-se a sensibilidade à iluminação e textura nos métodos baseados em visão, a acumulação de erro nos sensores inerciais, interferências no sinal GPS, e os custos computacionais elevados de algoritmos de processamento de imagem [7, 11].

2.1.2 Displays

Azuma et al. [4] abordou diferentes tipos de *displays* utilizados em sistemas de RA, destacando três principais categorias:

- *Head-Worn Displays* (HWDs) também conhecidos como *Head-Mounted Displays* (HMDs): são dispositivos montados na cabeça do utilizador e dividem-se em dois tipos principais, como se pode observar na Figura 2.6 [4]:
 - *Optical see-through*: permitem que o utilizador veja o mundo real através de uma janela transparente, sobre a qual são projetadas informações virtuais. Uma das suas principais vantagens é permitir uma experiência visual mais natural, o que os torna adequados para situações em que a perceção direta do ambiente

físico é essencial [4, 15]. No entanto, estes dispositivos enfrentam limitações em termos de brilho, resolução, campo de visão e contraste, o que dificulta uma fusão visual convincente entre os elementos reais e virtuais. Além disso, não conseguem ocultar objetos reais, o que compromete certos tipos de aplicações [4];

- *Video see-through*: utilizam câmaras para capturar o ambiente real e apresentam as imagens combinadas com elementos virtuais num ecrã opaca. Estes permitem maior controlo sobre a fusão visual e podem ocultar elementos reais, mas sofrem com problemas como erro de paralaxe, causado pela posição deslocada das câmaras em relação aos olhos, e com atrasos (latência), que afetam a precisão da sobreposição. Algumas versões mais avançadas alinham os eixos óticos da captura e da exibição com cada olho, o que melhora significativamente a perceção visual [4].

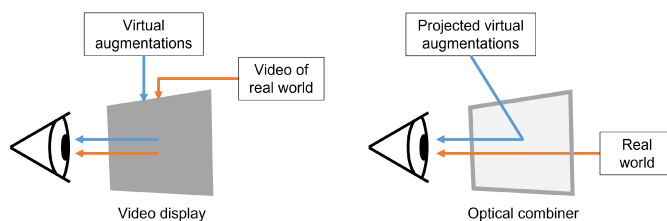


Figura 2.6: Representação simplificada de *video see-through display* (esquerda) e *optical see-through display* (direita) (Adaptada de [16])

Apesar das vantagens em termos de controlo visual e qualidade gráfica, os HMDs podem apresentar desvantagens importantes: são frequentemente pesados, caros, e podem exigir que o utilizador utilize câmaras e *hardware* montados na cabeça, o que compromete o conforto e a naturalidade da interação [15].

- *Handheld Displays*: referem-se a dispositivos móveis portáteis, como *smartphones* e *tablets*, frequentemente utilizados para aplicações de RA. Nestes dispositivos, o utilizador observa o ambiente real através da câmara e os elementos virtuais são sobrepostos na imagem exibida no ecrã. Assim, o ecrã do dispositivo funciona como uma *interface* onde o mundo físico é complementado com informação digital. Estes dispositivos são amplamente adotados devido à sua portabilidade, acessibilidade e pela existência de sensores integrados como câmaras, GPS e acelerómetros, que suportam experiências interativas em áreas como manutenção ou entretenimento [7, 15]. No entanto, apresentam algumas limitações, como o facto de exigirem que o utilizador os segure continuamente, o que pode ser desconfortável em tarefas prolongadas [4], e o tamanho reduzido do ecrã, por exemplo, dos *smartphones*, também pode dificultar a visualização de informação detalhada em cenários complexos [15].
- *Projection Displays*: são sistemas de visualização em RA que recorrem à projeção direta de imagens virtuais sobre superfícies físicas do ambiente, como mesas, paredes ou objetos. Esta abordagem dispensa o uso de dispositivos pessoais de visualização ao integrar os conteúdos digitais diretamente no espaço físico. Esta característica faz com que estes sejam particularmente eficazes em ambientes colaborativos, como museus ou laboratórios, onde diversos utilizadores podem visualizar a mesma informação em simultâneo. Apesar destas vantagens, estes sistemas apresentam limitações associadas

à sua instalação fixa, bem como à dependência da qualidade da superfície de projeção e das condições de iluminação, que afetam a visibilidade dos elementos virtuais [7, 15].

Diferentes tipos de *displays* podem ser adotados consoante as necessidades do sistema e o contexto de utilização, considerando fatores como portabilidade, naturalidade da experiência, custo e complexidade técnica.

Head-Mounted Displays

Tendo em conta que o objetivo do presente trabalho é explorar o potencial da tecnologia de RA recorrendo a HMDs, justifica-se a análise de algumas das soluções mais relevantes atualmente disponíveis. Entre estas, destaca-se o *Microsoft HoloLens*, o *Meta Quest* e o *Magic Leap*, que ilustram diferentes abordagens à visualização e interação em RA.

O *Microsoft HoloLens* é um dispositivo portátil de RM baseado em visor *optical see-through*, que permite visualizar e interagir com hologramas sobrepostos ao ambiente real. Funciona de forma autónoma, sem fios, e combina gestos, comandos de voz e *tracking* ocular para oferecer uma interação natural com conteúdos digitais. A primeira geração incluía um visor estereoscópico, sensores inerciais e câmara de profundidade, constituindo uma solução inovadora para aplicações em medicina, indústria e formação técnica. O *Microsoft HoloLens 2* introduziu melhorias substanciais, como maior resolução (2K por olho), *tracking* manual articulado, *tracking* ocular por infravermelhos e sensor *Time-of-Flight*, além de um *design* mais ergonómico com visor articulado e suporte para óculos. Corre o sistema *Windows Holographic*, permitindo integrar aplicações profissionais como o *Dynamics 365 Remote Assist* [17, 18].

A linha *Meta Quest* integra dispositivos autónomos que combinam RV e RA através de tecnologia *video see-through*. Neste caso, o utilizador observa o ambiente real por intermédio de câmaras externas, sobre o qual são projetados elementos virtuais. Esta abordagem simplifica o desenvolvimento e garante portabilidade, ao integrar processador, sensores e câmaras no próprio dispositivo. O *Quest Pro* introduziu passagem a cores de alta resolução, enquanto o *Quest 3* trouxe melhorias de desempenho gráfico e sensores de profundidade, aumentando a fidelidade da experiência. O *Quest 3S* representa uma versão mais acessível, com o mesmo processador do *Quest 3*, mas com lentes *Fresnel* e ecrãs semelhantes ao *Quest 2*, equilibrando desempenho e custo [19].

O *Magic Leap* é um dispositivo de RA concebido para contextos profissionais, com visor *optical see-through* baseado em tecnologia de projeção fotónica. Recorre ao conceito de *spatial computing*, ou seja, a integração contextual e em tempo real de elementos digitais no espaço físico do utilizador [20]. A primeira versão, o *Magic Leap One*, incluía os óculos (*Lightwear*), uma unidade portátil de processamento (*Lightpack*) e um controlador, permitindo interação através de gestos, voz e movimento. O *Magic Leap 2* trouxe melhorias ao nível do campo de visão (até 70°), lentes com escurecimento segmentado para ambientes muito iluminados e maior conforto ergonómico. Além disso, passou a suportar a norma *OpenXR*, simplificando a integração com motores como *Unity* e *Unreal*. Este dispositivo tem sido utilizado em áreas como arquitetura, retalho e medicina, pela capacidade de projetar conteúdos digitais no espaço físico de forma realista [20, 21].

2.1.3 Motores de Desenvolvimento de Jogos

Os motores de desenvolvimento de jogos, ou *game engines*, são ferramentas fundamentais que facilitam a criação de videojogos. Estes motores disponibilizam funcionalidades essenciais para tarefas como renderização gráfica, simulação física, gestão de áudio, controlo da entrada do utilizador e lógica de jogo, permitindo aos programadores concentrarem-se na mecânica e na experiência do utilizador, sem a necessidade de desenvolver todas essas funcionalidades de raiz [22]. As funcionalidades típicas de um *game engines* incluem [22]:

- **Motor de renderização:** responsável por apresentar gráficos 2D e 3D no ecrã, recorrendo a técnicas avançadas de computação gráfica;
- **Motor de física:** simula leis da física no ambiente do jogo, como colisões, gravidade ou forças dinâmicas;
- **Sistema de som:** gere o áudio, incluindo música, efeitos sonoros e locução;
- **Sistema de entrada:** trata os comandos recebidos de dispositivos externos, como teclado, rato ou comandos;
- **Lógica do jogo:** define o comportamento do jogo.

Além disso, os *game engines* modernos integram frequentemente editores de níveis, ferramentas de depuração, suporte multiplataforma, animação e sistemas de *scripting*. Estes elementos contribuem para reduzir a complexidade e os custos de produção, além de acelerarem significativamente o processo de desenvolvimento [22]. Entre os motores mais utilizados na atualidade, destacam-se dois nomes: o *Unity* e o *Unreal Engine*, amplamente adotados não só na indústria dos videojogos, mas também em áreas como simulação ou publicidade.

O *Unity* é um motor de jogo e *Integrated Development Environment* (IDE), criado com o objetivo de democratizar o desenvolvimento de jogos. Concebido para ser acessível a utilizadores com diferentes níveis de experiência, desde iniciantes a profissionais, o *Unity* destaca-se por possuir uma *interface* gráfica intuitiva, um fluxo de trabalho simplificado e uma curva de aprendizagem reduzida [23]. A sua estrutura assenta numa arquitetura baseada em componentes, onde os objetos do jogo são construídos a partir de módulos reutilizáveis. O desenvolvimento é feito principalmente em linguagem C#, sendo apoiado por uma comunidade ativa e por uma vasta biblioteca de recursos disponíveis na *Asset Store*, o que facilita o trabalho de equipas pequenas ou programadores independentes. O *Unity* é especialmente adequado a projetos de pequena e média dimensão, sendo amplamente utilizado em aplicações móveis graças à sua flexibilidade na exportação e ao suporte nativo a diferentes sistemas operativos [24].

O *Unreal*, desenvolvido pela *Epic Games*, é um motor de desenvolvimento orientado para projetos de elevada complexidade e exigência visual, sendo amplamente utilizado na produção de videojogos de grande orçamento, também conhecidos como jogos AAA. Entre os seus principais pontos fortes encontram-se: a capacidade de renderização em tempo real de alta fidelidade, a criação de ambientes visuais realistas e o suporte a efeitos gráficos avançados. O *Unreal* combina programação em C++ com um sistema de *scripting* visual designado por *Blueprints*, permitindo desenvolver lógica de jogo de forma visual, sem ser necessário escrever código diretamente, o que o torna acessível a perfis técnicos e não técnicos [22, 24]. Apesar de apresentar uma estrutura mais complexa e uma curva de aprendizagem mais acentuada em comparação com o *Unity*, o *Unreal* oferece um conjunto de ferramentas

poderosas para depuração, otimização e controlo de desempenho, sendo frequentemente a escolha preferencial em produções com elevadas exigências gráficas e realismo visual [22].

2.1.4 Software Development Kit

Os *Software Development Kit* (SDK) são conjuntos de ferramentas essenciais que permitem aos programadores desenvolver aplicações de RA. Estes *kits* integram funcionalidades como *tracking* de objetos, renderização 3D, integração com sensores e suporte a múltiplas plataformas. A escolha do SDK mais adequado depende de fatores como a plataforma-alvo, o tipo de *tracking*, os requisitos gráficos, a facilidade de integração, a licença e o suporte da comunidade [25]. Entre os SDKs mais relevantes destacam-se o *Vuforia*, um dos mais utilizados em aplicações comerciais, que oferece suporte a múltiplos tipos de alvos (imagens, objetos 3D, cilindros) e *tracking* robusto mesmo em condições adversas de iluminação. A sua integração com *Unity* tornou-o particularmente popular para prototipagem e aplicações industriais [25, 26]. A *ARKit* (Apple) e o *ARCore* (Google) representam as soluções de referência em dispositivos móveis. Ambos recorrem à técnica de *Visual Inertial Odometry* (VIO) para combinar dados de câmara com sensores inerciais, garantindo *tracking* preciso sem necessidade de marcadores. Suportam ainda superfícies planas, imagens 2D e objetos 3D, sendo amplamente utilizados em experiências móveis interativas [27]. O *Meta XR SDK*, disponibilizado pela *Meta* como um conjunto de pacotes *Unity Package Manager* (UPM), destaca-se pela sua modularidade. Inclui o *Meta XR Core SDK*, que reúne os *Building Blocks* fundamentais como *passthrough*, gestos, controladores físicos e sistemas de captura visual, formando a base para aplicações em dispositivos Quest [28]. Outros SDKs, como o *Wikitude*, que combina *tracking* visual e geolocalização, ou o *D'Fusion SDK*, orientado para aplicações industriais de elevado desempenho gráfico, também contribuíram para a evolução do setor, embora tenham hoje uma utilização mais restrita face às soluções anteriores [25, 29].

2.1.5 Visualização e Interação

A interação em aplicações de RA é identificada como um dos elementos mais críticos para o sucesso e aceitação da tecnologia. Como destacado por Carmigniani et al. [15], interações naturais e intuitivas aumentam o conforto e a eficácia dos utilizadores, sobretudo em contextos profissionais, onde métodos artificiais ou demasiado complexos podem gerar resistência. Entre as abordagens discutidas destacam-se as *interfaces* tangíveis, que recorrem à manipulação de objetos físicos como forma de interação com o conteúdo virtual, e as *interfaces* multimodais, que combinam gestos, fala, toque ou movimento ocular, oferecendo maior flexibilidade e adaptação ao utilizador.

A aceitação social dos dispositivos é outro fator relevante. Dispositivos demasiado invasivos ou visíveis, como óculos grandes ou sensores externos, podem comprometer esta aceitação, enquanto soluções mais discretas, como sensores embutidos ou gestos naturais, favorecem a integração no quotidiano [15]. Neste cenário, os comandos de voz representam uma modalidade de interação complementar, particularmente útil em situações em que as mãos estão ocupadas. Embora estudos recentes evidenciem o seu potencial em dispositivos como o *Microsoft HoloLens 2* [30] e o *Meta Quest 3* [31], a sua adoção continua condicionada por desafios como o ruído ambiental, a ambiguidade linguística e preocupações com a privacidade. Para ultrapassar estas limitações, têm sido exploradas soluções como palavras-chave de ativação, microfones direcionais e validação por modelos de linguagem. A tendência atual

aponta assim, para *interfaces* híbridas que combinam diferentes modalidades de interação, equilibrando naturalidade, eficiência e acessibilidade.

2.1.6 Avaliação de Sistemas de RA

A avaliação da eficácia e da usabilidade em aplicações de RA deve abranger não só o desempenho técnico do sistema, mas também a experiência subjetiva dos utilizadores, especialmente em contextos industriais. A literatura identifica quatro grupos principais de métricas [32–34]:

1. **Métricas de Desempenho**, avaliam a eficácia do sistema em função do impacto gerado na execução das tarefas de apoio à decisão [32]:
 - **Tempo de execução de tarefas:** mede a eficiência comparada com métodos tradicionais;
 - **Precisão nas tarefas:** avalia erros cometidos na realização de tarefas, fator crítico em processos industriais;
 - **Taxa de sucesso:** percentagem de tarefas concluídas corretamente.
2. **Métricas de Usabilidade**, analisam a facilidade de utilização e a adequação ao contexto:
 - **Facilidade de uso**, avalia a intuitividade e a facilidade de utilização da *interface* [32, 33];
 - **Capacidade de resposta do sistema:** mede a latência do *feedback* da solução de RA, assegurando resposta adequada em tempo real [33, 35];
 - **Adaptabilidade:** flexibilidade para diferentes cenários e requisitos operacionais [33, 36].
3. **Métricas de Experiência do Utilizador**, centram-se na perceção e aceitação dos operadores:
 - **Satisfação:** avaliada com questionários como o *Post-Study System Usability Questionnaire* (PSSUQ), reflete a aceitação e satisfação global do sistema [33, 37];
 - **Imersão e envolvimento:** mede até que ponto a solução mantém o utilizador focado e motivado [33, 34, 38].
4. **Métricas de Eficiência do Sistema**, consideram a viabilidade económica e técnica:
 - **Impacto no custo operacional:** avalia se os benefícios compensam o investimento [34, 39];
 - **Escalabilidade:** capacidade de suportar mais utilizadores ou maior volume de dados sem comprometer o desempenho [34].

A utilização combinada destas métricas possibilita uma avaliação abrangente da eficácia e usabilidade da RA, assegurando que a sua integração em ambientes industriais não só melhora a execução das tarefas, como é aceite e valorizada pelos operadores.

2.1.7 Desafios Técnicos na Aplicação de RA

Apesar do seu potencial no apoio à decisão industrial, a implementação da RA enfrenta vários desafios técnicos que condicionam a sua adoção em larga escala. Estes obstáculos, identificados na literatura, relacionam-se com limitações tecnológicas e dificuldades de integração em ambientes produtivos, que exigem elevados níveis de precisão e fiabilidade.

- **Hardware e desempenho:** a resolução limitada e a qualidade das projeções virtuais reduzem o realismo e a sensação de imersão. Aliado a isto, a autonomia da bateria e a capacidade de processamento dos HMDs continuam a ser insuficientes para suportar aplicações intensivas em contexto industrial, exigindo um equilíbrio delicado entre potência computacional e duração [40–42];
- **Interação e usabilidade:** criar *interfaces* intuitivas e confortáveis permanece um desafio, sobretudo em dispositivos usados por longos períodos. Movimentos complexos reduzem a eficiência em tarefas repetitivas, e a falta de compatibilidade entre plataformas e dispositivos agrava as dificuldades de integração em ambientes industriais heterogêneos [40, 41, 43];
- **Integração com sistemas existentes:** a incorporação da RA em linhas de produção requer alterações nos fluxos de trabalho e calibração rigorosa dos elementos virtuais. A instabilidade do *tracking* em ambientes com iluminação insuficiente ou superfícies complexas compromete frequentemente a fiabilidade das soluções [41];
- **Desempenho em tempo real e comunicação:** as aplicações industriais exigem latência reduzida para garantir interações fluídas. No entanto, redes convencionais nem sempre respondem a estas exigências. O recurso a tecnologias emergentes como 5G e *Mobile Edge Computing* (MEC) é apontado como essencial para suportar processamento intensivo e comunicação rápida em larga escala [44];
- **Privacidade e segurança:** o tratamento de dados sensíveis, incluindo informação operacional e biométrica, representa um risco crítico. Medidas de encriptação, anonimização e transparência na gestão de dados são fundamentais para assegurar a confiança dos utilizadores e o cumprimento regulamentar [42, 45].

Em síntese, embora a RA revele elevado potencial em contextos industriais, a sua adoção generalizada depende de avanços técnicos, normalização de práticas e soluções que conciliem desempenho, usabilidade e segurança.

2.2 Trabalhos Relacionados

A RA tem sido explorada em múltiplos setores, cada um com necessidades específicas e desafios comuns que ajudam a compreender os limites e oportunidades desta tecnologia. Nesta secção analisam-se exemplos representativos em quatro áreas distintas: medicina, turismo, arquitetura e construção e indústria.

2.2.1 Medicina

A RA tem ganho relevância no setor da saúde, sobretudo pela sua capacidade de melhorar a precisão das intervenções e apoiar os profissionais com informação clínica diretamente no campo de visão. Contudo, muitos projetos permanecem ainda em fase experimental, com integração limitada em contextos clínicos reais. Um exemplo é o sistema proposto

por Wen et al. [46], que criou um campo cirúrgico aumentado através da projeção de informação diretamente sobre o corpo do paciente (Figura 2.7a). A arquitetura do sistema (Figura 2.7b) integrou um robô móvel, câmaras estéreo, sensor de profundidade e um centro computacional responsável por coordenar o controlo robótico e gerar a *interface* aumentada. Esta acrescentava ao corpo do paciente a visualização de órgãos, vasos e trajetórias de inserção. A interação era feita por gestos manuais do cirurgião, reconhecidos em tempo real, e complementada por mecanismos de controlo por estados e autenticação biométrica através do reconhecimento da palma da mão.

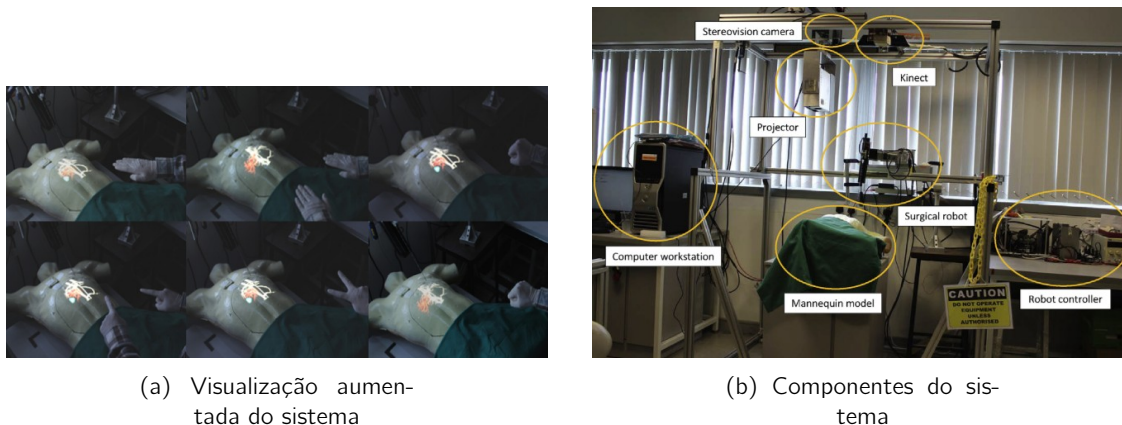


Figura 2.7: Sistema desenvolvido por Wen et al. (Adaptada de [46])

Focado no planeamento neurocirúrgico, Fick et al. [47] exploraram a utilização de modelos 3D gerados a partir de imagens de ressonâncias magnéticas. Após segmentação e otimização, os modelos eram visualizados em dispositivos como o *Microsoft HoloLens*, oferecendo uma perceção espacial imersiva para apoio à preparação das cirurgias (Figura 2.8). A arquitetura do sistema seguiu um fluxo automatizado, desde a aquisição das imagens à visualização aumentada, recorrendo a armazenamento na nuvem e segmentação automática antes da conversão para formatos compatíveis com múltiplas plataformas.

