

Robô Q.bo One autónomo com reconhecimento facial e acompanhamento de movimento humano



RELATÓRIO FINAL

Francisco Conceição al73819 | Laboratório de projeto em engenharia informática |
2 de junho de 2023

Conteúdo

Introdução	3
Objetivos	3
Robô Q.BO One	4
Aplicação desenvolvida	5
Reconhecimento de voz	5
Fala do Robô	5
Movimento do robô conforme a posição da face	5
Guardar novas faces no modelo	6
Treinar o modelo de reconhecimento	6
Reconhecimento facial	6
Testes realizados	8
Problemas e Alternativas	11
Conclusão	12

Introdução

Este projeto consiste na participação do robô Q.bo One com sistemas de inteligência artificial para desenvolver um robô autónomo capaz de seguir o movimento de pessoas em ambientes movimentados mantendo-as no seu campo de visão. A programação baseia-se num microcontrolador baseado em Arduino, que será responsável pelo controlo dos movimentos do robô. A análise das imagens será feita através da biblioteca OpenCV que permitirá a identificação de pessoas através de técnicas de reconhecimento facial.

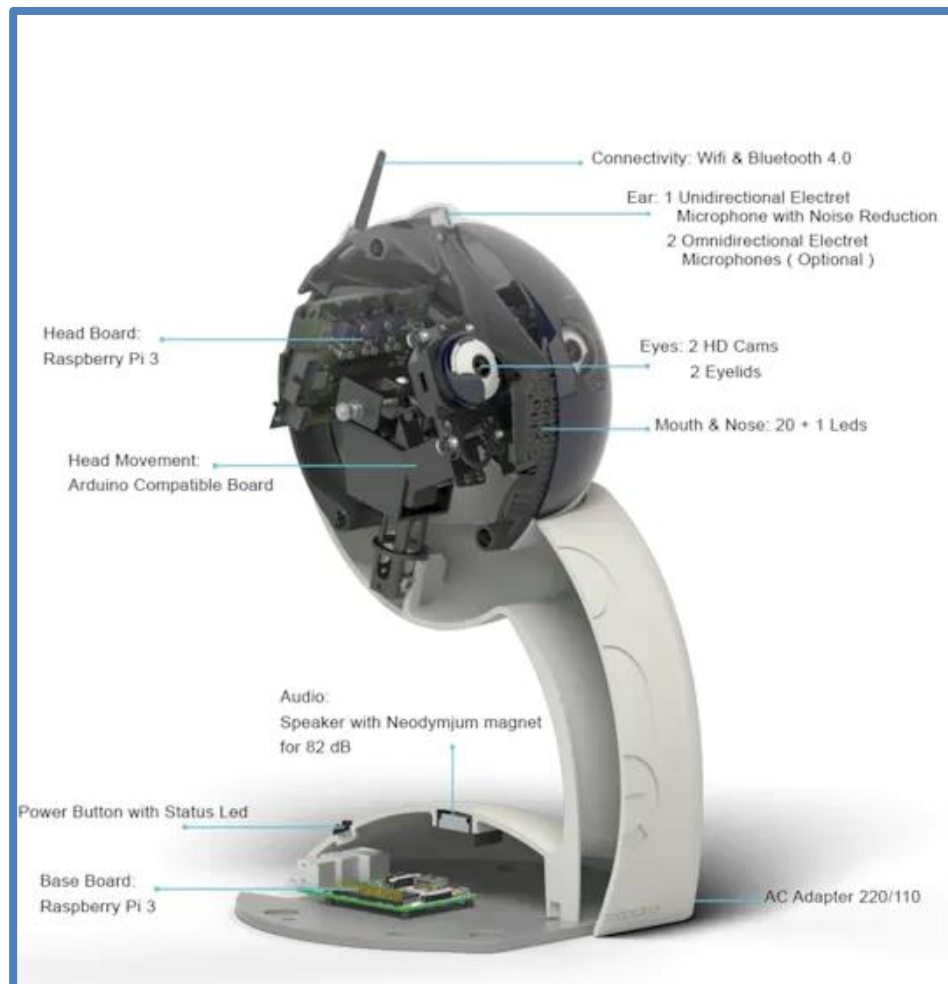
Objetivos

O objetivo principal deste projeto é desenvolver uma aplicação baseada em técnicas de inteligência Artificial e integrá-la no robô Q.bo One para torná-lo capaz de realizar rastreamento facial de indivíduos em ambientes com grande movimentação, mantendo-os em seu campo de visão e identificando-os por meio de reconhecimento facial, além de desenvolver a sua interação mais autónoma. Como objetivo mais específicos temos:

- Levantamento do estado da arte;
- Criar a aplicação baseada na biblioteca OpenCV para permitir a identificação de indivíduos por meio de técnicas de reconhecimento facial;
- Utilizar o microcontrolador baseado em Arduino para controlar os movimentos do robô e permitir o acompanhamento de indivíduos rastreados;
- Integrar a aplicação ao robô Q.bo One;

Robô Q.BO One

O Q.bo One é um robô open source que usa uma combinação de Raspberry Pi e Arduino e pode ser programado em Scratch, Python e outras linguagens. O robô possui dois olhos ambos contêm uma câmara com os quais pode ver, reconhecer, rastrear faces, capturar fotos e ativar chamadas de vídeo. Contém também um microfone, um alto-falante, e uma boca LEDs que fornece sincronização com o áudio.



A cabeça do Q.bo tem dois servos digitais para controlar o movimento para cima/para baixo e esquerda/direita. A parte superior da cabeça também possui três sensores de toque capacitivos que permitem que o Q.bo reaja adequadamente ao toque. O sistema operativo do robô é o Raspbian OS e o robô contém diversos sistemas de inteligência artificial para assistência virtual como o Google Assistant, o IBM Watson ou até mesmo o Mycroft.

Aplicação desenvolvida

Neste projeto, foi desenvolvida uma aplicação para o robô apresentado em python3.5 aonde é possível:

- O robô falar em português com saída nos speakers;
- Detetar comandos de voz através do microfone;
- Reconhecer faces;
- Guardar novas faces no modelo;
- Treinar o modelo de reconhecimento;
- Movimentar conforme a posição da face;
- Imitar o que a pessoa diz;
- Dizer que horas são;
- Desligar através de comando de voz;

Reconhecimento de voz

Para reconhecimento de voz foi construído uma função que