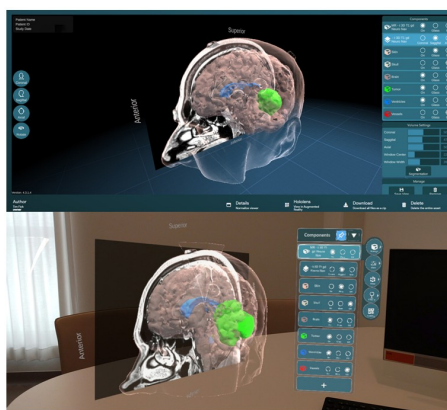


Figura 2.8: Sistema desenvolvido por Fick et al. numa plataforma web (em cima) e num dispositivo de RA (em baixo) (Adaptada de [47])

De forma mais orientada à prática clínica, Ivanov et al. [48] apresentaram uma aplicação de

RA para cirurgia abdominal oncológica, recorrendo ao *Microsoft HoloLens 2* como dispositivo de visualização intraoperatória. O sistema integrou imagens de tomografia convertidas em modelos 3D, exportados para *Unity* e projetados sobre o corpo do paciente (Figura 2.9). O registo espacial foi assegurado com marcadores óticos, que podiam ser invasivos ou não, e a interação foi feita por gestos manuais reconhecidos pelo dispositivo, permitindo manipular os modelos em tempo real.

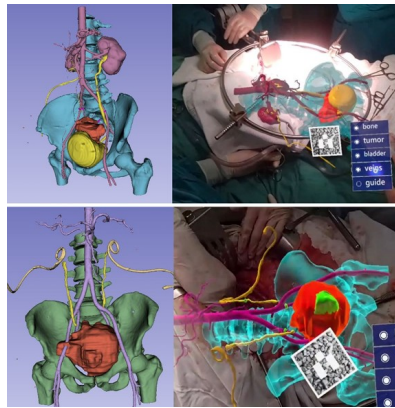


Figura 2.9: Sistema desenvolvido por Ivanov et al. (Adaptada de [48])

Al Janabi et al. [49] exploraram o uso do *Microsoft HoloLens* em procedimentos de ureteroscopia simulada. A arquitetura assentou na transmissão em tempo real das imagens do endoscópio diretamente para o visor do dispositivo, permitindo ao cirurgião manter uma linha de visão contínua com o campo operatório (Figura 2.10). A interação foi feita através de gestos simples e comandos básicos de navegação, e os resultados evidenciaram uma adaptação rápida por parte dos participantes, mesmo sem experiência prévia em RA.

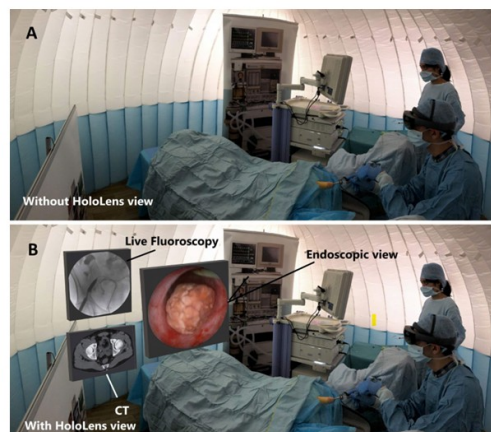


Figura 2.10: Sistema desenvolvido por Al Janabi et al. (Adaptada de [49])

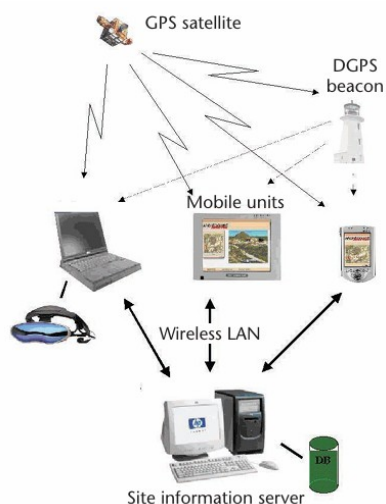
No conjunto, estes trabalhos mostram como a RA tem sido aplicada na medicina, sobretudo para melhorar a perceção espacial, apoiar o planeamento cirúrgico e guiar intervenções. As abordagens variam desde projeção direta no corpo à utilização de modelos 3D imersivos, refletindo tanto a diversidade de soluções técnicas como o carácter ainda exploratório da tecnologia no setor.

2.2.2 Turismo

No turismo, a RA foi usada para contextualizar conteúdos históricos, culturais e geográficos, aumentando a imersão das visitas. O projeto Archeoguide [50] foi pensado para sítios arqueológicos como Olímpia. O sistema recorreu a óculos *Sony Glasstron* ligados a sensores de GPS e magnéticos (Figura 2.11b), que permitiam sobrepor reconstruções digitais às ruínas reais (Figura 2.11a). A interação foi feita através de controladores físicos, como *gamepads* ou *interfaces* táteis, usados para navegar e aceder a conteúdos multimédia. Os testes mostraram o potencial da tecnologia para contextualização histórica, mas também revelaram limitações como o peso do equipamento e a dependência de uma infraestrutura de rede estável, que reduziam a fluidez da experiência.



(a) Paisagem real (à esquerda) e reconstrução digital sobreposta das ruínas (à direita)



(b) Arquitetura funcional do sistema Archeoguide

Figura 2.11: Sistema Archeoguide desenvolvido por Vlahakis et al. (Adaptada de [50])

O projeto PRISMA [51] introduziu uma abordagem diferente com binóculos aumentados instalados em miradouros de San Sebastián. Estes dispositivos tinham uma câmara embutida e sensores inerciais, permitindo captar a paisagem real e sobrepor-lhe conteúdos multimédia. A interação foi feita através da direção do olhar, ao alinhar os binóculos com ícones visíveis, o utilizador selecionava conteúdos adicionais, como vídeos ou reconstruções históricas. A implementação recorreu ao *ARToolKit*, adaptado para usar sensores inerciais em vez de marcadores visuais. Embora a experiência fosse mais natural e acessível a utilizadores sem

formação técnica, a precisão era limitada e a interação restrita à seleção passiva, sem maior liberdade de exploração. Williams et al. [52] desenvolveram a aplicação *ToARist* para *smartphones Android*, focada na navegação urbana. O sistema recorreu apenas aos sensores do dispositivo, dispensando infraestrutura externa. Os utilizadores visualizaram em tempo real pontos de interesse (POI) sobrepostos à imagem da câmara, interagindo diretamente no ecrã tátil para obter informações adicionais ou alternar entre vista de mapa e vista aumentada (Figura 2.12). A aplicação revelou-se intuitiva e útil, mas limitada por problemas técnicos como a sobreposição excessiva de ícones, que reduzia a clareza visual em locais com muitos POI.

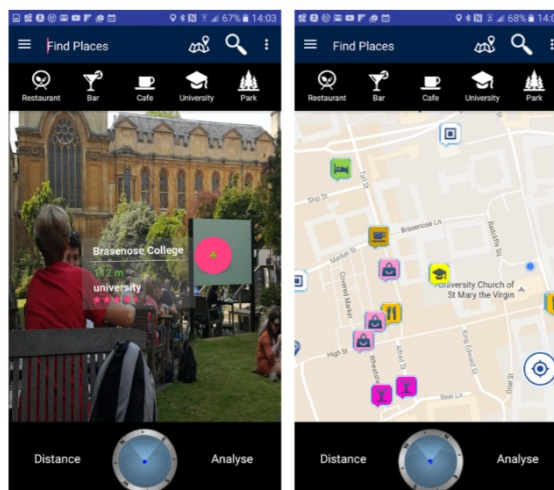


Figura 2.12: Sistema ToARist desenvolvido por Williams et al. (Adaptada de [52])

Estes exemplos mostraram diferentes formas de interação, desde controlos físicos a *interfaces* táteis e até métodos passivos como a direção do olhar. Os principais desafios recaíram sobre a ergonomia, a precisão do *tracking* e a sobrecarga visual, revelando a dificuldade em equilibrar acessibilidade e fiabilidade técnica.

2.2.3 Arquitetura e Construção

Nos setores da arquitetura e construção, a RA apoia a visualização de projetos e a deteção de erros em obras. Chen et al. [53] exploraram a RA na manutenção de edifícios com integração de modelos *Building Information Modeling* (BIM) (Figura 2.13). A interação variava consoante o dispositivo, em HMDs usava-se voz e gestos, enquanto em *tablets/smartphones* a interação era feita por toque no ecrã. O sistema incluía ainda instruções animadas e partilha remota em tempo real, promovendo colaboração entre equipas. Entre as limitações registaram-se problemas de localização em interiores e necessidade de rede estável para funcionamento fluido.

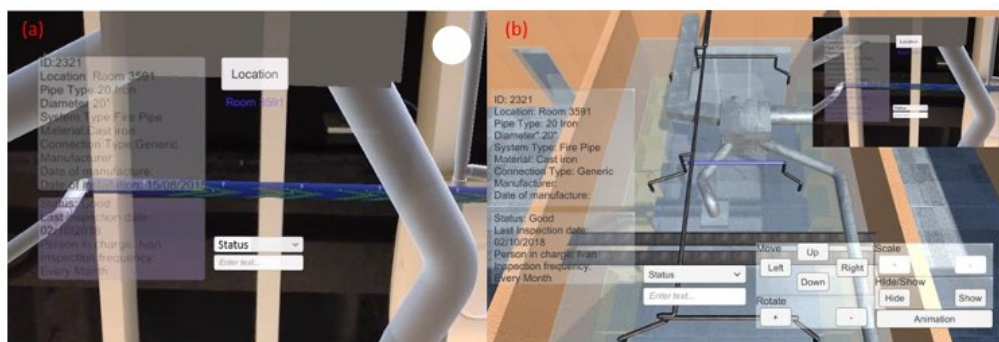


Figura 2.13: Sistema desenvolvido por Chen et al., com *interface* aumentada no local (à esquerda) e remota (à direita) (Adaptada de [53])

Pan e Isnaeni [54] focaram-se em inspeções visuais no local, integrando RA com modelos BIM através de *Unity* + *Vuforia* (Figura 2.14). A correspondência entre o modelo e a realidade foi garantida por múltiplos marcadores físicos posicionados no terreno. A interação foi feita via toque no ecrã do *smartphone*, permitindo comparar diretamente elementos reais e digitais. Os testes revelaram uma precisão de alinhamento de 98,5%, mas a abordagem ficou limitada em termos de escalabilidade devido à necessidade de múltiplos marcadores físicos e à dependência da iluminação do ambiente.

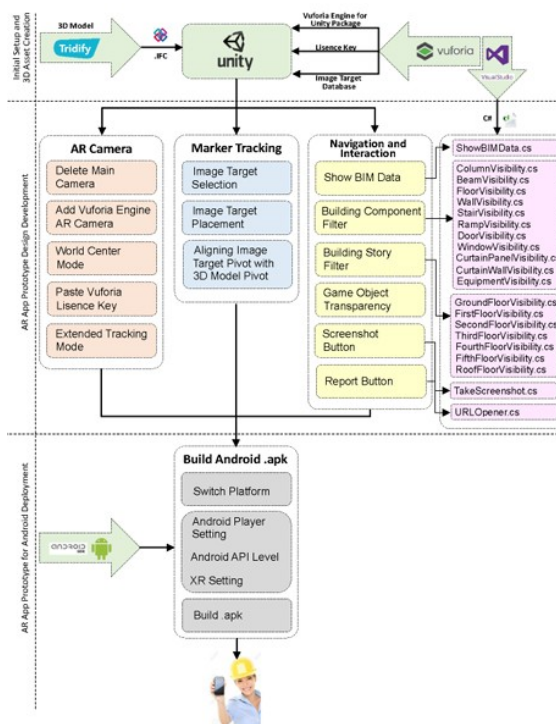


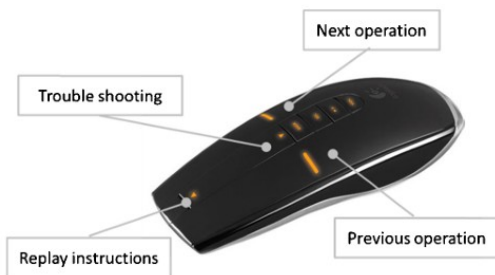
Figura 2.14: Arquitetura da aplicação desenvolvida por Pan e Isnaeni (Adaptada de [54])

Gerger et al. [55] aplicaram RA ao licenciamento urbano no projeto BRISE-Vienna. A interação foi híbrida, com gestos e toque em *tablets*, bem como visualização colaborativa multiutilizador em *Microsoft HoloLens*. O sistema mostrou ganhos na comunicação entre

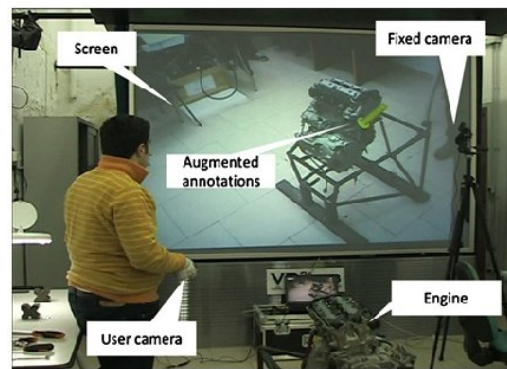
especialistas e público não técnico, mas persistiram problemas de instabilidade do rastreamento e desconforto ergonómico no uso prolongado de HMDs. Os resultados evidenciaram maior precisão e melhor comunicação entre utilizadores, mas também revelaram limitações no *tracking* e no conforto durante períodos prolongados de utilização.

2.2.4 Indústria

No setor industrial, a RA tem sido aplicada em diferentes fases do processo produtivo, nomeadamente em operações de montagem, manutenção, inspeção e controlo de qualidade. O objetivo comum passa por reduzir erros, aumentar a eficiência operacional e apoiar os operadores através da sobreposição de instruções digitais diretamente no ambiente físico. Fiorentino et al. [56] propuseram um sistema de manutenção com projeção em grandes ecrãs (Figura 2.15b), evitando o uso de *headsets* por questões ergonómicas. A interação foi feita com um rato sem fios adaptado (Figura 2.15a), que controlava a navegação entre instruções 3D projetadas. Apesar dos bons resultados em tempo e redução de erros, a solução exigia uma infraestrutura fixa complexa (câmaras + projetores).



(a) Rato sem fios adaptado para interação



(b) Componentes da arquitetura do sistema

Figura 2.15: Sistema desenvolvido por Fiorentino et al. (Adaptada de [56])

Marino et al. [57] desenvolveram uma aplicação em *tablets*, baseada em marcadores fiduciais (Figura 2.16). A interação era feita diretamente no ecrã tátil, permitindo acompanhar instruções de montagem passo a passo. A arquitetura cliente-servidor facilitava a atualização de conteúdos, mas surgiram problemas com falhas no reconhecimento de marcadores em condições de iluminação inadequada e desalinhamentos visuais ocasionais.

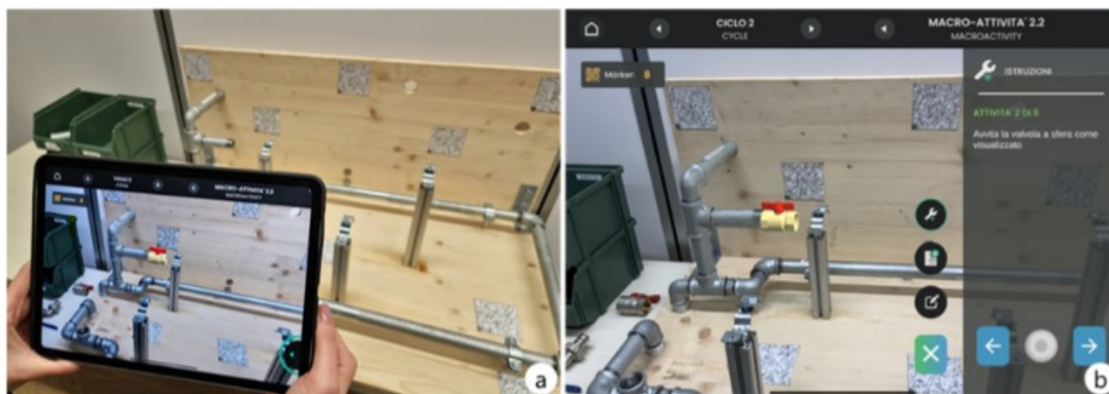


Figura 2.16: Sistema desenvolvido por Marino et al. (Adaptada de [57])

Seeliger et al. [2] aplicaram RA ao controlo de qualidade, sobrepondo diretamente no produto as falhas detetadas por visão computacional (Figura 2.17). O operador usava um HMD, observando as anomalias projetadas sem necessidade de comandos adicionais (interação passiva). Apesar dos ganhos em consistência e rapidez, a calibração entre projeções virtuais e superfícies físicas mostrou-se sensível a variações ambientais.



Figura 2.17: Sistema desenvolvido por Seeliger et al. (Adaptada de [2])

Alves et al. [58] desenvolveram uma solução móvel em *Unity*, com câmara RGB-D num tripé. A interação era mínima, uma vez que o sistema avançava automaticamente para a instrução seguinte quando detetava a posição correta. Esta abordagem reduzia carga cognitiva, mas dependia de recalibrações frequentes devido a vibrações no chão fabril. (Figura 2.18).

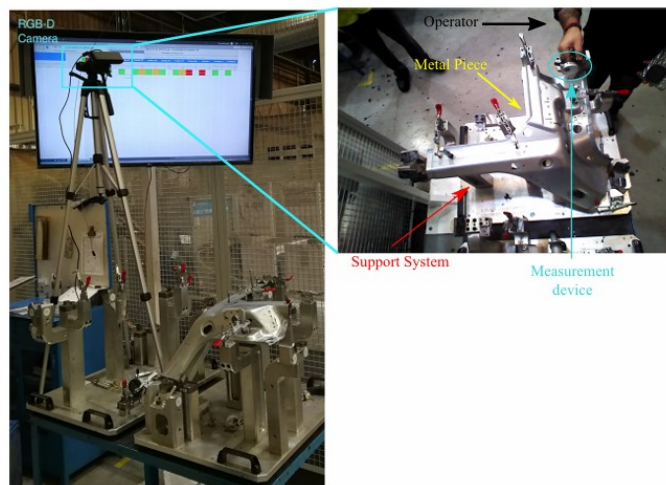


Figura 2.18: Infraestrutura proposta por Alves et al. (Adaptada de [58])

Szajna et al. [59] apresentaram um sistema industrial com óculos *Microsoft HoloLens*, integrando sensores digitais via *Azure IoT Hub* (Figura 2.19). A interação combinava gestos e comandos de voz, permitindo ao operador trabalhar com as mãos livres. Os testes mostraram ganhos no *tracking* e no tempo de execução, mas surgiram limitações como o campo de visão restrito do *Microsoft HoloLens* e o desconforto em uso prolongado.

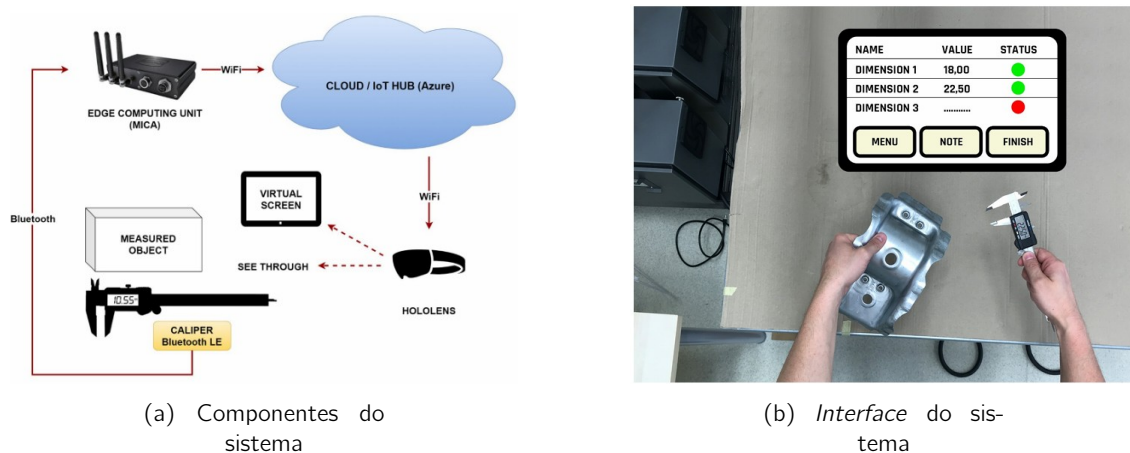


Figura 2.19: Sistema desenvolvido por Szajna et al. (Adaptada de [59])

Estes trabalhos demonstram o contributo da RA para reduzir erros e apoiar operadores em tarefas complexas. A diversidade de soluções, desde projeção em ecrãs a *tablets* e HMDs, revela diferentes estratégias de implementação consoante os requisitos ergonómicos e técnicos. O presente trabalho enquadra-se no domínio da inspeção visual e do controlo de qualidade, uma vertente que, apesar do seu elevado potencial industrial, continua a ser menos explorada do que restantes aplicações de RA.

2.2.5 Síntese Comparativa

A Tabela 2.1 sintetiza os trabalhos analisados, destacando as arquiteturas, modalidades de interação e principais limitações. Observa-se que, apesar da diversidade de setores, os

desafios convergem: ergonomia dos dispositivos, robustez do *tracking* e integração fluida em ambientes reais.

Tabela 2.1: Síntese comparativa dos trabalhos relacionados

Artigo	Área	Game engine	Dispositivo	Interação	Limitações
Wen et al. [46]	Medicina	—	Projeção direta	Gestos manuais	Sem validação clínica, <i>hardware</i> complexo
Fick et al. [47]	Medicina	—	Microsoft HoloLens	Visualização imersiva	Apenas planeamento, não em cirurgia
Ivanov et al. [48]	Medicina	Unity	Microsoft HoloLens 2	Gestos manuais	Conforto, alguns métodos invasivos
Al Janabi et al. [49]	Medicina	—	Microsoft HoloLens	Gestos simples	Apenas simulação, ergonomia limitada
Archeoguide [50]	Turismo	—	Sony Glasstron	Gamepad, tátil	Hardware pesado, dependência de rede
PRISMA [51]	Turismo	—	Binóculos fixos	Direção do olhar	Precisão limitada, interação restrita
Williams et al. [52]	Turismo	—	Smartphone	Toque no ecrã	Interferências, sobreposição de ícones
Chen et al. [53]	Arquitetura e Construção	Unity	HMDs, tablets, PCs	Voz, gestos, toque	Tracking limitado, rede necessária
Pan & Isnaeni [54]	Arquitetura e Construção	Unity + Vuforia	Smartphone	Toque no ecrã	Marcadores físicos, escalabilidade
Gerger et al. [55]	Arquitetura e Construção	—	Microsoft HoloLens, tablets	Gestos, <i>multiuser</i>	Tracking instável, ergonomia
Fiorentino et al. [56]	Indústria	—	Projetores + câmaras	Rato sem fios adaptado	Hardware fixo, sem HMD
Marino et al. [57]	Indústria	Unity	Tablet	Toque no ecrã	Falhas em marcadores, desalinhamento
Seeliger et al. [2]	Indústria	Unity	Microsoft HoloLens	Passiva (visual)	Calibração sensível, ergonomia
Alves et al. [58]	Indústria	Unity	Tripé com portátil	Passiva (automática)	Vibração, necessidade de recalibração
Szajna et al. [59]	Indústria	—	Microsoft HoloLens	Voz, gestos	Campo de visão limitado, desconforto

De forma geral, os trabalhos analisados mostram que a RA tem vindo a ser aplicada em múltiplos setores com objetivos distintos, mas partilhando uma mesma ambição: aproximar o utilizador da informação digital de forma contextual e intuitiva. Seja na medicina, no turismo, na construção ou na indústria, a RA procura apoiar a perceção, simplificar processos complexos e melhorar a comunicação entre pessoas e sistemas. Apesar da diversidade de abordagens e ferramentas utilizadas, destacam-se tendências comuns: a aposta em dispositivos móveis e óculos de RA, a procura por interações naturais (gestos, voz, toque) e a integração com dados digitais já existentes (modelos 3D, BIM, informação clínica). Em contrapartida, surgem também desafios transversais, como o desconforto associado ao uso prolongado de HMDs, a instabilidade do *tracking* em ambientes reais e a dificuldade em garantir uma integração fluida sem depender de *hardware* ou infraestrutura complexa. Este panorama evidencia que a RA é uma tecnologia promissora, mas ainda em consolidação. O seu valor prático depende não apenas da inovação técnica, mas também da capacidade de adaptação às condições reais de utilização e às necessidades concretas dos utilizadores finais.

Capítulo 3

Desenho

Este capítulo apresenta a especificação do protótipo desenvolvido. É introduzido um fluxo global que sintetiza a utilização do sistema, desde a configuração inicial até ao registo das decisões. De seguida são descritos os requisitos funcionais e não funcionais, o esquema dos dados recebidos pela aplicação e os fluxos de decisão que orientam o seu funcionamento.

3.1 Cenário Narrativo de Utilização

Numa linha de produção da Neutroplast, o operador Tomás inicia o seu turno dedicado à inspeção de qualidade das embalagens plásticas farmacêuticas. Coloca os óculos Meta Quest 3S e abre a aplicação, onde tem acesso a um painel de configuração inicial. Nesse painel define as suas preferências quanto ao tipo e forma de alerta e quanto ao modo como deseja realizar a inspeção. Tem ainda a possibilidade de ajustar a transparência da interface, adaptando-a à luminosidade do ambiente e ao seu conforto visual. Concluída a configuração, o sistema permanece ativo enquanto o operador desempenha outras tarefas, até que é emitido o alerta configurado, acompanhado da opção "Begin Analysis" no visor. Ao selecionar essa opção, a embalagem correspondente é apresentada com as zonas suspeitas destacadas através de sobreposições visuais. O Tomás observa e interage com a aplicação de forma simples, confirmando ou rejeitando cada sugestão de defeito, podendo, sempre que necessário, ampliar ou afastar a imagem para inspecionar com maior precisão. No final do turno, consulta o histórico de inspeções realizadas, revendo imagens e decisões anteriores.

Fluxo Global de Utilização

A Figura 3.1 apresenta o fluxo global do protótipo, desde a configuração inicial até ao registo das inspeções realizadas. Este diagrama sintetiza os diferentes elementos como, configuração de alertas e modos de inspeção, receção de dados, emissão de notificações e validação de defeitos, permitindo compreender de forma integrada o percurso seguido pelo operador ao longo do seu turno.

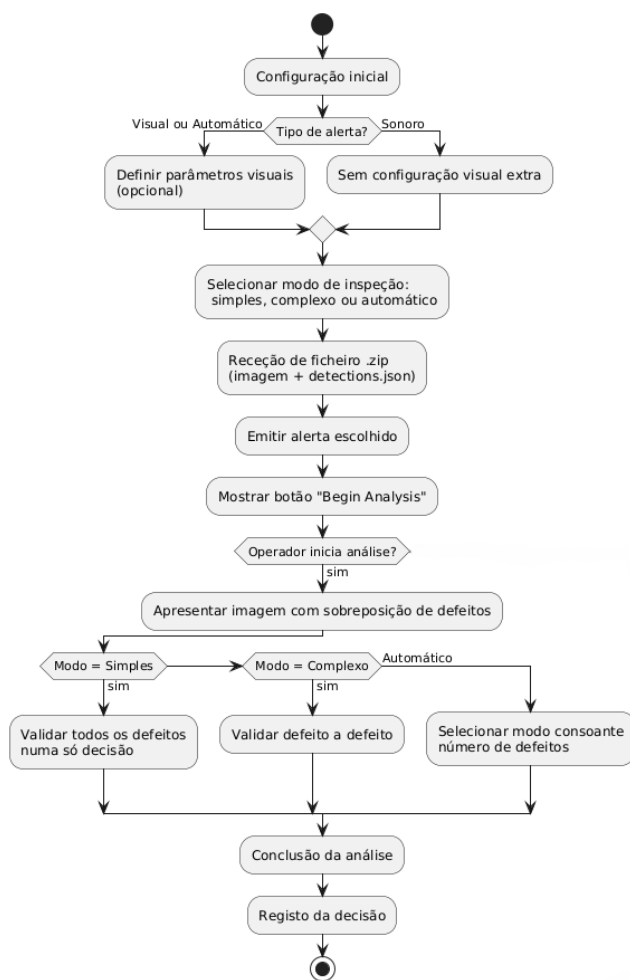


Figura 3.1: Fluxo global de utilização do protótipo

3.2 Requisitos Funcionais

O protótipo desenvolvido responde a um conjunto de requisitos que refletem as funcionalidades exploradas:

- **Configuração inicial:** permite ao operador definir preferências gerais de utilização, incluindo:
 - escolha do modo de alerta (visual, sonoro ou automático em função do ruído ambiente),
 - escolha do modo de inspeção (simples, complexo ou automático) e
 - ajuste de parâmetros visuais, como o formato do alerta visual, a transparência da *interface* e o nível de ampliação das embalagens;
- **Receção de dados:** obtém em tempo real imagens das embalagens, acompanhadas da indicação de potenciais defeitos a validar;
- **Visualização aumentada:** apresenta a embalagem em modo *passthrough*, sobrepondo elementos gráficos que destacam as regiões suspeitas de defeito;

- **Interação e validação:** possibilita ao operador confirmar ou rejeitar cada sugestão de defeito, recorrendo a modalidades de interação distintas, como gestos manuais ou controladores físicos;
- **Registo de feedback:** as decisões do operador são armazenadas, permitindo o seu aproveitamento em futuras evoluções da aplicação;

3.3 Requisitos Não Funcionais

Além das funcionalidades principais, foram considerados requisitos de caráter transversal que asseguram a experiência de utilização:

- **Imersão e mobilidade:** utilização de óculos de RA que proporcionam liberdade de movimentos e apresentam a informação diretamente no campo de visão do operador;
- **Baixa carga cognitiva:** apresentação clara e em tempo útil da informação, direcionando a atenção do operador para zonas críticas sem sobrecarregar a percepção e apoiando a tomada de decisão;
- **Prototipagem rápida:** desenvolvimento orientado para testar hipóteses e validar conceitos, com foco na exploração da tecnologia e não na criação de uma solução final.

3.4 Esquema dos Dados Recebidos

Para que o protótipo consiga apresentar de forma contextual a informação necessária ao operador, é essencial que os dados recebidos incluam um conjunto mínimo de elementos:

- **Identificação da embalagem:** um identificador único que permita associar os resultados da inspeção a uma embalagem concreta;
- **Lista de defeitos detetados:** cada defeito deve incluir informação suficiente para permitir a sua visualização e validação, nomeadamente:
 - **Tipo de defeito**, indicando o tipo de anomalia;
 - **Localização**, expressa de forma relativa à imagem da embalagem, que permita sobrepor uma marcação visual no local correto;
 - **Grau de confiança**, indicando a probabilidade do defeito corresponder a uma anomalia real.
- **Imagem associada:** fotografia da embalagem a inspecionar, sobre a qual as anotações são aplicadas.

Este conjunto de informações constitui o mínimo necessário para que a aplicação apresente corretamente os potenciais defeitos sobre a embalagem, fornecendo ao operador os elementos necessários para efetuar a validação.

3.5 Fluxos de Decisão

O funcionamento do protótipo pode ser descrito através dos fluxos de decisão que orientam a sua lógica. Estes fluxos representam as regras centrais que guiam o sistema, desde a forma como os alertas são emitidos até ao modo como é conduzida a inspeção das embalagens.

3.5.1 Emissão de Alerta

O sistema emite uma notificação sempre que existe uma embalagem a validar. O tipo de alerta depende da configuração inicial escolhida pelo operador, que pode optar por visual, sonoro ou, opcionalmente, automático:

- Visual: aciona um efeito gráfico no visor;
- Audio: emite um sinal de áudio;
- Auto: avalia o nível de ruído ambiente para escolher entre visual ou sonoro.

No modo Auto, se a leitura for válida e o ruído for superior ao limiar definido, considera-se o ambiente demasiado ruidoso e emite-se um alerta visual. Se a leitura for válida e o ruído estiver abaixo desse limiar, emite-se um alerta sonoro. Já se a leitura falhar, o sistema recorre à preferência base definida pelo operador (*Visual* ou *Audio*). A Figura 3.2 apresenta o esquema correspondente a este fluxo de decisão.

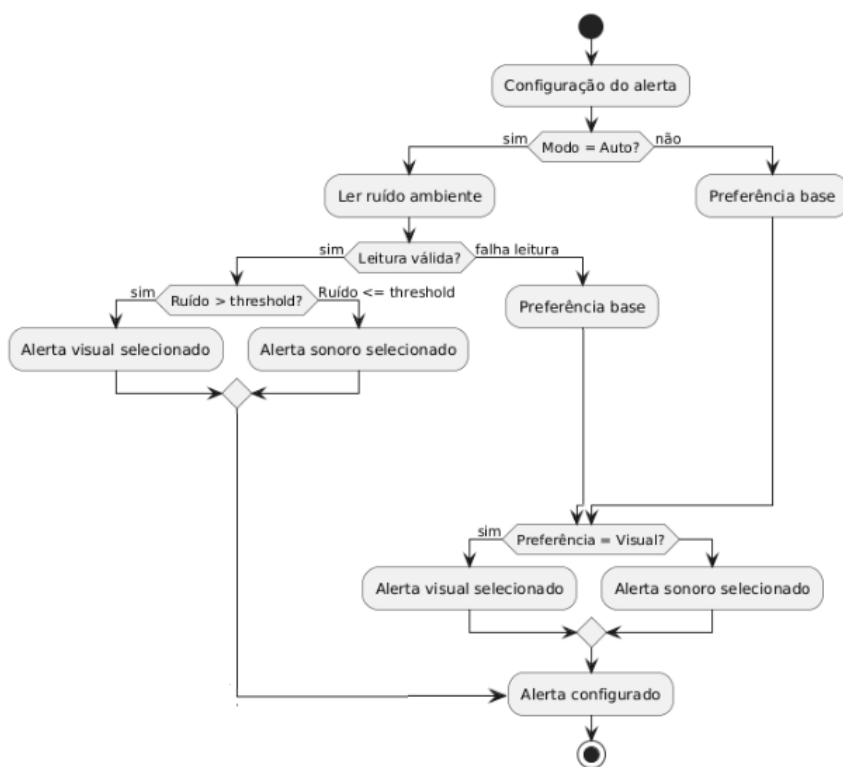


Figura 3.2: Árvore de decisão para emissão de alertas

3.5.2 Modos de Inspeção

A validação das embalagens pode ser realizada segundo três modalidades distintas, simples, complexa ou automática:

- Simple Mode: todos os defeitos são apresentados em simultâneo, permitindo ao operador tomar uma decisão única relativamente à embalagem;
- Complex Mode: cada defeito de uma embalagem é apresentado individualmente, solicitando ao operador uma decisão específica para cada caso;

- Auto Mode: seleciona automaticamente entre os dois modos anteriores em função do número de defeitos detetados:
 - se o número de defeitos for igual ou superior ao limiar definido, aplica-se o Simple Mode,
 - se o número de defeitos for inferior, aplica-se o Complex Mode.

A Figura 3.3 ilustra este fluxo de decisão.

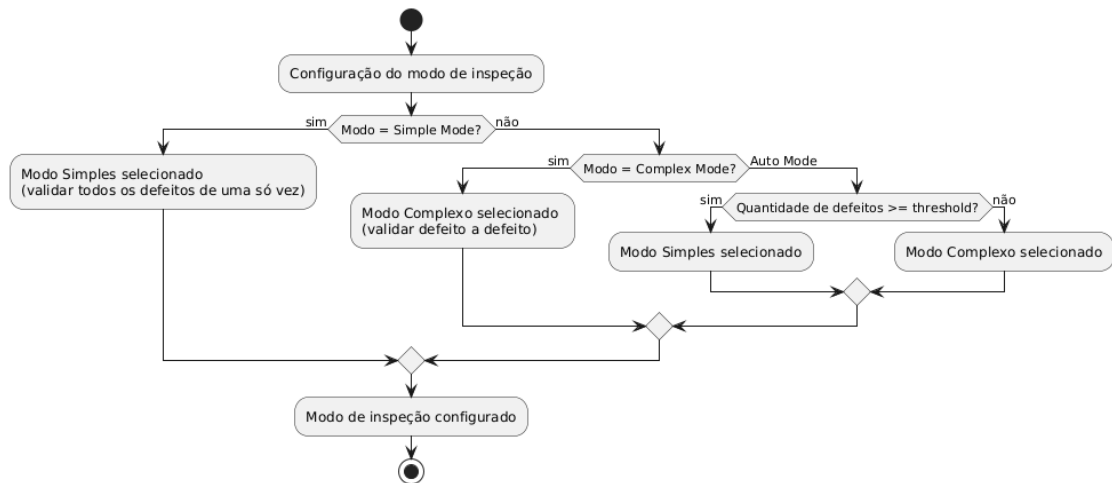


Figura 3.3: Árvore de decisão para os modos de inspeção

O conjunto de elementos apresentados anteriormente permite enquadrar o protótipo como uma prova de conceito destinada a explorar o potencial da RA no contexto industrial. Não se trata de uma solução final, mas de uma base experimental que demonstra como diferentes modalidades de alerta e de inspeção podem apoiar o operador humano no processo de controlo de qualidade.

Capítulo 4

Implementação

Neste capítulo é apresentada a implementação do protótipo desenvolvido, descrevendo as principais decisões técnicas e funcionais. São introduzidas a arquitetura geral e as ferramentas utilizadas, seguidas da caracterização dos módulos centrais do sistema: gestão de alertas, modos de inspeção, métodos de interação e opções de configuração e personalização da *interface*.

4.1 Arquitetura Geral

A arquitetura do protótipo foi concebida de forma modular, permitindo isolar responsabilidades e facilitar a evolução futura do sistema. A Figura 4.1 apresenta uma visão global desde a receção dos dados até à interação final com o operador.

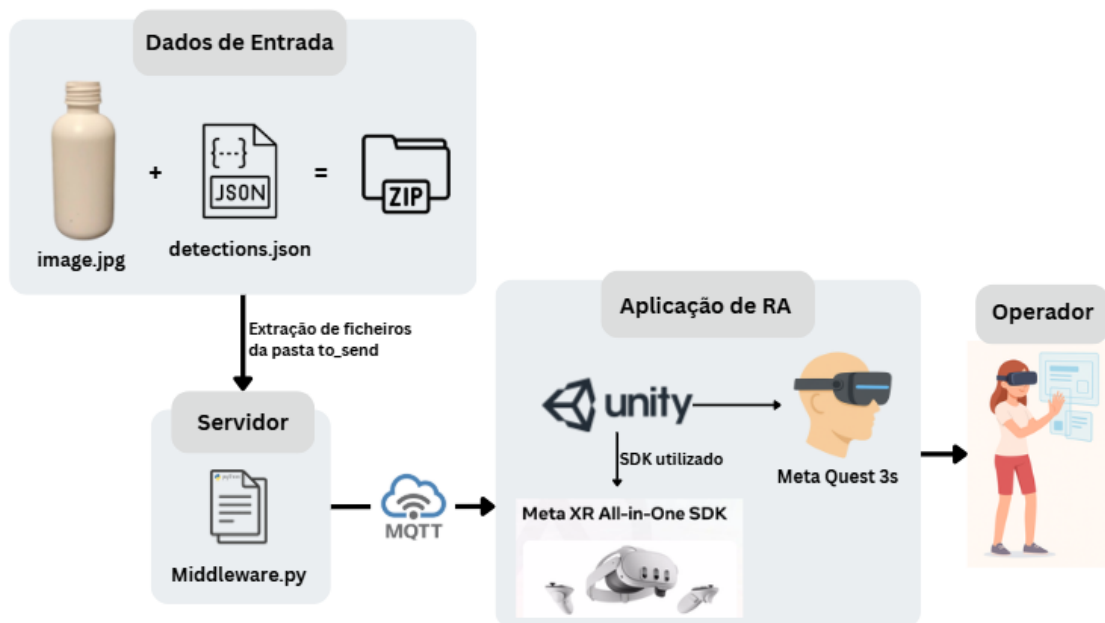


Figura 4.1: Arquitetura geral do sistema

- **Dados de entrada:** cada notificação chega sob a forma de um ficheiro .zip contendo a imagem da embalagem e o respetivo ficheiro detections.json, com a descrição das anomalias possíveis detetadas;

- **Servidor:** o ficheiro `middleware.py` é responsável por preparar e disponibilizar os ficheiros `.zip` através do protocolo MQTT, garantindo que a aplicação de RA os recebe no momento adequado;
- **Aplicação de RA:** desenvolvida em *Unity* e executada nos óculos *Meta Quest 3S*, incorpora os módulos de receção de dados, configuração da sessão de validação dos defeitos com uma configuração inicial, emissão de alertas visuais e sonoros, personalização da *interface*, e suporte a diferentes modos de inspeção de defeitos e interação, apresentando a informação de forma sobreposta no campo de visão do operador;
- **Operador:** interage com os óculos *Meta Quest 3S* para configurar a sessão de validação, receber notificações e validar os defeitos.

4.2 Dados de entrada

Os dados utilizados no protótipo consistem em pares formados por uma imagem da embalagem e um ficheiro `detections.json`, no qual se encontram registadas as coordenadas e etiquetas dos defeitos detetados. Para simplificar o transporte e garantir que a imagem e as respetivas anotações chegam em conjunto à aplicação de RA, estes elementos são agrupados num ficheiro `.zip`. No contexto deste trabalho, a criação destes pacotes foi realizada manualmente, recorrendo a imagens recolhidas e a anotações controladas, de forma a simular o funcionamento de um sistema automatizado de deteção de defeitos.

O ficheiro JSON segue uma estrutura simples, organizada em três elementos principais, como podemos ver pelo Excerto 4.1:

- Identificação da imagem (`filename`);
- Lista de defeitos (`defects`), em que cada defeito é descrito por:
 - posição (`x`, `y`) relativa à imagem,
 - dimensões (`width`, `height`) da área a destacar,
 - grau de confiança (`confidence`),
 - tipo de defeito (`type`).
- Linha de produção (`linha_producao`), que contextualiza a origem da embalagem.

```
1 {
2   filename: string ,
3   defects: [
4     {
5       x: float ,
6       y: float ,
7       width: float ,
8       height: float ,
9       confidence: float ,
10      type: string
11    },
12    ...
13  ],
14  linha_producao: string
15 }
```

Excerto 4.1: Ficheiro `detections.json`

Para as imagens, surgiu a necessidade de criar um conjunto de referência que permitisse simular um fluxo real de detecção de defeitos em embalagens plásticas. Assim sendo, foi recolhido material diretamente no chão de fábrica da empresa Neutroplast, concebido como ferramenta de apoio para assegurar consistência nos testes e aproximar a validação do protótipo a um ambiente industrial. Embora com esta finalidade imediata, o conjunto foi estruturado de forma genérica, podendo futuramente ser explorado no treino de modelos de visão computacional. Neste conjunto de imagens foram incluídos três tipos de embalagens plásticas farmacêuticas distintas (Figura 4.2a), bem como quatro tipos de defeitos típicos identificados em contexto produtivo (Figura 4.2b), com base em informação disponibilizada pela empresa.

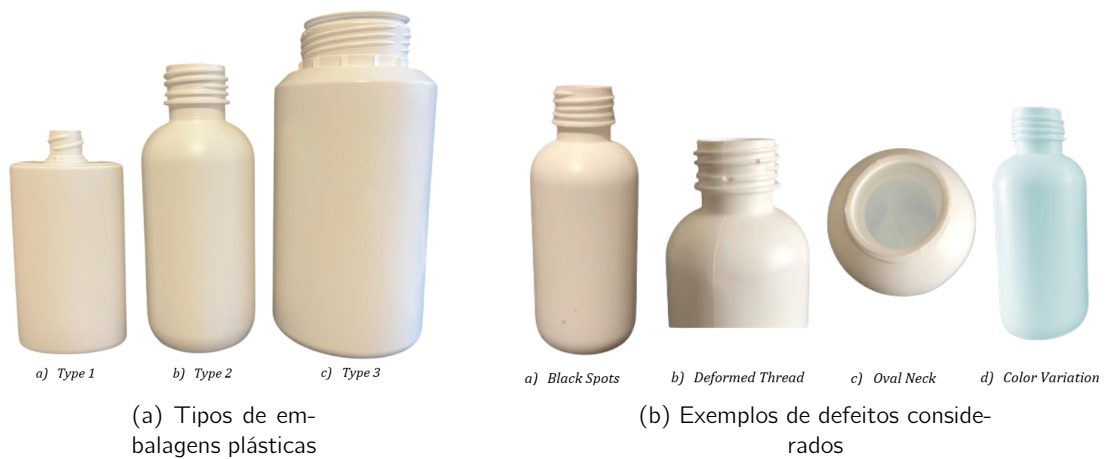


Figura 4.2: Conjunto de dados de referência

Na Figura 4.3 apresentam-se exemplos de imagens reais adquiridas, ilustrando de forma concreta os dados utilizados. É possível observar embalagens com os defeitos: *deformed thread* (imagens a e f), *oval neck* (imagens b e d), *color variation* (imagens e e h) e *black spot* (imagens c e j), bem como embalagens sem defeito (imagens g e i).

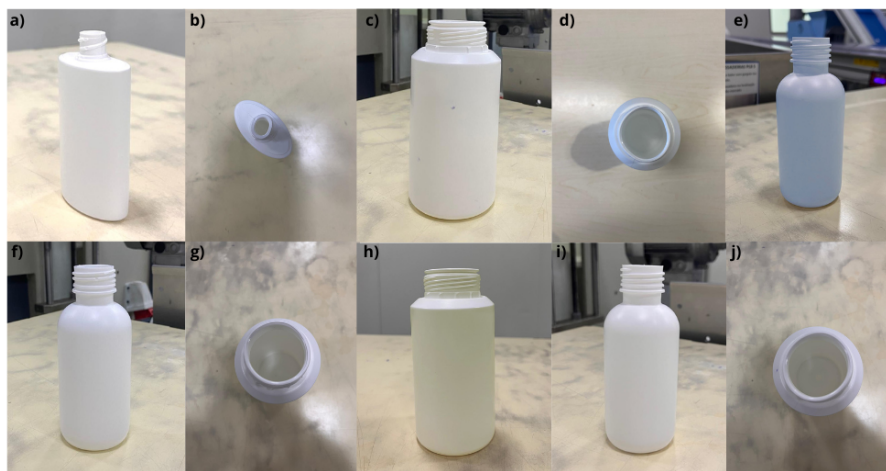


Figura 4.3: Exemplos de imagens adquiridas, identificadas de a) a j)

Complementarmente, foi ainda desenvolvido um *dataset* adicional destinado ao treino de modelos de visão computacional, como contributo para os objetivos globais do projeto *Sustainable Plastics*. Este conjunto de dados não foi recolhido em chão de fábrica, mas sim construído em contexto controlado para efeitos de investigação. Embora não seja utilizado no protótipo apresentado, encontra-se documentado no Anexo B como parte do trabalho realizado.

4.3 Servidor

O servidor do sistema foi implementado em *Python*, através do *script* `Middleware.py`. Este módulo funciona como ponto central entre os dados de entrada e a aplicação de RA, com ênfase em sincronização e fiabilidade em contexto de rede local. A decisão pelo protocolo *Message Queuing Telemetry Transport* (MQTT) resulta da sua leveza, baixo *overhead* e suporte a entrega em tempo útil mesmo em redes com variabilidade de latência [60]. No sistema desenvolvido, o *middleware* assume o papel de *publisher* e publica num *broker* central ficheiros `.zip`. Do lado do cliente, a aplicação *Unity* atua como *subscriber*. A integração foi feita com a biblioteca `M2MqttUnity`, uma adaptação da `M2Mqtt` concebida especificamente para este motor de desenvolvimento, tendo como referência o exemplo disponibilizado pela EMQX [61]. Toda a lógica de comunicação foi encapsulada na classe `MqttHandler`, garantindo a separação entre a gestão da ligação ao *broker* e as restantes funcionalidades da aplicação.

A sincronização entre servidor e cliente é assegurada através de um mecanismo simples: o *middleware* só inicia o envio de pacotes após receber do *Unity* a mensagem de prontidão `"ready"`, publicada no tópico `quest/ready` (Excerto 4.3). No código, esta condição é representada pela variável `unity_ready`, que passa a `true` assim que a confirmação é recebida. A partir deste momento, cada ZIP é publicado no tópico `defect/warning`, sob a forma de um objeto JSON contendo o `filename` e os dados em `base64`, conforme exemplificado no Excerto 4.2.

```
1 msg = {"filename": filename, "data": data_b64}
2 client.publish("defect/warning", json.dumps(msg), qos=1)
```

Excerto 4.2: `Middleware.py`: envio após `"ready"`

No cliente *Unity*, a classe `MqttHandler` gere a ligação ao *broker*, a subscrição e a receção de mensagens. O envio do sinal de prontidão `"ready"` só ocorre depois do utilizador confirmar as opções de configuração da sessão de validação no ecrã inicial. Para evitar repetições, o componente mantém dois estados internos, garantindo que a mensagem é publicada apenas uma vez por sessão, recorrendo ao nível de qualidade de serviço `QoS1` (*at least once*). Esta lógica assegura que o arranque decorre sem falhas, independentemente de a aplicação ou a rede ficarem prontas primeiro (Excerto 4.3).

```
1 _client.Publish("quest/ready",  
2     Encoding.UTF8.GetBytes("ready"),  
3     MqttMsgBase.QOS_LEVEL_AT_LEAST_ONCE, true);
```

Excerto 4.3: MqttHandler.cs: publicação do "ready"

A receção de cada ficheiro .zip decorre em duas etapas. Primeiro, o cliente grava o ficheiro em *persistentDataPath* e lança um processo assíncrono para evitar bloqueio do ciclo principal (Excerto 4.4).

```
1 string zipPath = Path.Combine(Application.persistentDataPath, filename);  
2 File.WriteAllBytes(zipPath, Convert.FromBase64String(base64Zip));  
3 StartCoroutine(ProcessZipImage(zipPath));
```

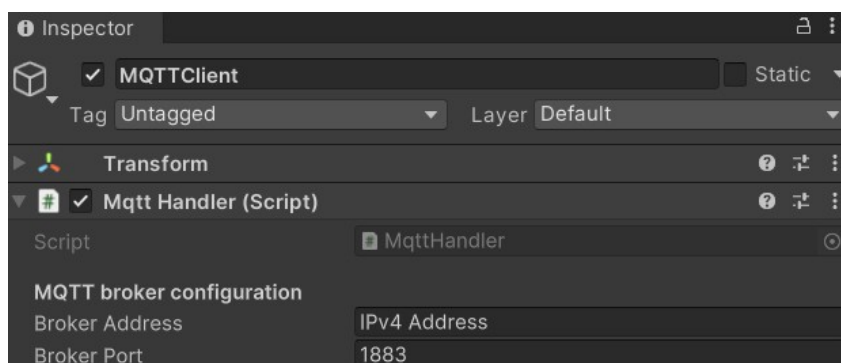
Excerto 4.4: MqttHandler.cs: escrita do ZIP, extração e encaminhamento

Na etapa seguinte, este processo cria uma pasta de trabalho com nome exclusivo, extrai o conteúdo, localiza o ficheiro *detections.json* e entrega o caminho ao componente de sobreposição, que atualiza a imagem e desenha as caixas correspondentes (Excerto 4.5). Todo o percurso fica registado no log: ligação, subscrição, escrita, extração e encaminhamento, o que facilita o diagnóstico em caso de falhas.

```
1 ZipFile.ExtractToDirectory(zipPath, extractPath);  
2 string jsonPath = Path.Combine(extractPath, "detections.json");  
3 overlayManager?.LoadDetectionsFromJson(jsonPath);
```

Excerto 4.5: MqttHandler.cs: extração do ZIP e carregamento das deteções

A parametrização concentra-se em dois pontos. No *Middleware.py*, os campos *BROKER* e *PORT* definem o endereço e a porta do *broker*. No *Unity*, os parâmetros *brokerIp* e *brokerPortValue* estão expostos no *Inspector*, apresentados como *Broker Address* e *Broker Port* (Figura 4.4).

Figura 4.4: Objeto MQTTClient no *Inspector* do *Unity*, parâmetros *brokerIp* e *brokerPortValue* visíveis para configuração.

4.4 Aplicação de Realidade Aumentada

A aplicação foi desenvolvida com recurso a ferramentas de suporte que garantem a sua execução e desenvolvimento e integra um conjunto de funcionalidades próprias que materializam o protótipo. Nos pontos seguintes apresentam-se, em primeiro lugar, as ferramentas utilizadas, depois a descrição dos objetos do *Unity* e da organização da cena e, por fim, a implementação da aplicação.

4.4.1 Ferramentas de suporte

Para o desenvolvimento e validação da aplicação de RA foi utilizado o dispositivo *Meta Quest 3S* (Figura 4.5), disponibilizado especificamente para este trabalho. Trata-se de um equipamento autónomo, que dispensa a ligação a um computador e garante maior mobilidade ao operador. As câmaras integradas permitem utilizar *passthrough* de vídeo a cores, funcionalidade ideal para cenários de RA, uma vez que possibilita projetar defeitos diretamente sobre as embalagens reais. Em complemento, o suporte a diferentes formas de interação permite validar defeitos de forma intuitiva. Estas características tornam o *Quest 3S* adequado a um cenário de exploração tecnológica em contexto industrial.



Figura 4.5: Óculos *Meta Quest 3S* (Adaptada de [62])

Como motor de desenvolvimento foi adotado o *Unity*, pela sua consolidação no mercado e pelo suporte nativo a aplicações de RA. A vasta comunidade de utilizadores e a documentação disponível simplificam a aprendizagem e a resolução de problemas, aspetos particularmente relevantes num projeto com prazos limitados. Outro fator determinante foi a capacidade de compilar aplicações diretamente para *Android*, requisito fundamental para garantir a execução no *Meta Quest 3S*. A integração nativa com os SDK dos fabricantes de *hardware* de RA assegura ainda compatibilidade e reduz significativamente o esforço de desenvolvimento.

A utilização do *Unity* implicou a definição de uma estrutura modular, organizada em *GameObjects* com responsabilidades bem definidas, o que assegura a separação de funções, facilita a manutenção e permite compreender facilmente o papel de cada componente na aplicação. Os elementos principais são:

- **SettingsController:** centraliza a lógica da configuração, regista as escolhas do utilizador relativas ao modo de alerta e inspeção, valida os passos do processo e notifica os restantes módulos de que a sessão está pronta;
- **PageController:** gere a navegação entre os diferentes painéis de configuração, assegurando a ordem correta dos passos, a validação antes de avançar, e a transição para o painel de validação após guardar as definições;

- **AlertVisual, NoiseDetector, Audio Source e AlertSettingsDisplay:** em conjunto, constituem o sistema de alertas. O `AlertVisual` apresenta o componente visual, o `NoiseDetector` e o `AudioSource` tratam da vertente sonora, e o `AlertSettingsDisplay` fornece ao operador *sliders* para ajustar os parâmetros de funcionamento do alerta visual;
- **InspectionController:** coordena o processo de inspeção, invocando o validador apropriado de acordo com o modo selecionado pelo utilizador;
- **DefectSimpleValidator e DefectComplexValidator:** implementam a lógica dos diferentes modos de inspeção, respetivamente global e defeito a defeito;
- **DefectOverlayController:** desenha as caixas de deteção sobre a imagem recebida, assegurando a integração entre dados e *interface*;
- **CanvasZoom e OpacityController:** permitem personalizar a *interface*, ajustando zoom e opacidade de acordo com as preferências do operador;
- **FollowHead:** mantém o painel de *interface* alinhado com a câmara do utilizador, a uma distância fixa à frente dos óculos, o que evita perda de visibilidade durante movimentos de cabeça.

A Figura 4.6 mostra a hierarquia da cena no *Unity* e a associação de cada *GameObject* ao *script* correspondente. Esta organização modular facilita a leitura da implementação, clarificando onde cada funcionalidade está localizada.

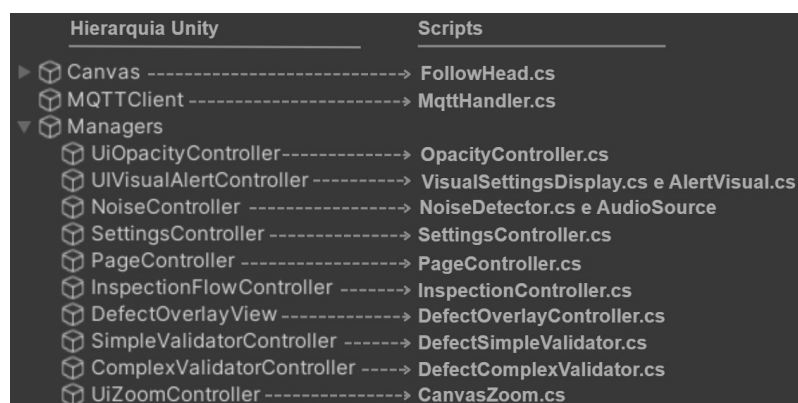


Figura 4.6: Relação entre os principais *GameObjects* da cena *Unity* e os respetivos *scripts* de implementação

4.4.2 Implementação da aplicação

Na fase inicial da implementação foi necessário garantir que o utilizador pudesse visualizar o ambiente real através dos óculos. Para isso foi utilizada a funcionalidade de *passthrough*, implementada através dos Building Blocks do Meta XR All-in-One SDK para *Unity*. Este pacote agrega os módulos principais do ecossistema Meta XR SDK, garantindo a instalação automática das versões mais recentes. Entre as funcionalidades disponibilizadas encontram-se o suporte nativo para *passthrough*, interação com mãos e controladores, *tracking* de cabeça e corpo, gestão de âncoras espaciais e integração com sistemas de entrada/saída do dispositivo. No protótipo foram adicionados dois blocos principais pelo menu `Meta > Tools > Building Blocks`:

- [BuildingBlock] Camera Rig, que fornece a estrutura base da aplicação em RA, com câmaras, âncoras e suporte de renderização;
- [BuildingBlock] Passthrough, que ativa de imediato o *passthrough* em tempo de execução, adicionando os componentes e *shaders* necessários à fusão entre mundo real e conteúdo digital.

Com esta configuração, ao iniciar a aplicação o operador visualiza de imediato o ambiente físico real através das câmaras a cores do *Meta Quest 3S*, sem necessitar de ajustes adicionais. Esta solução simplifica a implementação e adequa-se ao caráter exploratório do protótipo, mantendo o foco na experiência de inspeção e validação de defeitos. Desta forma, após garantida a base de RA, passou-se à implementação dos diferentes módulos que compõem a aplicação.

4.4.2.1 Alertas

O sistema de alertas foi concretizado em torno do componente *SettingsController*, responsável por decidir, a cada detecção recebida, se deve ser acionado um estímulo visual ou sonoro. A partir desta decisão, a execução divide-se em duas vertentes:

Alerta Sonoro

O alerta sonoro recorre ao componente nativo *AudioSource* do *Unity*, exposto no *NoiseController*. O *SettingsController* mantém a respetiva referência e aciona a reprodução sempre que a política de decisão seleciona o modo *Audio*. O encadeamento é simples, após a chegada de uma embalagem para validação, é solicitado ao *SettingsController* a emissão do alerta, caso a opção guardada seja sonoro, invoca a reprodução no *AudioSource*. A configuração deste (Figura 4.7) segue os seguintes parâmetros, *Spatial Blend* = 0 para som 2D ouvido de igual forma independentemente da posição do utilizado, *Play On Awake* = *false* para evitar reprodução automática e *Loop* = *false* para impedir repetições.

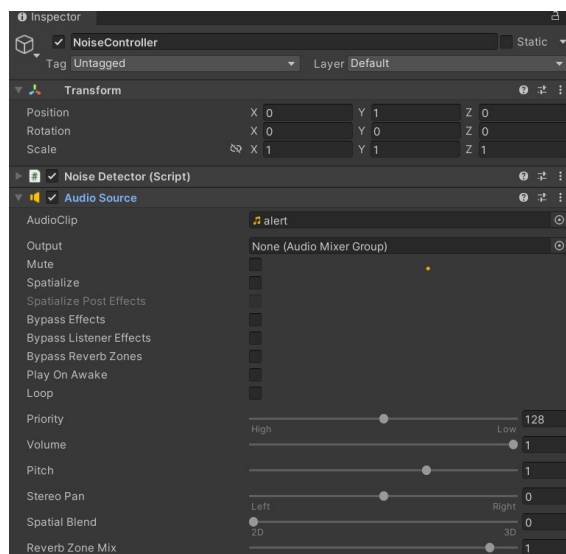


Figura 4.7: *Inspector* para configuração do *AudioSource* para alerta sonoro

O método `PlayAlertAudioAndWait()` (Excerto 4.6) interrompe qualquer reprodução ativa, repõe o cursor no início do *clip*, inicia a reprodução e aguarda até o som terminar. Esta estratégia elimina sobreposição de sons, assegura a percepção integral do estímulo e preserva a ordem do fluxo.

```
1 public IEnumerator PlayAlertAudioAndWait()
2 {
3     if (audioSource == null || audioSource.clip == null) yield
4     break;
5
6     audioSource.Stop();
7     audioSource.time = 0f;
8     audioSource.Play();
9
10    float dur = audioSource.clip.length / Mathf.Max(0.01f,
11    audioSource.pitch);
12    yield return new WaitForSecondsRealtime(dur);
13 }
```

Excerto 4.6: SettingsController.cs: Reprodução do alerta sonoro

Para efeitos de protótipo foi utilizado o ficheiro de áudio `alert.wav`, correspondente ao *clip Software interface start* disponibilizado gratuitamente pela plataforma *Mixkit* [63].

Alerta Visual

O alerta visual aparece como um ícone discreto na *interface*. No projeto, este ícone é a imagem `AlertImage`. A lógica do alerta está no componente `AlertVisual`, localizado no `UIVisualAlertController`. Sempre que a política de decisão escolhe o modo `Visual`, `SettingsController` invoca o método `TriggerVisual()` do `AlertVisual`, ver (Excerto 4.7). A sequência de ativação segue os seguintes passos: repõe o alfa da `AlertImage` a zero, ativa o objeto, coloca a imagem no topo da hierarquia do `Canvas` através de `SetAsLastSibling`, cancela animações em curso e inicia a rotina de realce `FlashCorner`.

```
1 public void TriggerVisual()
2 {
3     if (alertImage == null) return;
4
5     var c = alertImage.color; c.a = 0f; alertImage.color = c;
6     alertImage.gameObject.SetActive(true);
7     alertImage.transform.SetAsLastSibling();
8
9     StopAllCoroutines();
10    StartCoroutine(FlashCorner());
11 }
```

Excerto 4.7: AlertVisual.cs: Ativação do alerta visual

A animação baseia-se em três parâmetros do `AlertVisual` (Excerto 4.8): `flashCount`, número de ciclos de realce, `fadeDuration`, duração de cada transição de opacidade e de cada pulso, e `pulseScale`, intensidade do pulso de escala. O componente

`VisualSettingsDisplay` expõe *sliders* ligados diretamente a estes campos. Sempre que o operador altera um *slider*, o `AlertVisual` recebe o novo valor e o efeito reflete a alteração de imediato. Este ajuste direto no próprio dispositivo simplifica a calibração em contexto real de utilização da aplicação, permitindo adaptar-se às preferências do operador.

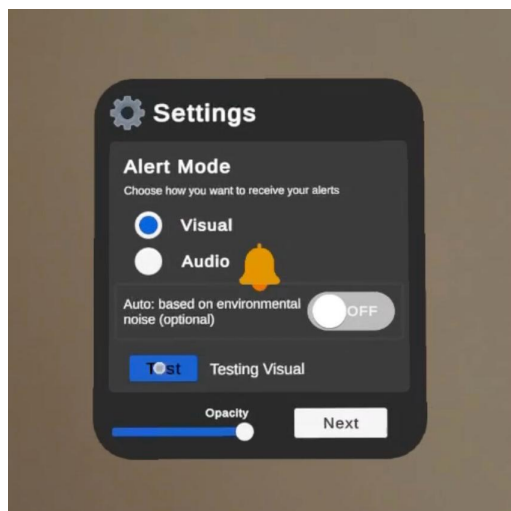
```

1 for (int i = 0; i < flashCount; i++)
2 {
3     StartCoroutine(Fade(alertImage, 0f, 1f, fadeDuration));
4     yield return Pulse(rt, original, original * pulseScale,
5         fadeDuration);
6
7     StartCoroutine(Fade(alertImage, 1f, 0f, fadeDuration));
8     yield return Pulse(rt, original * pulseScale, original,
9         fadeDuration);
10 }

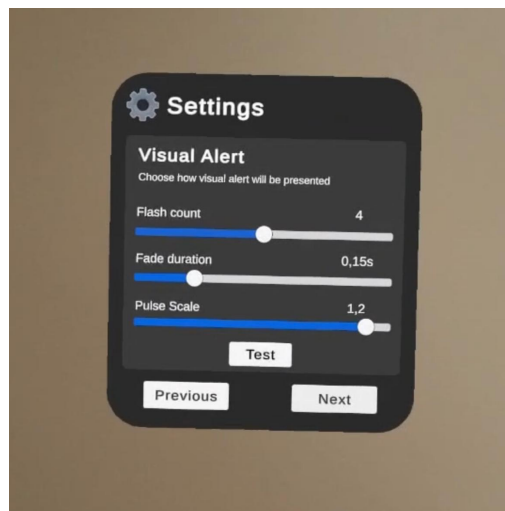
```

Excerto 4.8: `AlertVisual.cs`: Ciclo de *fade* e pulso

A Figura 4.8a ilustra o alerta visual ativo no visor e a Figura 4.8b apresenta os *sliders* de configuração dos parâmetros de alerta, o que permite ao operador ajustar os valores em tempo real.



(a) Interface de seleção do modo de alerta



(b) Interface de afinação do alerta visual

Figura 4.8: Interfaces relativas ao alerta visual

Alerta Automático

O modo Auto atua sobre a preferência base do operador. A decisão ocorre no momento em que chega uma nova imagem para validação. O `SettingsController` consulta o `NoiseDetector` para obter uma leitura rápida do ambiente, classifica o contexto como ruidoso, calmo ou indisponível e escolhe o tipo de alerta, `Visual` em ambiente ruidoso, `Audio` em ambiente calmo ou preferência base quando a leitura falha. A rotina `TriggerAlertForNewZip` (Excerto 4.9) aplica a regra acima e executa o estímulo escolhido, `AlertVisual` para `Visual`, `PlayAlertAudioAndWait` para `Audio`.

```

1 public IEnumerator TriggerAlertForNewZip()
2 {
3     AlertMode effective = SelectedAlertMode;
4
5     if (effective == AlertMode.Auto)
6     {
7         bool? loud = null;
8         if (noiseDetector != null)
9             yield return StartCoroutine(noiseDetector.
10 CheckEnvironmentSound(b => loud = b));
11
12         effective = loud == null ? BaseAlertPreference
13             : (loud.Value ? AlertMode.Visual : AlertMode.
14 Audio);
15     }
16
17     if (effective == AlertMode.Visual) alertVisual.TriggerVisual
18 (); else if (effective == AlertMode.Audio) yield return
19 StartCoroutine(PlayAlertAudioAndWait());
20 }

```

Excerto 4.9: SettingsController.cs: Decisão automática de alerta

O `NoiseDetector` confirma a disponibilidade do microfone do dispositivo, recolhe amostras durante cerca de um segundo, calcula o valor médio absoluto e compara com um limiar configurável, `noiseThreshold = 0.02` (Excerto 4.10). Devolve `true` para ruidoso, `false` para calmo, `null` quando o microfone não está disponível. Estes valores ficam expostos no *Inspector* e permitem ajustar o comportamento ao contexto de utilização.

```

1 AverageVolume = totalSamples > 0 ? sum / totalSamples : 0f;
2 bool noisyEnvironment = AverageVolume > noiseThreshold;
3 callback(noisyEnvironment);

```

Excerto 4.10: NoiseDetector.cs: Cálculo do volume médio e retorno do estado do ambiente

Tanto o limiar de som como o intervalo de leitura foram definidos apenas como parâmetros experimentais para prova de conceito, sendo totalmente ajustáveis no *inspector* do *Unity* em função do contexto real de utilização.

4.4.2.2 Modos de Inspeção

Foram implementados três modos distintos de inspeção (Figura 4.9): *Simple Mode*, *Complex Mode* e *Auto Mode*. A lógica de funcionamento foi encapsulada em classes próprias (`DefectSimpleValidator` e `DefectComplexValidator`), chamadas pelo `InspectionController`, que atua como coordenador do processo. Em todos os casos, a apresentação visual das anomalias é assegurada pelo `DefectOverlayController`, que sobrepõe as *bounding boxes* na imagem recebida.

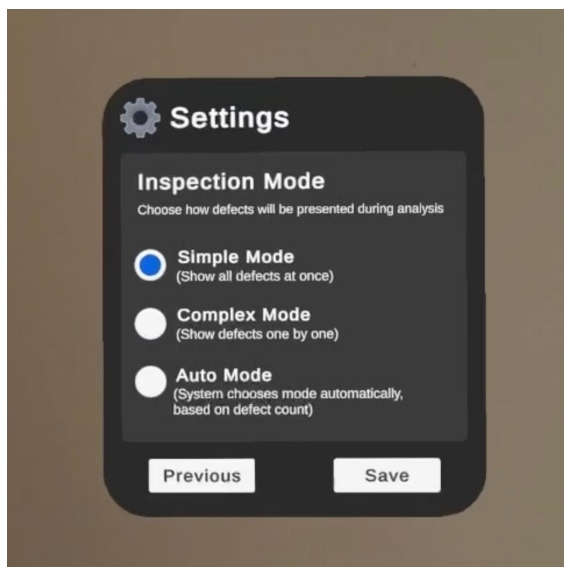


Figura 4.9: Interface de configuração do modo de inspeção

Inspeção Simples

No Simple Mode, todos os defeitos de uma embalagem são apresentados em simultâneo (Figura 4.10). O operador pode validar ou rejeitar a totalidade com uma única ação, o que permite um processo mais rápido em situações de elevado volume de inspeções. Este comportamento é implementado na classe `DefectSimpleValidator`, através do método `OnDecision`, que recebe a decisão global ("*Reject Defects*" ou "*Confirm Defects*") e a encaminha para o `InspectionController`, responsável por terminar a análise de acordo com a escolha do operador (ver Excerto 4.11).

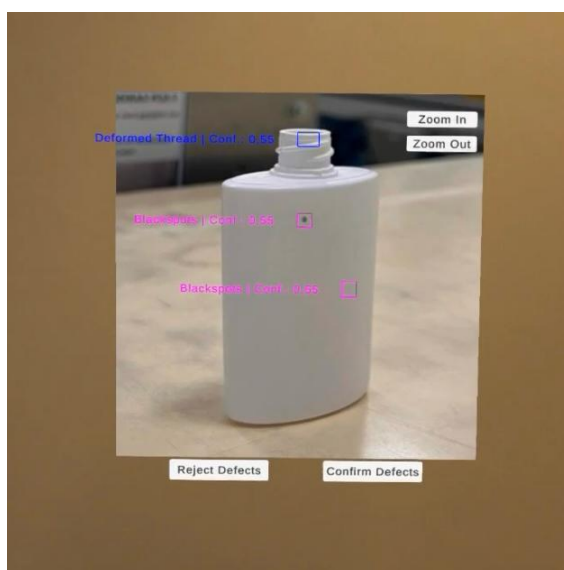


Figura 4.10: Interface do modo de inspeção simples

```
1 private void OnDecision(bool approved) {  
2     if (inspectionController != null)  
3         inspectionController.EndAnalysis(approved);  
4     ...  
5 }
```

Excerto 4.11: DefectSimpleValidator.cs: aceitação ou rejeição global

Inspeção Complexa

No *Complex Mode*, cada defeito é validado individualmente (Figura 4.11). O sistema apresenta um a um os defeitos reportados, permitindo que o operador confirme ou rejeite cada sugestão separadamente ("*Reject Defect*" ou "*Confirm Defect*"). Durante o processo, um contador visível indica quantos defeitos ainda faltam validar, reforçando a noção de progresso. Este modo é gerido pela classe *DefectComplexValidator*, através do método *OnDefectResponse*, que processa a resposta do operador a cada defeito. Quando o operador confirma a existência de defeito, a inspeção termina de imediato, uma vez que a embalagem já não cumpre os requisitos de qualidade e não há necessidade de validar os restantes. Pelo contrário, quando o operador rejeita a sugestão de defeito, o sistema avança para o próximo até concluir a lista (ver Excerto 4.12).

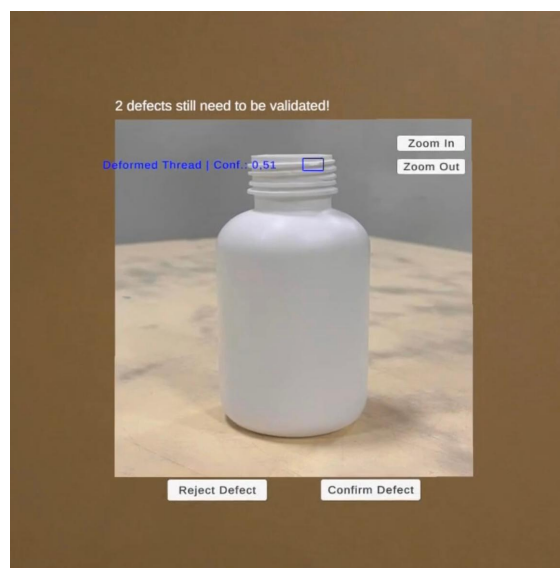


Figura 4.11: Interface do modo de inspeção complexo

```
1 public void OnDefectResponse(bool isDefect)  
2 {  
3     if (isDefect) { inspectionController.EndAnalysis(true); return; }  
4  
5     currentDefectIndex++;  
6     ShowCurrentDefect();  
7 }
```

Excerto 4.12: DefectComplexValidator.cs: resposta por defeito

Inspeção Automática

O Auto Mode procura adaptar o processo ao contexto de cada embalagem. A decisão entre utilizar o Simple Mode ou Complex Mode é tomada com base no número de defeitos detetados: se o valor ultrapassar um limiar configurável (`defectThreshold`), o sistema escolhe o modo simples, privilegiando rapidez, caso contrário, aplica o modo complexo, assegurando maior detalhe na validação. No protótipo desenvolvido, este limiar foi definido por omissão como quatro, apenas para efeitos de demonstração, sendo totalmente ajustável conforme o contexto ou os requisitos da aplicação. Esta lógica está implementada no `InspectionController`, que redefine dinamicamente o modo a partir do número de defeitos registados (ver Excerto 4.13).

```
1 [SerializeField] private int defectThreshold = 4;
2
3 if (mode == SettingsController.InspectionMode.Auto) {
4     int defeitos = overlayManager != null ? overlayManager.
5     GetCurrentDefects().Count : 0;
6     mode = defeitos >= defectThreshold
7         ? SettingsController.InspectionMode.Simple
8         : SettingsController.InspectionMode.Complex;
9 }
```

Excerto 4.13: `InspectionController.cs`: decisão do modo de inspeção

Independentemente do modo de inspeção escolhido, todas as decisões do operador ficam registadas. Estes registos permitem não só documentar o processo de inspeção, como também servir de base a futuras melhorias, nomeadamente a possibilidade de integração com modelos de visão computacional. No entanto, no âmbito deste trabalho, o histórico não é ainda explorado em detalhe, funcionando apenas como repositório de *feedback*. A implementação dos três modos de inspeção permitiu explorar diferentes equilíbrios entre rapidez e detalhe na validação de defeitos. Esta diversidade confirma a flexibilidade do protótipo e abre espaço para futuras extensões que considerem critérios adicionais no processo de decisão.

Apresentação da Informação

O componente `DefectOverlayController` é responsável pela camada visual associada às deteções. A partir do ficheiro `detections.json`, carrega a imagem correspondente e armazena a lista de defeitos da inspeção atual. Garante ainda a apresentação nos dois níveis: `Mode Simple` ou `Mode Complex`. Esta separação mantém a lógica de desenho isolada e reutilizável, o que simplifica a manutenção e a evolução do sistema. Após a receção de um ficheiro `.zip`, já descrita anteriormente, o sistema identifica o caminho para o `detections.json` e delega ao `DefectOverlayController` o carregamento da deteção. Quando o processo termina, é emitido o evento `OnNewDetection`, escutado pelo `InspectionController`, que ativa o painel de validação, trata o alerta e apresenta o botão *Begin Analysis*. O método `ShowImage` aplica a textura da embalagem ao componente `RawImage`, ajustando visibilidade e dimensões. Já o método `ShowDetectionAtIndex` sincroniza o `overlayParent` com as dimensões do `RawImage`, estabelecendo origem e escala para o desenho das caixas. O cálculo da posição e dimensão de cada caixa está ilustrado no Excerto 4.14. Assim, a mesma base gráfica serve ambos os modos de inspeção, evitando duplicação de código.

```
1 RectTransform rt = box.GetComponent<RectTransform>();  
2 rt.sizeDelta = new Vector2(defect.width, defect.height);  
3 rt.anchoredPosition = new Vector2(defect.x - imageWidth / 2f,  
4     defect.y - imageHeight / 2f);
```

Excerto 4.14: DefectOverlayController.cs: Cálculo da posição e dimensão da caixa de defeito

No modo *Simple*, o método `DisplayAllDefects(lista)` limpa a sobreposição anterior e instancia caixas para cada defeito. No modo *Complex*, `DisplaySingleDefect(defeito)` destaca apenas elemento de cada vez. As posições das caixas são calculadas a partir de coordenadas normalizadas, convertidas para o referencial do *RawImage*, cuja origem está no centro. Cada defeito surge com uma cor associada ao tipo e acompanhado por uma etiqueta com nome e valor de confiança (Figura 4.12).



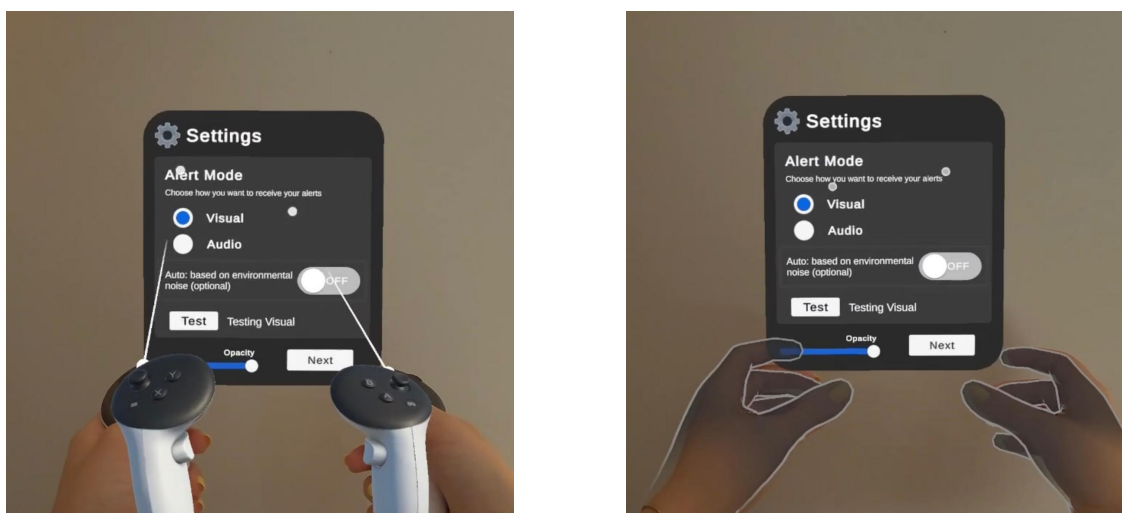
Figura 4.12: Interface com sobreposição de defeitos: caixas coloridas por tipo e etiquetas com nome e confiança

4.4.2.3 Interação

A interação foi concebida de forma multimodal, permitindo ao operador validar ou rejeitar defeitos através de duas modalidades distintas: mãos virtuais, e controladores físicos.

Mãos Virtuais e Controladores Físicos

A base da interação foi construída com os componentes disponibilizados pelo *Meta Quest All-in-One SDK*, nomeadamente o *Camera Rig* e o módulo *OVRInteractionComprehensive*, que fornece uma configuração integrada para interações com suporte a mãos (*OVRHands*) e controladores (*OVRControllers*). Com os controladores (Figura 4.13a) e com as mãos virtuais (Figura 4.13b), o operador aponta para os elementos do painel de inspeção através de um ponteiro *laser*. A diferença está no método de seleção: no caso dos controladores, a confirmação é feita pressionando o gatilho, enquanto com as mãos a seleção resulta do gesto de *pinch*.



(a) Interação através de controladores físicos

(b) Interação através de mãos virtuais

Figura 4.13: Exemplo de suporte a interação por mãos e controladores

Para que os gestos das mãos e os cliques dos controladores fossem corretamente reconhecidos pela *interface* gráfica do *Unity*, foi necessário complementar os componentes de UI padrão com módulos específicos da *Oculus/Meta*. No *Canvas*, ao *Graphic Raycaster* foi adicionado o *OVR Raycaster*, que processa os apontadores das mãos e dos controladores. No *EventSystem*, o *OVR Input Module* e o *Pointable Canvas Module* asseguram que as interações são convertidas em eventos da UI, como o *Click*. Por fim, o objeto auxiliar *ISDK_RayCanvasInteraction* define a superfície interativa do painel e encaminha os apontadores recebidos para o sistema de eventos.

4.4.2.4 Configuração e Personalização

A personalização da *interface* foi tratada como requisito funcional, uma vez que os contextos industriais variam em ruído, iluminação e preferências de operação. O protótipo inclui dois mecanismos de adaptação, opacidade regulável do *Canvas* e controlo de *zoom* da imagem de inspeção.

Ajuste da opacidade

O painel de configuração utiliza um *CanvasGroup* cuja propriedade *alpha* varia de forma contínua entre um mínimo e a opacidade total. O mínimo impede perda de contraste e mantém elementos essenciais visíveis, o que evita estados de *interface* inutilizável. O valor resulta do mapeamento linear do *slider* para o intervalo $[minAlpha, 1]$, seguido de aplicação simultânea aos painéis de configuração e à imagem exibida. Este comportamento encontra-se no controlador de opacidade, *OpacityController*. O método *SetTransparency* mapeia o valor do *slider* para o intervalo válido e invoca *ApplyAlpha*, que atualiza o *CanvasGroup* e a *Image* (Excerto 4.15).

```

1 void SetOpacity(float value) {
2     float mapped = Mathf.Lerp(minAlpha, 1f, value);
3     ApplyAlpha(mapped);
4 }
5

```

Excerto 4.15: OpacityController: Método SetTransparency

A Figura 4.14 ilustra dois níveis de opacidade distintos.

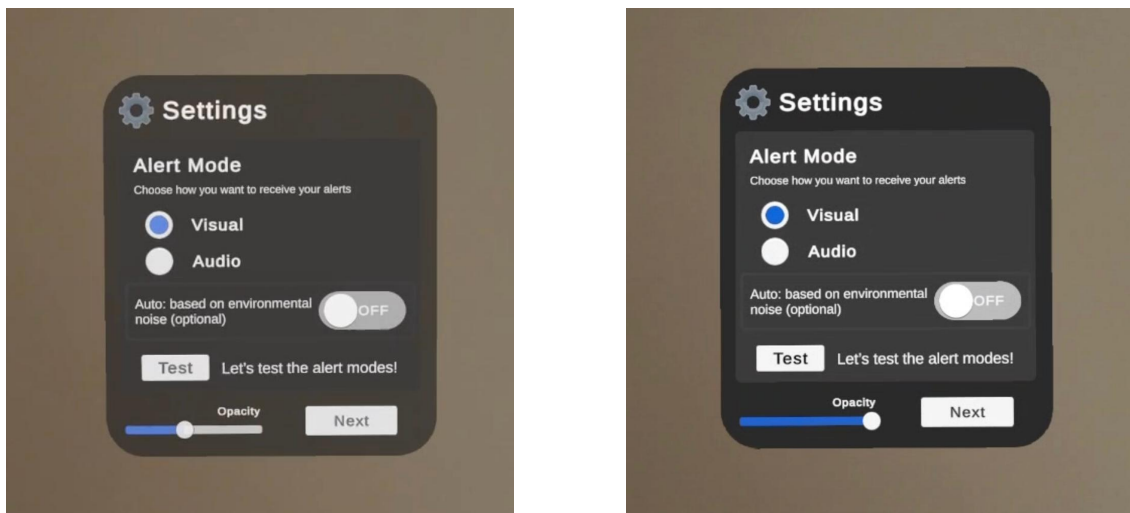


Figura 4.14: Interface com diferentes níveis de opacidade

O componente `CanvasZoom` implementa o sistema de *zoom* sobre a imagem de inspeção. A lógica atua diretamente no `RectTransform` que contém a imagem, aplicando incrementos definidos por `zoomStep` e respeitando limites mínimo e máximo (`minScale` e `maxScale`) para preservar proporção e legibilidade. A interação é feita através dos botões *Zoom In* e *Zoom Out*, que invocam os métodos públicos `ZoomIn()` e `ZoomOut()`. Ambos recorrem ao método interno `ApplyZoom()` (Excerto 4.16), que calcula a nova escala, aplica os limites e inicia a animação assíncrona através de `SmoothZoom()`. Esta rotina interpola gradualmente entre a escala inicial e a final, garantindo uma transição suave durante o tempo configurado em `zoomDuration`.

```

1 private void ApplyZoom(float step)
2 {
3     if (targetCanvas == null) return;
4
5     Vector3 targetScale = targetCanvas.localScale + Vector3.one * step;
6     targetScale = ClampScale(targetScale);
7
8     if (currentZoom != null)
9         StopCoroutine(currentZoom);
10
11     currentZoom = StartCoroutine(SmoothZoom(targetScale));
12 }

```

Excerto 4.16: CanvasZoom.cs: cálculo e animação de zoom

Os botões de *zoom* apenas são apresentados após o início da análise, de modo a evitar competição visual com o alerta inicial e reduzir distrações. A Figura 4.15 mostra um exemplo da *interface*, com a imagem de inspeção em modo complexo e a mesma após *zoom out*. Esta lógica está integrada no *InspectionController*, que gere a visibilidade e ativação dos controladores.



Figura 4.15: Exemplo de *zoom* na imagem de inspeção

Capítulo 5

Avaliação

Este capítulo apresenta o processo de avaliação do protótipo. São descritos os procedimentos metodológicos seguidos, o questionário aplicado e a análise dos resultados obtidos. A análise inclui a caracterização da amostra, os resultados quantitativos e qualitativos e a comparação entre diferentes perfis de utilizadores. O capítulo termina com a discussão dos resultados.

5.1 Metodologia

A avaliação do protótipo foi conduzida com um grupo de participantes sem qualquer experiência prévia em processos de inspeção de qualidade manual no contexto industrial. Esta decisão surgiu como resposta prática à realidade do projeto: a única empresa parceira envolvida dispunha de poucos colaboradores com funções de inspeção, e a distância geográfica tornava inviável a sua participação direta. Ao recorrer a participantes não especializados, foi possível agilizar o processo de avaliação, garantir maior flexibilidade na organização das sessões e evitar a dependência de um grupo profissional restrito e pouco acessível. Para mitigar a ausência de experiência técnica, estes foram instruídos a adotar o papel de um operador numa linha de produção, de modo a simular o contexto real e, ainda assim, recolher perceções relevantes sobre a aplicação. A sessão de avaliação foi estruturada em duas fases principais:

1. **Inspeção manual de defeitos:** os participantes foram convidados a simular o trabalho de um operador numa linha de produção, recebendo embalagens físicas, algumas com defeitos e outras sem. A atividade foi enquadrada na realidade da inspeção industrial, tendo-lhes sido pedido que identificassem manualmente a presença de anomalias. O exercício permitiu evidenciar limitações típicas do processo tradicional, como a repetição contínua, o esforço cognitivo e a influência da fadiga ou da variabilidade humana na consistência dos resultados de validação;
2. **Utilização do protótipo em RA:** de seguida, foi apresentado o dispositivo *Meta Quest 3S*, através do qual os participantes puderam interagir livremente com a aplicação desenvolvida. Nesta fase, exploraram as diferentes funcionalidades, incluindo os modos de inspeção, alertas, opções de configuração e interação, sendo incentivados a testar diversos cenários de utilização para avaliar a flexibilidade da aplicação.

No final de cada sessão experimental, foi aplicado um questionário com o objetivo de recolher a opinião dos participantes e avaliar a sua experiência com o sistema.

5.2 Questionário de Avaliação

O questionário foi desenvolvido com base em métricas reconhecidas para a avaliação de sistemas de RA, apresentadas no Capítulo 2, nomeadamente desempenho percebido, usabilidade, conforto, utilidade percebida, preferências e experiência subjetiva do participante. O objetivo foi recolher tanto dados quantitativos como qualitativos, assegurando uma avaliação abrangente do protótipo. Seguindo a estrutura do questionário, a primeira secção do mesmo teve como foco a caracterização do perfil dos participantes, em particular a sua familiaridade prévia com óculos de RA. Este dado revelou-se essencial para interpretar adequadamente as respostas, uma vez que participantes com experiência anterior tendem a adotar critérios de avaliação distintos face aos sem contacto prévio com a tecnologia. As questões de usabilidade avaliaram a clareza da configuração inicial, a facilidade de compreensão dos alertas visuais e sonoros, bem como a intuição da interação através de mãos virtuais ou controladores. Esta dimensão permitiu aferir não apenas a curva de aprendizagem exigida, mas também a simplicidade de utilização. O bloco dedicado à utilidade percebida teve como objetivo compreender até que ponto os participantes consideram que a aplicação poderia contribuir para a redução de erros, a aceleração do processo de inspeção e a sua aplicabilidade em cenários industriais reais, refletindo o impacto potencial do protótipo. No que diz respeito ao conforto, analisou-se de que forma o uso do dispositivo influenciava a experiência. Foram considerados fatores como o conforto dos óculos durante a tarefa, a fadiga visual, o cansaço associado ao uso prolongado das mãos virtuais e o grau de intrusão dos alertas. Estes aspetos são particularmente relevantes em cenários industriais, onde os operadores permanecem ativos durante longos períodos. Foram ainda incluídas questões de preferências, destinadas a identificar os modos de inspeção e os tipos de alerta considerados mais adequados pelos participantes, fornecendo indicações úteis para o desenho de futuras versões da aplicação. Por fim, foi disponibilizado um espaço para *feedback* qualitativo, no qual os participantes puderam partilhar opiniões abertas sobre os aspetos mais valorizados, as limitações percecionadas e sugestões adicionais de melhoria. O questionário combinou escalas de *Likert* (de 1 a 5), questões de escolha múltipla e perguntas abertas, o que possibilitou a recolha integrada de dados quantitativos e qualitativos. A versão completa encontra-se disponível no Anexo C.

5.3 Análise dos Resultados

A análise dos dados recolhidos através do questionário combinou duas abordagens complementares, quantitativa e qualitativa. Na vertente quantitativa foram calculadas as médias (*M*) e os desvios padrão amostrais (*DP*), permitindo caracterizar tendências centrais e avaliar a consistência das respostas. Com base nestes indicadores, foi possível distinguir os componentes da aplicação que suscitaram avaliações mais homogêneas daqueles que revelaram maior variabilidade, sugerindo diferentes níveis de perceção ou experiência entre os participantes. Em paralelo, na vertente qualitativa, as respostas abertas foram analisadas na íntegra e organizadas em categorias, pontos fortes, aspetos a melhorar e propostas de otimização, contribuindo para contextualizar os resultados numéricos e aprofundar a interpretação dos dados. Para além desta análise global, foi ainda efetuada uma comparação entre participantes com e sem experiência prévia em óculos de RA, permitindo identificar diferenças na consistência das respostas e no detalhe das observações.

5.3.1 Caracterização da amostra

A amostra foi constituída por 21 participantes, cuja experiência prévia com óculos de RA foi considerada um fator relevante para a interpretação dos resultados. Destes, 11 nunca tinham utilizado este tipo de dispositivo, enquanto os restantes 10 relataram algum contacto anterior (Figura 5.1). Esta distinção revelou-se pertinente para compreender variações nas métricas de interação, nomeadamente no uso de mãos virtuais e controladores, onde a familiaridade tecnológica pode influenciar a perceção e o desempenho.

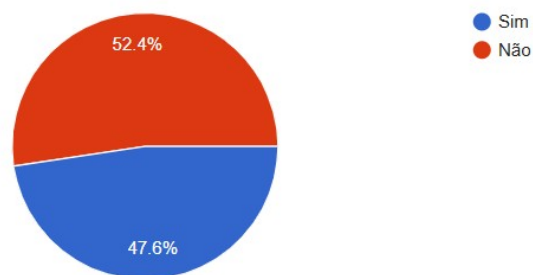


Figura 5.1: Distribuição da experiência prévia com óculos de RA

5.3.2 Resultados Quantitativos

Os resultados do questionário foram organizados em quatro áreas principais, usabilidade, conforto, utilidade e preferências. A análise foi conduzida de forma agregada, considerando o total de participantes, e em cada área são destacadas as tendências mais relevantes, apoiadas por representações gráficas que facilitam a interpretação dos dados.

Usabilidade

Na dimensão da usabilidade, os participantes avaliaram aspetos que vão desde a clareza do processo de configuração inicial até à interação com mãos virtuais e controladores. De forma geral, as médias obtidas situam-se acima do ponto médio da escala, o que indica que a aplicação foi considerada utilizável. No entanto, observam-se diferenças relevantes entre dimensões específicas. Como ilustrado no gráfico de barras da Figura 5.2, os elementos mais valorizados foram a clareza dos alertas ($M = 4,81$, $DP = 0,51$) e a rapidez de resposta do sistema ($M = 4,76$, $DP = 0,44$), evidenciando um desempenho estável e previsível. A compreensão do processo de validação ($M = 4,57$, $DP = 0,75$) e a utilidade do zoom ($M = 4,52$, $DP = 0,75$) também receberam avaliações positivas, embora com maior dispersão, o que sugere uma experiência menos uniforme entre participantes. A clareza das sobreposições visuais ($M = 4,29$, $DP = 0,85$) obteve uma avaliação mais moderada, com variabilidade acentuada. Este resultado indica que, enquanto alguns participantes interpretaram facilmente as caixas e etiquetas, outros enfrentaram dificuldades, possivelmente relacionadas com a forma como foram apresentadas. As maiores variações foram observadas na interação com mãos virtuais ($M = 4,24$, $DP = 0,77$) e com controladores ($M = 4,24$, $DP = 1,26$). Apesar das médias iguais, o desvio padrão mais elevado nos controladores revela uma experiência mais desigual, sugerindo que a familiaridade com tecnologia de RA ou preferências individuais influenciaram significativamente a perceção da interação. Em suma, os resultados não apontam para problemas críticos de usabilidade, mas revelam que certos elementos do protótipo funcionaram de forma mais consistente do que outros. O sistema de alertas destacou-se pela fiabilidade, enquanto a interação manual e a clareza das sobreposições visuais surgem

como áreas prioritárias para refinamento, seja através de ajustes na *interface* ou de mais opções de personalização adaptadas ao perfil do utilizador.

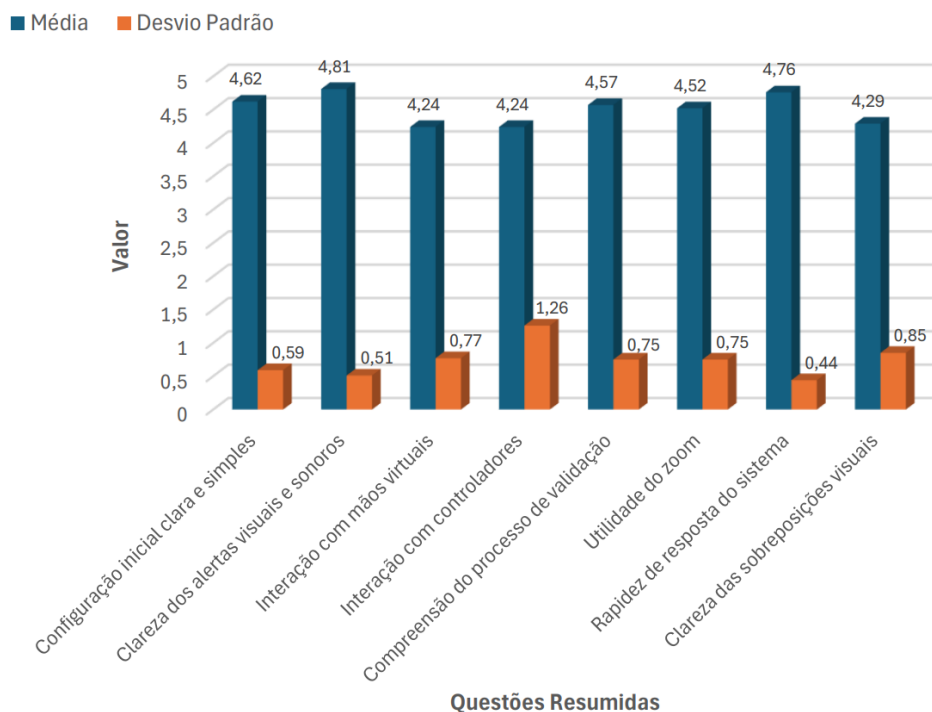


Figura 5.2: Resultados da média e desvio padrão das questões de usabilidade

Conforto

No gráfico de barras da Figura 5.3 observa-se que os aspetos relacionados com o conforto foram, em geral, avaliados de forma positiva, embora com maior variabilidade do que na dimensão da usabilidade. Os alertas pouco invasivos destacaram-se como o item mais bem classificado ($M = 4,71$, $DP = 0,56$), evidenciando que o sistema de notificações foi eficaz sem interferir negativamente na experiência de utilização. Os controladores também foram considerados confortáveis ($M = 4,62$, $DP = 0,74$), enquanto os óculos receberam uma avaliação aceitável ($M = 4,00$, $DP = 1,00$), ainda que com uma dispersão mais acentuada, o que sugere diferenças na perceção individual do seu uso. A interação com mãos virtuais obteve uma média de 3,9 ($DP = 1,04$), indicando uma avaliação globalmente positiva, mas com sinais de cansaço ou necessidade de maior adaptação por parte de alguns participantes. O indicador mais crítico foi a fadiga visual, que registou a média mais baixa de toda a avaliação ($M = 2,95$, $DP = 1,47$). Este resultado revela que o uso prolongado dos óculos provocou desconforto em parte dos participantes, sublinhando a importância de considerar melhorias ergonómicas e otimizações visuais em futuras versões do protótipo. Em síntese, os dados demonstram que o protótipo proporciona uma experiência confortável em interações de curta duração. No entanto, também evidenciam limitações relevantes para uma utilização prolongada em contexto industrial, com destaque para os efeitos da fadiga visual e a variabilidade na adaptação à interação.



Figura 5.3: Resultados da média e desvio padrão das questões de conforto

Utilidade

Na dimensão da utilidade, os resultados foram consistentemente elevados, com todas as médias acima de 4,6 pontos e desvios padrão inferiores a 0,7 (Figura 5.4). Este padrão revela um consenso alargado entre os participantes quanto ao valor acrescentado do protótipo, destacando-se o potencial apoio à deteção de defeitos ($M = 4,67$, $DP = 0,66$) e a perceção de redução de erros ($M = 4,71$, $DP = 0,64$). A aceleração das inspeções ($M = 4,62$, $DP = 0,59$) foi igualmente valorizada. Quanto à aplicabilidade em contexto real ($M = 4,62$, $DP = 0,57$), esta obteve também uma classificação elevada, embora condicionada pelo facto do protótipo ter sido testado num cenário simulado com participantes sem experiência prévia na área de inspeção de qualidade. Globalmente, esta foi a dimensão mais consensual do questionário. Mesmo num ambiente controlado, os participantes reconheceram o potencial da aplicação para apoiar tarefas reais, especialmente na deteção de defeitos, na redução de erros e na aceleração das inspeções.

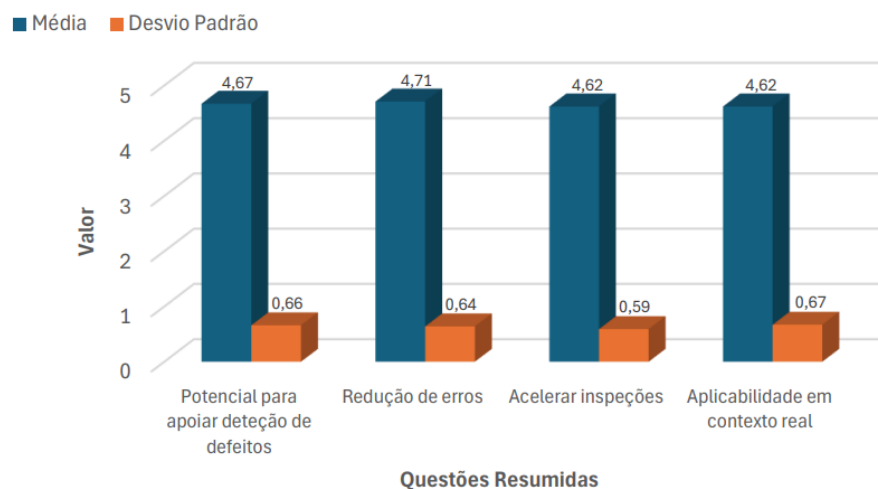


Figura 5.4: Resultados da média e desvio padrão das questões de utilidade

Preferências

Para além das escalas de *Likert*, os participantes foram convidados a indicar preferências relativamente ao tipo de alerta (Figura 5.5a) e ao modo de inspeção (Figura 5.5b). A maioria (66,7%) demonstrou preferência pelo alerta automático adaptativo ao ruído, evidenciando a valorização de mecanismos que se ajustam dinamicamente às condições ambientais. O alerta sonoro foi selecionado por 19% dos participantes, enquanto o visual obteve apenas 14,3%, o que reforça a menor aceitação de soluções estáticas em contextos variáveis. Relativamente ao modo de inspeção, a opção automática (47,6%) foi igualmente a mais votada, seguida do modo complexo (28,6%) e do simples (23,8%). Os dados recolhidos evidenciam a valorização de soluções que se ajustem de forma inteligente ao contexto de utilização, demonstrando que a adaptabilidade do sistema é um fator central para a sua aceitação e eficácia.

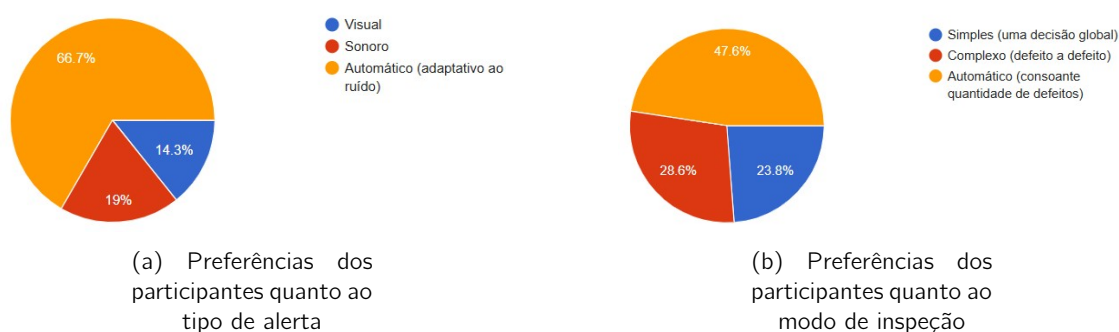


Figura 5.5: Distribuição das preferências dos participantes

5.3.3 Resultados Qualitativos

Da análise das respostas abertas emergiram três grupos temáticos:

- **Pontos fortes:** simplicidade de utilização, clareza dos alertas e rapidez do sistema. Houve referências frequentes à confiança transmitida pelo processo de validação e à intuição do fluxo de utilização (alerta, iniciar análise, validar);
- **Aspetos a melhorar:** fadiga visual após uso prolongado, cansaço associado às mãos virtuais e necessidade de maior personalização dos alertas e de maior robustez em cenários ruidosos;
- **Sugestões:** inclusão de novos tipos de alerta ou possibilidade de apresentar mais do que um ao mesmo tempo, bem como maior controlo sobre zoom e opacidade.

A análise qualitativa complementa os dados quantitativos, destacando a aceitação geral da aplicação e apontando limitações concretas relacionadas com a ergonomia e a adaptabilidade da *interface*.

5.3.4 Comparação entre Perfis de Participantes

Para além da análise global, foi realizada uma comparação entre os participantes com e sem experiência prévia na utilização de óculos de RA. O objetivo foi identificar diferenças de avaliação entre os dois grupos e perceber de que forma a familiaridade influenciava a perceção da aplicação.

De forma geral, ambos os grupos atribuíram classificações positivas, com médias consistentemente acima do ponto médio da escala. No entanto, os dados revelam padrões distintos. Os participantes sem experiência prévia demonstraram maior entusiasmo, atribuindo valores mais elevados em diversos itens, como a configuração inicial ($M = 4,73$ vs. $4,50$), a utilidade do *zoom* ($M = 4,73$ vs. $4,30$), a rapidez de resposta ($M = 4,82$ vs. $4,70$) e a perceção de redução de erros ($M = 4,91$ vs. $4,50$). Estes resultados sugerem que o protótipo foi intuitivo e bem recebido por participantes inexperientes, possivelmente devido à ausência de expectativas técnicas prévias. Por outro lado, os participantes com experiência destacaram-se em aspetos mais específicos e técnicos, como a clareza dos alertas visuais e sonoros ($M = 4,90$ vs. $4,73$) e das sobreposições visuais ($M = 4,40$ vs. $4,18$). Além disso, reportaram menor fadiga visual ($M = 2,60$ vs. $3,27$), o que indica uma maior adaptação ao esforço visual e ao uso prolongado dos óculos.

Relativamente à variabilidade, verificou-se que o grupo sem experiência apresentou desvios padrão mais elevados em vários itens, como a interação com controladores ($DP = 1,30$ vs. $1,26$) e a fadiga visual ($DP = 1,56$ vs. $1,35$). Em contrapartida, os participantes com experiência revelaram respostas mais consistentes, com valores de dispersão mais baixos em itens como a clareza dos alertas ($DP = 0,32$ vs. $0,65$) ou conforto dos controladores ($DP = 0,52$ vs. $0,92$).

As Figuras 5.6 e 5.7 ilustram estas diferenças, evidenciando o maior entusiasmo, mas também maior dispersão, entre os participantes sem experiência, e uma avaliação mais consistente e técnica por parte dos participantes com contacto prévio com dispositivos de RA.

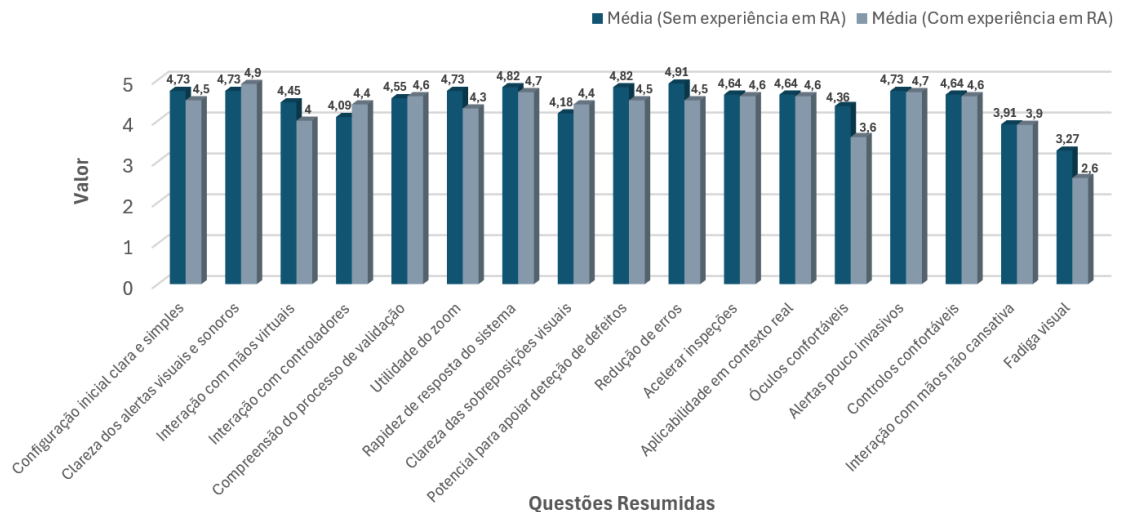


Figura 5.6: Comparação das médias das respostas por experiência prévia em RA

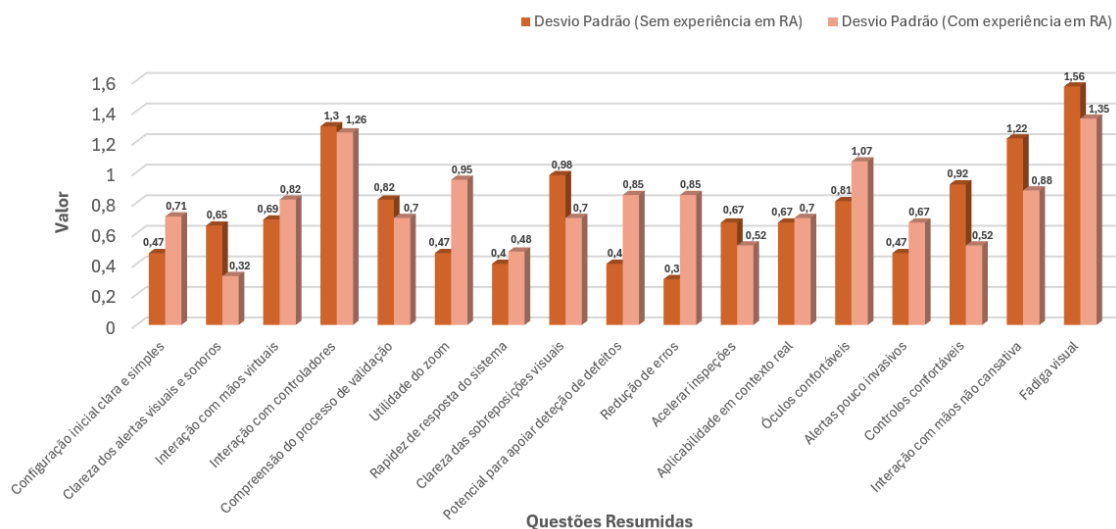


Figura 5.7: Comparação dos desvios padrão das respostas por experiência prévia em RA

No que toca às preferências registadas, estas não diferem de forma substancial da análise global. Tanto participantes com experiência prévia como sem atribuíram maior valor ao alerta automático adaptativo ao ruído ambiente e ao modo de inspeção automático, mantendo-se as restantes opções em percentagens residuais. Estes resultados sugerem que a experiência prévia em RA não teve impacto relevante nas escolhas relativas a preferências de interação.

No plano qualitativo, embora não se tenham verificado diferenças estruturais entre participantes com e sem experiência prévia, emergiram algumas tendências. Os participantes inexperientes focaram-se sobretudo em aspetos gerais de conforto e clareza da *interface*, enquanto que os participantes com experiência prévia apresentaram comentários mais técnicos e direcionados, sugerindo melhorias específicas na interação, no *zoom* e na integração dos diferentes tipos de alerta. Este contraste sugere que a familiaridade com tecnologias de RA conduz a uma avaliação mais detalhada e crítica, ainda que as perceções globais de simplicidade, rapidez e utilidade tenham sido partilhadas transversalmente pelos dois grupos.

Em síntese, a experiência prévia em RA não alterou de forma radical a perceção global da aplicação, mas influenciou a consistência das respostas e o nível de detalhe das observações, reforçando a importância de considerar perfis distintos de utilizadores no desenho de futuras soluções.

5.4 Discussão

A análise dos resultados permite discutir até que ponto o protótipo desenvolvido respondeu aos objetivos propostos e quais as suas implicações em contexto de inspeção industrial. De forma geral, as avaliações indicam que a aplicação foi bem recebida, sobretudo no que diz respeito à clareza dos alertas e à rapidez de resposta do sistema, dimensões que obtiveram as pontuações mais elevadas. Estes aspetos sugerem que as opções de desenho implementadas conseguiram reduzir a curva de aprendizagem e proporcionar uma experiência de utilização intuitiva desde o primeiro contacto.

Ainda assim, os resultados não foram homogêneos em todas as dimensões. A interação com mãos virtuais e controladores apresentou maior variabilidade, indicando diferenças na adaptação dos participantes. A análise por grupos revelou que esta dispersão foi mais acentuada nos participantes sem experiência prévia em RA, enquanto os experientes mostraram avaliações mais consistentes, ainda que ligeiramente mais críticas. Este padrão sugere que o contacto prévio com dispositivos de RA facilita a adaptação e reduz a variabilidade das respostas, mas também eleva as expectativas face à aplicação.

Também a clareza das sobreposições visuais, embora avaliada positivamente, gerou maior dispersão, apontando para a necessidade de rever aspetos da legibilidade gráfica. A inclusão de mecanismos de personalização, por exemplo ao nível da opacidade, cor ou formato das etiquetas, pode permitir adaptar a camada visual a diferentes perfis de utilizadores e contextos industriais. Além disso, a introdução de interações alternativas, como comandos de voz, pode mitigar as limitações observadas na interação manual e reduzir o esforço visual e físico associado ao uso prolongado.

Na dimensão da utilidade, as médias elevadas e os desvios padrão baixos mostram um consenso claro, no qual os participantes reconheceram o potencial do protótipo para apoiar a deteção de defeitos, reduzir erros e acelerar inspeções. Este resultado foi consistente tanto em participantes com experiência prévia em RA como nos inexperientes, o que sugere que a percepção de valor não depende do grau de familiaridade tecnológica. Contudo, a validação decorreu num cenário experimental, pelo que será necessário testar a solução em ambiente real de produção para confirmar estes benefícios em métricas objetivas de desempenho.

O conforto foi a dimensão mais ambivalente. Os participantes valorizaram o carácter pouco invasivo dos alertas e o conforto dos controladores, mas a fadiga visual destacou-se como limitação relevante, em especial entre os inexperientes. Isto indica que o uso prolongado dos óculos pode comprometer a experiência de utilização, tornando essencial investir em melhorias ergonómicas e visuais nas próximas versões do protótipo.

As preferências declaradas confirmaram a relevância de mecanismos adaptativos. Tanto nos alertas como no modo de inspeção, a maioria dos participantes optou pela modalidade automática, que ajusta o funcionamento ao contexto. Este resultado confirma que os mesmos privilegiam soluções que aliviem a carga cognitiva e simplifiquem o processo de decisão. Ainda assim, a existência de participantes que optaram por modos manuais indica que a flexibilidade deve ser preservada, permitindo acomodar estilos de utilização distintos. Importa salientar que a experiência prévia em RA não se traduziu em diferenças relevantes nas preferências, o que sugere que variáveis como a experiência em inspeção de qualidade poderiam ser mais determinantes para escolhas como a opção pelo modo complexo. Esta hipótese abre caminho para investigações futuras que cruzem a familiaridade tecnológica com a experiência prática em tarefas de inspeção.

A análise qualitativa reforçou estas conclusões. Os comentários livres destacaram como pontos fortes a simplicidade de utilização, a clareza dos alertas e a rapidez do sistema, confirmando a confiança transmitida pelo processo de inspeção. Em contrapartida, foram apontadas limitações como fadiga visual, esforço na interação com mãos virtuais e necessidade de maior personalização dos alertas. Estas observações alinham-se com os resultados quantitativos e fornecem pistas concretas para evoluir o protótipo, como a introdução de novos tipos de alerta, maior controlo sobre parâmetros visuais e reforço da robustez em cenários industriais ruidosos.

Em síntese, os resultados sugerem que o protótipo oferece uma base sólida para inspeção assistida por RA, mas também revelam que a sua aceitação depende de fatores como a experiência prévia com tecnologias semelhantes e a capacidade de adaptação da *interface* às necessidades de diferentes utilizadores. Para validar e consolidar estas conclusões, será essencial avançar para estudos em contexto real de produção e com amostras mais alargadas.

Capítulo 6

Conclusões e Trabalho Futuro

Neste capítulo são apresentadas as conclusões finais sobre o trabalho realizado, bem como aspectos a considerar em trabalho futuro.

6.1 Conclusões

O trabalho desenvolvido permitiu demonstrar a viabilidade da utilização de óculos de RA como ferramenta de apoio à inspeção visual de defeitos em contexto industrial. A aplicação concebida integra um conjunto de funcionalidades que respondem às exigências operacionais do setor, desde a receção de dados externos até à validação em diferentes modos de inspeção, incluindo sistemas de alerta configuráveis e opções de personalização da *interface*. Esta abordagem revela-se particularmente relevante num cenário de crescente automatização e digitalização dos processos produtivos.

A avaliação realizada junto de 21 participantes, recorrendo a métodos quantitativos e qualitativos, permitiu recolher dados significativos sobre a usabilidade e a perceção da aplicação. Os resultados indicam que a solução é intuitiva, clara e funcional, com destaque para a eficácia dos alertas e para a rapidez de resposta do sistema, validando as escolhas de desenho adotadas. Verificou-se ainda uma preferência generalizada pelos modos automáticos de alerta e inspeção, associados à capacidade do sistema de responder de forma eficiente e autónoma às exigências da tarefa, sem necessidade de intervenção constante por parte do utilizador. Contudo, foram identificadas limitações que merecem atenção em futuras iterações. A fadiga visual após períodos prolongados de utilização, bem como a variabilidade na adaptação às diferentes modalidades de interação, nomeadamente as mãos virtuais e controladores físicos, evidenciam a necessidade de aprofundar aspectos relacionados com a ergonomia e a experiência do utilizador. Embora estes fatores não inviabilizem a proposta, constituem áreas críticas de melhoria que poderão influenciar a sua adoção em larga escala.

Este projeto exploratório não se limita à validação de uma solução tecnológica, mas sim constitui uma base sólida para aprofundar o estudo da RA como ferramenta industrial, abrindo caminho para investigações futuras que explorem a sua integração em sistemas mais complexos e adaptativos. Embora os resultados sejam preliminares, revelam um potencial significativo que poderá ser desenvolvido e consolidado em contextos reais de produção, promovendo soluções mais robustas, escaláveis e alinhadas com os desafios da indústria moderna.

6.2 Trabalho Futuro

Os resultados obtidos evidenciam diversas oportunidades de evolução que poderão reforçar significativamente o impacto e a aplicabilidade do sistema. A ergonomia e o conforto visual surgem como áreas prioritárias, sendo essencial otimizar os elementos gráficos da *interface* para mitigar a fadiga ocular em sessões prolongadas. Neste sentido, a introdução de mecanismos de ajuste automático de luminosidade e contraste poderá contribuir para uma experiência de utilização mais confortável e adaptada às condições ambientais. A interação com o sistema também poderá beneficiar de abordagens mais diversificadas e naturais, como a integração de alertas hápticos, o aperfeiçoamento da manipulação através de mãos virtuais e controladores físicos, e até mesmo a exploração de comandos de voz. Estas melhorias visam promover uma utilização mais intuitiva, consistente e acessível, ajustando-se aos diferentes perfis e preferências dos utilizadores. A validação do protótipo em ambiente industrial real constitui uma etapa fundamental para aferir a sua robustez e eficácia em cenários práticos. Testes realizados com operadores especializados, em condições típicas de produção, como ruído ambiente, variações de iluminação e ritmo de trabalho elevado, permitirão avaliar o desempenho do sistema em contextos exigentes e dinâmicos. Por fim, a integração com sistemas de aprendizagem automática abre novas possibilidades de evolução. A associação dos registos de decisão dos operadores a modelos de visão computacional poderá potenciar abordagens híbridas, combinando deteção automática com supervisão humana, e promovendo uma inspeção mais eficiente, colaborativa e inteligente. Estas linhas de desenvolvimento não visam apenas aperfeiçoar o protótipo existente, mas posicioná-lo como uma solução tecnológica escalável e adaptável, capaz de responder aos desafios reais da indústria e de acompanhar a transformação dos processos de inspeção de qualidade em ambientes produtivos cada vez mais exigentes.

Bibliografia

- [1] Agnieszka Kujawińska e Katarzyna Vogt. «Human Factors in Visual Quality Control». Em: *Management and Production Engineering Review* 6 (jun. de 2015). doi: 10.1515/MPER-2015-0013.
- [2] Arne Seeliger, Long Cheng e Torbjørn Netland. «Augmented reality for industrial quality inspection: An experiment assessing task performance and human factors». Em: *Computers in Industry* 151 (2023). doi: 10.1016/j.compind.2023.103985.
- [3] Li Da Xu, Eric L. Xu e Ling Li. «Industry 4.0: state of the art and future trends». Em: *International Journal of Production Research* 56.8 (2018), pp. 2941–2962. doi: 10.1080/00207543.2018.1444806.
- [4] R. Azuma, Y. Baillet, R. Behringer, S. Feiner, S. Julier e B. MacIntyre. «Recent advances in augmented reality». Em: *IEEE Computer Graphics and Applications* 21.6 (2001), pp. 34–47. doi: 10.1109/38.963459.
- [5] Sustainable Plastics. *Sustainable Plastics*. Acedido em: 9 de setembro 2025. url: <https://sustainableplastics.pt/en/>.
- [6] Neutroplast. *Neutroplast*. Acedido em: 9 setembro 2025. url: <https://www.neutroplast.com/>.
- [7] Toqeer Ali Syed, Muhammad Shoaib Siddiqui, Hurria Binte Abdullah, Salman Jan, Abdallah Namoun, Ali Alzahrani, Adnan Nadeem e Ahmad B. Alkhodre. «In-Depth Review of Augmented Reality: Tracking Technologies, Development Tools, AR Displays, Collaborative AR, and Security Concerns». Em: *Sensors* 23.1 (2023). doi: 10.3390/s23010146.
- [8] Ronald T. Azuma. «A Survey of Augmented Reality». Em: *Presence: Teleoperators and Virtual Environments* 6.4 (1997), pp. 355–385. doi: 10.1162/pres.1997.6.4.355.
- [9] Paul Milgram e Fumio Kishino. «A Taxonomy of Mixed Reality Visual Displays». Em: *IEICE Transactions on Information Systems* E77-D.12 (1994), pp. 1321–1329.
- [10] Pedro Alberto. *Virtual Reality (VR), Augmented Reality (AR) Mixed Reality (MR)*. Acedido em: 29 de novembro de 2024. Jun. de 2022. url: <https://www.adeso-mobile.de/xr/vr-ar-mr-unterschiede/>.
- [11] Gazi Erkan Bostanci, Nadia Kanwal, Shoaib Ehsan e Adrian Clark. «User Tracking Methods for Augmented Reality». Em: *International Journal of Human-Computer Studies* 5.1 (fev. de 2013), pp. 93–98.
- [12] Feng Zhou, Henry Duh e Mark Billinghurst. «Trends in Augmented Reality Tracking, Interaction and Display: A Review of Ten Years of ISMAR». Em: *2008 7th IEEE/ACM International Symposium on Mixed and Augmented Reality 2* (set. de 2008), pp. 193–202. doi: 10.1109/ISMAR.2008.4637362.
- [13] Yassir El Filali e Krit Salah-ddine. «Augmented Reality Types and Popular Use Cases». Em: *Proceedings of the International Conference of Computer Science and Renewable Energies*. Vol. 8. Jan. de 2018, pp. 107–110. doi: 10.5220/0009776301070110.

- [14] Greg Welch, Gary Bishop, Leandra Vicci, Stephen Brumback, Kurtis Keller e D'ardo Colucci. «High-Performance Wide-Area Optical Tracking: The HiBall Tracking System». Em: *Presence* 10.1 (2001), pp. 1–21. doi: 10.1162/105474601750182289.
- [15] Julie Carmigniani, Borko Furht, Marco Anisetti, Paolo Ceravolo, Ernesto Damiani e Misa Ivkovic. «Augmented Reality Technologies, Systems and Applications». Em: *Multimedia Tools and Applications* 51.1 (dez. de 2010), pp. 341–377. doi: 10.1007/s11042-010-0660-6.
- [16] Mitchell Doughty, Nilesh R. Ghugre e Graham A. Wright. «Augmenting Performance: A Systematic Review of Optical See-Through Head-Mounted Displays in Surgery». Em: *Journal of Imaging* 8.7 (jul. de 2022).
- [17] Matthew Hanna, Ishtiaque Ahmed, Jeffrey Nine, Shyam Prajapati e Liron Pantanowitz. «Augmented Reality Technology Using Microsoft HoloLens in Anatomic Pathology». Em: *Archives of Pathology Laboratory Medicine* 142 (jan. de 2018). doi: 10.5858/arpa.2017-0189-0A.
- [18] Microsoft. *HoloLens 2 hardware*. Acedido em: 22 de maio de 2025. 2023. url: <https://learn.microsoft.com/en-us/hololens/hololens2-hardware>.
- [19] Meta Platforms. *Compare Devices*. Acedido em: 22 de maio de 2025. 2024.
- [20] Mercedes Soto Ramos. «Development of Mixed Reality Applications Using the Magic Leap One Device». Tese de mestrado. Universidad Autónoma de Madrid, Escuela Politécnica Superior, set. de 2019.
- [21] Magic Leap, Inc. *Magic Leap | Groundbreaking augmented reality solutions*. Acedido em: 22 de maio de 2025. 2025. url: <https://www.magicleap.com/>.
- [22] Carlos Pérez, Javier Verón, Francisca Pérez, M Ángeles Moraga, Coral Calero e Carlos Cetina. *A Comparative Analysis of Energy Consumption Between The Widespread Unreal and Unity Video Game Engines*. 2024. arXiv: 2402.06346 [cs.SE]. url: <https://arxiv.org/abs/2402.06346>.
- [23] John K Haas. «A History of the Unity Game Engine». Em: 2014. url: <https://api.semanticscholar.org/CorpusID:86824974>.
- [24] Antonín Šmíd. «Comparison of unity and unreal engine». Em: *Czech Technical University in Prague* (2017), pp. 41–61.
- [25] Dhiraj Amin e Sharvari Govilkar. «Comparative Study of Augmented Reality Sdk's». Em: *International Journal on Computational Science Applications* 5.1 (fev. de 2015), pp. 11–26. doi: 10.5121/ijcsa.2015.5102.
- [26] Inc. Qualcomm Technologies. *Vuforia Developer Portal*. Acedido em: 21 de maio de 2025. 2024. url: <https://developer.vuforia.com/>.
- [27] Zainab Oufqir, Abdellatif El Abderrahmani e Khalid Satori. «ARKit and ARCore in serve to augmented reality». Em: *2020 International Conference on Intelligent Systems and Computer Vision (ISCV)*. 2020, pp. 1–7. doi: 10.1109/ISCV49265.2020.9204243.
- [28] Meta. *Meta XR Core SDK Overview*. Acedido em: 21 de maio de 2025. url: https://developers.meta.com/horizon/documentation/unity/unity-core-sdk/?utm_source=chatgpt.com.
- [29] Suzanna, Sasmoko, Ford Lumban Gaol e Tanty Oktavia. «Augmented Reality SDK Overview for General Application Use». Em: *International Journal of Advanced Computer Science and Applications* 14.11 (2023). doi: 10.14569/IJACSA.2023.0141106.
- [30] Mikko Korkiakoski, Paula Alavesá e Panos Kostakos. «Preference in Voice Commands and Gesture Controls With Hands-Free Augmented Reality With Novel Users». Em: *IEEE Pervasive Computing* 23.1 (2024), pp. 18–26.

- [31] Yuchong Zhang, Bastian Orthmann, Michael Welle, Jonne Haastregt e Danica Kragic. «LLM-Driven Augmented Reality Puppeteer: Controller-Free Voice-Commanded Robot Teleoperation». Em: (fev. de 2025). doi: 10.48550/arXiv.2502.09142.
- [32] Yee-Yin Choong, Kurtis Goad e Kevin C. Mangold. «Augmented Reality (AR) Usability Evaluation Framework: The Case of Public Safety Communications Research». Em: (2022). doi: <https://doi.org/10.6028/NIST.IR.8422>.
- [33] Andreas Dünser e Mark Billinghurst. «Evaluating Augmented Reality Systems». Em: *Handbook of Augmented Reality*. Ed. por Borko Furht. Springer New York, jul. de 2011, pp. 289–307. doi: 10.1007/978-1-4614-0064-6_13.
- [34] Leonel Merino, Magdalena Schwarzl, Matthias Kraus, Michael Sedlmair, Dieter Schmalstieg e Daniel Weiskopf. «Evaluating Mixed and Augmented Reality: A Systematic Literature Review (2009-2019)». Em: *2020 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*. 2020, pp. 438–451. doi: 10.1109/ISMAR50242.2020.00069.
- [35] Pasquale Daponte, Luca De Vito, Francesco Picariello e Maria Riccio. «State of the Art and Future Developments of Augmented Reality for Measurement Applications». Em: *Measurement* 57 (2014), pp. 53–70. doi: 10.1016/j.measurement.2014.07.009.
- [36] Stefan Graser, Felix Kirschenlohr e Stephan Böhm. «User Experience Evaluation of Augmented Reality: A Systematic Literature Review». Em: *arXiv preprint* (2024). doi: 10.48550/arXiv.2411.12777.
- [37] Prokopia Vlachogianni e Nikolaos Tselios. «Perceived Usability Evaluation of Educational Technology Using the Post-Study System Usability Questionnaire (PSSUQ): A Systematic Review». Em: *Sustainability* 15.17 (2023). doi: 10.3390/su151712954.
- [38] Yulyani Arifin, Thomas Galih Sastria e Edo Barlian. «User Experience Metric for Augmented Reality Application: A Review». Em: *Procedia Computer Science* 135 (2018), pp. 648–656. doi: 10.1016/j.procs.2018.08.221.
- [39] Lea M. Daling e Sabine J. Schlittmeier. «Effects of Augmented Reality-, Virtual Reality-, and Mixed Reality-Based Training on Objective Performance Measures and Subjective Evaluations in Manual Assembly Tasks: A Scoping Review». Em: *Human Factors* 66.2 (2024), pp. 589–626. doi: 10.1177/00187208221105135.
- [40] Vighnesh Bharat Gholap e Wanwan Li. «Past, Present, and Future of the Augmented Reality (AR)-Enhanced Interactive Techniques: A Survey». Em: *2023 7th International Conference on Machine Vision and Information Technology (CMVIT)*. 2023, pp. 143–148. doi: 10.1109/CMVIT57620.2023.00035.
- [41] Preeti Sirohi, Amit Agarwal e Piyush Maheshwari. «A survey on Augmented Virtual Reality: Applications and Future Directions». Em: *2020 Seventh International Conference on Information Technology Trends (ITT)*. 2020, pp. 99–106. doi: 10.1109/ITT51279.2020.9320869.
- [42] Hardeep Patel, Manav Ukani, Meet Kavathiya e Yogesh Kumar. «Comprehensive Analysis of Artificial Intelligence in Virtual Reality and Augmented Reality». Em: *2023 10th IEEE Uttar Pradesh Section International Conference on Electrical, Electronics and Computer Engineering (UPCON)*. Vol. 10. 2023, pp. 457–463. doi: 10.1109/UPCON59197.2023.10434866.
- [43] Carlos Flavián, Sergio Ibáñez-Sánchez e Carlos Orús. «The impact of virtual, augmented and mixed reality technologies on the customer experience». Em: *Journal of Business Research* 100 (2019), pp. 547–560. doi: <https://doi.org/10.1016/j.jbusres.2018.10.050>.

- [44] Yushan Siriwardhana, Pawani Porambage, Madhusanka Liyanage e Mika Ylianttila. «A Survey on Mobile Augmented Reality With 5G Mobile Edge Computing: Architectures, Applications, and Technical Aspects». Em: *IEEE Communications Surveys Tutorials* 23.2 (2021), pp. 1160–1192. doi: 10.1109/COMST.2021.3061981.
- [45] John Soroushian, Sabine Neschke e Babu Jackson. *Thinking Ahead About XR*. Bipartisan Policy Center, abr. de 2022.
- [46] Rong Wen, Wei-Liang Tay, Binh P. Nguyen, Chin-Boon Chng e Chee-Kong Chui. «Hand gesture guided robot-assisted surgery based on a direct augmented reality interface». Em: *Computer Methods and Programs in Biomedicine* 116.2 (2014), pp. 68–80. doi: <https://doi.org/10.1016/j.cmpb.2013.12.018>.
- [47] T. Fick, J. A. M. van Doormaal, L. Tomic, R. J. van Zoest, J. W. Meulstee, E. W. Hoving e T. P. C. van Doormaal. «Fully automatic brain tumor segmentation for 3D evaluation in augmented reality». Em: *Neurosurgical Focus* 51.2 (ago. de 2021). doi: 10.3171/2021.5.FOCUS21200.
- [48] Vladimir Ivanov, Anton Krivtsov, Sergey Strelkov, Anton Smirnov, Roman Shipov, Vladimir Grebenkov, Valery Rumyantsev, Igor Gheleznyak, Dmitry Surov, Mikhail Korzhuk e Valery Koskin. «Practical Application of Augmented/Mixed Reality Technologies in Surgery of Abdominal Cancer Patients». Em: *Journal of Imaging* 8 (jun. de 2022), p. 183. doi: <https://doi.org/10.3390/jimaging8070183>.
- [49] Hasaneen Al Janabi, Abdullatif Aydin, Sharanya Palaneer, Nicola Macchione, Ahmed Al-Jabir, Muhammad Khan, Prokar Dasgupta e Kamran Ahmed. «Effectiveness of the HoloLens Mixed Reality Headset in Minimally Invasive Surgery: A Simulation-based Feasibility Study». Em: *Surgical Endoscopy* 34 (mar. de 2020). doi: 10.1007/s00464-019-06862-3.
- [50] V. Vlahakis, M. Ioannidis, John Karigiannis, M. Tsotros, Michail Gounaris, Didier Stricker, T. Gleue, P. Daehne e L. Almeida. «Archeoguide: An augmented reality guide for archaeologist sites». Em: *Computer Graphics and Applications, IEEE* 22 (out. de 2002), pp. 52–60. doi: 10.1109/MCG.2002.1028726.
- [51] S. Casas, M. Zancanaro, O. Stock e T. Kuflik. «Enhancing Cultural Tourism Experiences with Augmented Reality Technologies: The PRISMA Project». Em: *Handbook of Research on Technologies and Cultural Heritage: Applications and Environments*. 2011, pp. 314–331.
- [52] Meredydd Williams, Kelvin K. K. Yao e Jason R. C. Nurse. *ToARist: An Augmented Reality Tourism App created through User-Centred Design*. Acedido em: 21 de maio de 2025. 2018. url: <https://arxiv.org/abs/1807.05759>.
- [53] Keyu Chen, Weiwei Chen, Chun Li e Jack Cheng. «A BIM-based location aware AR collaborative framework for facility maintenance management». Em: *Electronic Journal of Information Technology in Construction* 24 (jul. de 2019), pp. 360–380.
- [54] Nai-Hsin Pan e Nurani Nanda Isnaeni. «Integration of Augmented Reality and Building Information Modeling for Enhanced Construction Inspection—A Case Study». Em: *Buildings* 14 (2024). doi: <https://doi.org/10.3390/buildings14030612>.
- [55] Alexander Gerger, Harald Urban e Christian Schranz. «Augmented Reality for Building Authorities: A Use Case Study in Austria». Em: *Buildings* 13 (2023). doi: <https://doi.org/10.3390/buildings13061462>.
- [56] Michele Fiorentino, Antonio Uva, Michele Gattullo, Saverio Debernardis e Giuseppe Monno. «Augmented reality on large screen for interactive maintenance instructions». Em: *Computers in Industry* 65 (jan. de 2013). doi: 10.1016/j.compind.2013.11.004.

- [57] Emanuele Marino, Loris Barbieri, Fabio Bruno e Maurizio Muzzupappa. «Assessing user performance in augmented reality assembly guidance for industry 4.0 operators». Em: *Computers in Industry* 157-158 (2024), p. 104085. doi: <https://doi.org/10.1016/j.compind.2024.104085>.
- [58] João Alves, Bernardo Marques, Paulo Dias e Beatriz Santos. «Using augmented reality for industrial quality assurance: a shop floor user study». Em: *The International Journal of Advanced Manufacturing Technology* 115 (jul. de 2021), pp. 105–116. doi: <https://doi.org/10.1007/s00170-021-07049-8>.
- [59] Andrzej Szajna, Roman Stryjski, Waldemar Woźniak, Norbert Chamier-Gliszczyński e Tomasz Królikowski. «The Production Quality Control Process, Enhanced with Augmented Reality Glasses and the New Generation Computing Support System». Em: *Procedia Computer Science* 176 (set. de 2020), pp. 3618–3625.
- [60] Muneer Bani Yassein, Mohammed Q. Shatnawi, Shadi Aljwarneh e Razan Al-Hatmi. «Internet of Things: Survey and open issues of MQTT protocol». Em: *2017 International Conference on Engineering MIS (ICEMIS)*. 2017, pp. 1–6.
- [61] EMQX Team. *Using MQTT in Unity with M2MqttUnity Library: A Step-by-Step Guide*. Acedido em: 30 de maio de 2025. 2023. url: <https://www.emqx.com/en/blog/using-mqtt-in-unity-with-m2mqttunity-library-a-step-by-step-guide>.
- [62] Meta. *Meta Quest 3S*. Acedido em: 20 setembro 2025. 2024. url: <https://www.meta.com/quest/quest-3s/buy-now/>.
- [63] Envato. *Mixkit Free Alerts Sound Effects*. Acedido em: 7 de setembro de 2025. url: <https://mixkit.co/free-sound-effects/alerts/>.

Apêndice A

Documentos Éticos e Legais



Política de Privacidade da Avaliação da Aplicação de Realidade Aumentada para Inspeção de Qualidade

A presente Política de Privacidade descreve o tratamento de dados pessoais no âmbito da avaliação da aplicação de Realidade Aumentada desenvolvida para apoiar a inspeção de qualidade de embalagens plásticas, realizada como parte de um projeto académico de mestrado.

1. Finalidade do tratamento

A aplicação tem como finalidade apoiar a inspeção de qualidade de embalagens plásticas, fornecendo informação adicional no campo de visão do operador para facilitar a validação de defeitos. Importa salientar que a aplicação não recolhe quaisquer dados pessoais durante a sua utilização. A única recolha de informação ocorre através do questionário de avaliação, destinado a perceber a experiência dos participantes. Este questionário foca-se apenas em aspetos como usabilidade, clareza e utilidade da solução, sendo todas as respostas tratadas de forma anónima, sem possibilidade de identificar individualmente os participantes.

2. Dados pessoais

O questionário não solicita informações pessoais ou sensíveis que permitam identificar os participantes. Apenas são recolhidas respostas relativas à experiência de uso da aplicação.

3. Responsável pelo tratamento

O responsável pelo tratamento dos dados é o autor do projeto de mestrado, sob a supervisão dos respetivos orientadores académicos.

4. Período de retenção dos dados pessoais

Os dados recolhidos através do questionário serão conservados apenas durante o período necessário para a análise e redação da tese de mestrado. Após a entrega e defesa da mesma, todos os dados serão eliminados.

5. Partilha de dados pessoais

Os dados não serão partilhados com entidades terceiras, exceto com a equipa académica diretamente envolvida na supervisão do trabalho. Os resultados serão sempre apresentados em formato agregado e anónimo, exclusivamente para fins de avaliação da aplicação no contexto da tese.



6. Medidas de segurança

Para salvaguardar a confidencialidade das respostas recolhidas, foram adotadas medidas que asseguram a proteção contra acessos indevidos ou utilização abusiva. As respostas ao questionário são recolhidas de forma anónima, sem guardar qualquer identificador direto ou indireto dos participantes. O acesso aos dados é limitado ao autor do projeto e aos orientadores académicos, garantindo que apenas a equipa responsável tem contacto com a informação. Após a conclusão e entrega da tese, todos os dados serão eliminados definitivamente, assegurando que não permanecem registos posteriores.

7. Permissões da aplicação

A aplicação em si não recolhe dados pessoais nem acede a funcionalidades sensíveis do dispositivo. As únicas permissões utilizadas são:

- Microfone: exclusivamente para deteção do nível de ruído ambiente, com o objetivo de ajustar automaticamente o tipo de alerta (sonoro ou visual);
- Armazenamento local: limitado ao carregamento de imagens previamente preparadas para demonstração, sem registo de novos conteúdos produzidos pelo utilizador.

8. Direitos do titular dos dados

Os participantes no questionário podem exercer os seus direitos relativamente às respostas fornecidas, nomeadamente o direito de acesso, retificação ou eliminação dos dados, contactando o responsável pelo tratamento.

Figura A.1: Documento Política de Privacidade



Termos e Condições da Aplicação de Realidade Aumentada para Inspeção de Qualidade

A aplicação de Realidade Aumentada foi desenvolvida no âmbito de um projeto académico de mestrado, em parceria com a Neutroplast e no contexto da iniciativa Sustainable Plastics. O protótipo foi desenvolvido com o propósito de explorar o potencial da Realidade Aumentada no apoio à inspeção de qualidade de embalagens plásticas.

1. Aceitação dos termos

Ao utilizar a aplicação, o utilizador declara que compreende que se trata de um protótipo desenvolvido para fins de investigação académica e concorda com os presentes Termos e Condições. Caso não concorde, não deverá testar a aplicação.

2. Finalidade da aplicação

A aplicação destina-se exclusivamente a fins académicos, para avaliação da experiência de utilização de tecnologias de Realidade Aumentada em cenários de inspeção de qualidade. Não se trata de um produto comercial, nem deve ser utilizada em contextos reais de produção.

3. Obrigações do utilizador

O utilizador compromete-se a utilizar a aplicação apenas no âmbito das sessões de teste definidas pelo projeto académico.

4. Direitos de propriedade intelectual

Todos os direitos sobre os conteúdos da aplicação pertencem ao autor do projeto e às entidades académicas associadas. É proibida a sua reprodução, modificação ou distribuição sem autorização prévia.

5. Limitação de responsabilidade

Nem o autor do projeto, nem a instituição académica ou parceiros, poderão ser responsabilizados por quaisquer danos diretos ou indiretos resultantes do uso da aplicação fora do contexto de investigação definido.

6. Modificações

Os presentes Termos e Condições podem ser ajustados durante o decorrer do projeto académico, sendo disponibilizada a versão mais atualizada em anexo à dissertação.

Campus da FEUP
Rua Dr. Roberto Frias
4200-465 Porto, Portugal

T +351 222 094 000
info@inesctec.pt
www.inesctec.pt



7. Contactos

Para qualquer questão relacionada com estes Termos e Condições, os utilizadores poderão contactar o responsável do projeto através do endereço: mariana.almeida@inesctec.pt

Figura A.2: Documento Termos e Condições

Apêndice B

Criação de um Dataset

No âmbito do projeto *Sustainable Plastics*, foi desenvolvido um *dataset* de embalagens plásticas farmacêuticas com o objetivo de apoiar o treino e a validação de modelos de visão computacional dedicados à deteção e classificação de defeitos. Este *dataset* constitui um contributo complementar para os objetivos globais do projeto. A recolha de dados reais em ambiente industrial revelar-se-ia um processo moroso e incerto, uma vez que as embalagens em produção variam frequentemente e seria difícil garantir uma quantidade suficiente de amostras com diferentes tipos de defeitos para assegurar diversidade representativa. Para ultrapassar esta limitação, e com o apoio da empresa *Neutroplast*, foram obtidas amostras em perfeito estado das embalagens de plástico, às quais se introduziram manualmente diferentes tipos de defeitos. Esta abordagem permitiu criar um *dataset* controlado e representativo, garantindo consistência, diversidade de exemplos e alinhamento com os objetivos de investigação do projeto

B.1 Constituição do Dataset

A constituição do *dataset* será apresentada com base em três aspetos principais: a identificação dos defeitos selecionados, a metodologia utilizada para a aquisição das imagens e a organização final do mesmo.

B.1.1 Descrição dos Defeitos

Os defeitos selecionados foram baseados num catálogo de defeitos visuais fornecido pela *Neutroplast* (ver Figura B.1), e a sua inclusão foi justificada com base em fatores de relevância prática e viabilidade técnica.



Figura B.1: Catálogo com defeitos visuais

Assim sendo, os defeitos foram organizados em três grupos principais:

Defeitos Considerados

Os seguintes defeitos foram identificados como prioritários, devido à sua relevância no contexto industrial e à facilidade de reprodução manual:

- *Deformed Thread*: Defeito dependente do grau de deformação, sendo mais evidente com uma deformação acentuada;
- *Oval Neck*: A sua identificação depende da posição da câmara e do grau do defeito;
- *Black Spots*: Considerando apenas embalagens que sejam brancas (de cor homogênea) pois, em polímeros alternativos, as características podem introduzir artefactos visuais que mascaram o defeito;
- *Color Variation*: Depende do grau de variação e sofre influência da iluminação. A comparação é feita com uma embalagem de referência para correção de cor da imagem, garantindo a uniformidade.

Defeitos Secundários

Incluídos de forma condicional, os seguintes defeitos foram considerados menos prioritários devido à dificuldade de reprodução manual ou à menor frequência de ocorrência:

- *Cut on top of the neck*: Considerado dependendo do grau do defeito;
- *Neck with Material Entrapment Inside*: O defeito torna-se mais relevante quanto mais visível for a deformação;
- *Flashes*: Incluído de forma condicional, também dependendo da severidade do defeito. Este defeito é mais difícil de implementar manualmente.

Defeitos Não Considerados

Os seguintes defeitos foram excluídos devido à baixa frequência de ocorrência ou à complexidade de reprodução manual:

- *Bottle with hole in the bottom*: Este defeito foi excluído por ser menos frequente e devido à dificuldade de reprodução manual do mesmo;
- *Bottle surface with orange peel*: Excluído devido à sua natureza sutil e difícil reprodução manual do mesmo.

Todos os defeitos considerados no *dataset*, nomeados no ponto "Defeitos Considerados", foram introduzidos manualmente nas embalagens em boas condições, à exceção do defeito de variação de cor ("*Color Variation*"), no qual foi utilizada a ferramenta de edição de imagem *GNU Image Manipulation Program* (GIMP), que permitiu simular discrepâncias cromáticas típicas do processo de produção. Para os restantes defeitos, foi adotado um processo manual de indução controlada: defeitos como "*Black Spots*" foram simulados desenhando diretamente sobre as superfícies das embalagens, "*Oval Neck*" foi reproduzido pelo amassar do gargalo das embalagens, e "*Deformed Thread*" foi obtido cortando o gargalo das mesmas. Este processo garantiu a representatividade de todos os defeitos considerados.

B.1.2 Metodologia de Aquisição

O processo de aquisição de imagens seguiu um conjunto de condições e etapas bem definidas, com o objetivo de assegurar a uniformidade das imagens capturadas para o *dataset*. Abaixo, são descritas as condições estabelecidas, as etapas seguidas durante a captura e os dispositivos utilizados para a mesma.

B.1.2.1 Aquisição de Imagens

Condições:

As imagens foram capturadas seguindo as seguintes condições:

- **Luminosidade adequada**: Utilizou-se uma fonte de luz consistente;
- **Variedade de iluminação**: A aquisição das imagens foi feita com diferentes tipos de iluminação;
- **Evitar luz direta**: A luz direta deve ser evitada para não mascarar os defeitos presentes;
- **Sem textos nas embalagens**: As embalagens não devem conter nenhum texto para evitar confusão com defeitos;
- **Proximidade na captura**: A aquisição de imagens foi feita próxima da embalagem, minimizando o fundo das mesmas.

Etapas de aquisição:

Para a captura das imagens foram seguidas as seguintes etapas:

- **1.** Instalação do fundo desobstruído e contrastante (fundo preto), numa superfície nivelada;

- **2.** Instalação da embalagem (com ou sem defeito), imediatamente em frente ao fundo anteriormente instalado;
- **3.** Aquisição das imagens relativas ao objeto com defeito e sem defeito;
- **4.** Armazenamento das imagens num diretório específico, de acordo com o tipo de defeito.

Dispositivos Utilizados:

As imagens do *dataset* foram capturadas usando o iPhone 13. O dispositivo dispõe de uma câmara de 12 MP, distância focal de 26 mm no modo padrão, abertura de f/1.6 e estabilização ótica de imagem (OIS).

B.1.3 Estrutura do Dataset

O *dataset* ('dataset') é composto por dois diretórios principais:

- 'images/': Contém todas as imagens das embalagens de plástico;
- 'labels/': Contém as respetivas anotações de cada imagem.

Cada um destes diretórios é organizado em três pastas ('type1/', 'type2/' e 'type3/') que representam os três tipos distintos de embalagens de plástico, Figura B.3.



Figura B.2: Tipos de embalagens de plástico consideradas

Por sua vez, estas pastas são compostas por subdiretórios ('blackspots/', 'color_variation/', 'deformed_thread/', 'oval_neck/' e 'no_defect/') que representam os defeitos específicos das embalagens, que podem ser observados na Figura 4.2b.



Figura B.3: Exemplos de defeitos considerados

B.2 Preparação do Dataset

O processo de preparação dos dados foi composto por várias etapas, incluindo o pré-processamento das imagens, a aplicação de técnicas de *Data Augmentation* e a anotação dos defeitos. A seguir, são detalhados os principais passos realizados.

Pré-processamento dos dados

O pré-processamento teve como objetivo padronizar o tamanho das imagens:

- **Redimensionamento das Imagens** Todas as imagens foram redimensionadas para 500x640 pixels utilizando a opção "*Resize*" do Roboflow. Este redimensionamento garante que todas as imagens tenham a mesma dimensão, facilitando o processamento e treino do modelo.

Data Augmentation

Com as imagens padronizadas, foi desenvolvido um *script* em *Python* utilizando a biblioteca *Torchvision Transforms*, do *PyTorch*, para aplicar técnicas de *Data Augmentation*. Estas técnicas introduziram variações controladas nas imagens, como alterações de iluminação e orientação, com o objetivo de aumentar a robustez do modelo. As transformações aplicadas foram:

- **Flip Vertical Aleatório:** Esta transformação inverte as imagens verticalmente com uma probabilidade de 100%. Isto ajuda o modelo a reconhecer defeitos independentemente da orientação vertical da embalagem;

```
'transforms.RandomVerticalFlip(p=1)'
```

- **Ajuste Aleatório de Cor (ColorJitter):** Esta técnica aplica variações aleatórias nos valores de brilho, contraste, saturação e tonalidade da imagem. Com isto, simularam-se condições de iluminação e variabilidade de cores que podem ocorrer nas imagens reais. Os parâmetros definidos foram:

- Brilho: $\pm 20\%$
- Contraste: $\pm 20\%$

- Saturação: $\pm 20\%$
- Tonalidade: $\pm 20\%$

```
'transforms.ColorJitter(brightness=0.2,contrast=0.2,saturation=0.2, hue=0.2)'
```

A aplicação destas técnicas garantiu que o modelo fosse exposto a uma ampla gama de condições possíveis, melhorando a capacidade de generalização para novas imagens.

Anotação de Defeitos

Os defeitos nas imagens foram anotados manualmente utilizando a ferramenta *Computer Vision Annotation Tool (CVAT)*, que permite desenhar *bounding boxes* em torno dos defeitos (Figura B.4), e gerar automaticamente ficheiros de anotação no formato desejado.

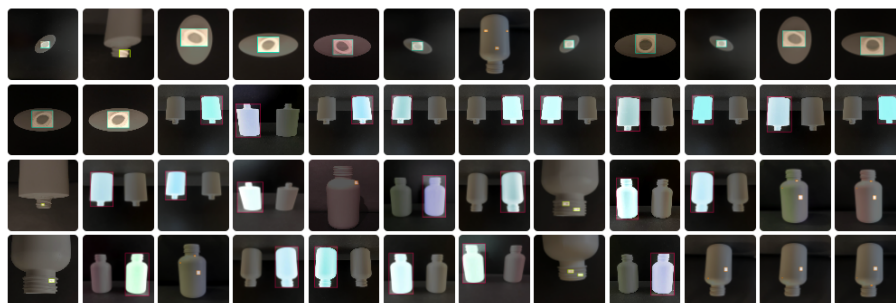


Figura B.4: Exemplos de embalagens com as *bounding boxes*

Formato de Anotação (Text File (TXT)):

Cada ficheiro de anotação foi representado no formato TXT composto por:

`class_i`, `x_center`, `y_center`, `width` e `height`:

- `class_i`: Identificador da classe do defeito;
- `x_center` e `y_center`: Coordenadas normalizadas do centro da bounding box;
- `width` e `height`: Largura e altura da bounding box normalizadas.

Tamanho do Dataset Final

Após a aplicação das técnicas de *Data Augmentation*, o número de imagens disponíveis para cada tipo de defeito foram as seguintes:

- *Black spots*: 374 imagens;
- *Color Variation*: 367 imagens;
- *Deformed Thread*: 368 imagens;
- *Oval Neck*: 372 imagens;
- *No Defect*: 371 imagens.

Totalizando em 1.852 imagens.

Apêndice C

Questionário

Questionário de Avaliação da Aplicação de Realidade Aumentada para Inspeção de Qualidade

Este questionário visa recolher a sua opinião sobre a utilização do protótipo de Realidade Aumentada desenvolvido no âmbito de um projeto académico de mestrado. O protótipo foi concebido para apoiar a inspeção de qualidade de embalagens plásticas, permitindo ao operador visualizar, no seu campo de visão, informações complementares que facilitam a validação de potenciais defeitos.

As respostas recolhidas são confidenciais, anónimas e utilizadas exclusivamente para fins académicos. Antes de responder, poderá consultar os seguintes documentos que asseguram a proteção da sua informação:

- [Política de Privacidade](#)
- [Termos e Condições](#)

Ao avançar, declara que leu e compreendeu os documentos acima referidos e aceita participar voluntariamente neste estudo.

* Indicates required question

Perfil do Utilizador

Já teve alguma experiência prévia com óculos de realidade aumentada? *

- Sim
- Não

Usabilidade

O processo de configuração inicial foi claro e simples. *

1 2 3 4 5

Discordo totalmente Concordo totalmente

Os alertas visuais e sonoros foram claros e fáceis de perceber.

1 2 3 4 5

Discordo totalmente Concordo totalmente

A interação com mãos virtuais foi intuitiva. *

1 2 3 4 5

Discordo totalmente Concordo totalmente

A utilização dos controladores físicos foi precisa. *

1 2 3 4 5

Discordo totalmente Concordo totalmente

Foi fácil compreender como validar ou rejeitar defeitos. *

1 2 3 4 5

Discordo totalmente Concordo totalmente

O zoom disponibilizado na validação dos defeitos foi útil para uma melhor observação dos mesmos? *

1 2 3 4 5

Discordo totalmente Concordo totalmente

O sistema respondeu de forma rápida às minhas ações. *

1 2 3 4 5

Discordo totalmente Concordo totalmente

As sobreposições visuais ao seu ambiente não interferiram com a execução de outras tarefas em simultâneo. *

1 2 3 4 5

Discordo totalmente Concordo totalmente

Utilidade Percebida

Considero que a aplicação pode ajudar a detetar defeitos de forma mais eficiente. *

1 2 3 4 5

Discordo totalmente Concordo totalmente

Considero que o sistema pode reduzir erros em inspeções de qualidade. *

1 2 3 4 5

Discordo totalmente Concordo totalmente

Considero que o protótipo contribui para tornar a inspeção mais rápida. *

1 2 3 4 5

Discordo totalmente Concordo totalmente

Considero que este tipo de tecnologia pode ser útil numa linha de produção real. *

1 2 3 4 5

Discordo totalmente Concordo totalmente

Conforto

Senti-me confortável a usar os óculos durante a tarefa. *

1 2 3 4 5

Discordo totalmente Concordo totalmente

Os alertas não foram demasiado intrusivos. *

1 2 3 4 5

Discordo totalmente Concordo totalmente

A interação com os controladores físicos foi confortável *

1 2 3 4 5

Discordo totalmente Concordo totalmente

A interação com as mãos virtuais não foi cansativa. *

1 2 3 4 5

Discordo totalmente Concordo totalmente

Senti fadiga visual ou desconforto após a utilização. *

1 2 3 4 5

Discordo totalmente Concordo totalmente

Preferências

Qual o tipo de alerta que preferiu? *

- Visual
- Sonoro
- Automático (adaptativo ao ruído)

Qual o modo de inspeção que considerou mais adequado? *

Simples (uma decisão global)

Complexo (defeito a defeito)

Automático (consoante quantidade de defeitos)

Feedback Qualitativo

O que mais gostou na aplicação? *

Your answer _____

O que menos gostou ou acha que deve ser melhorado? *

Your answer _____

Outras observações ou sugestões.

Your answer _____

[Get link](#)

Figura C.1: Questionário de avaliação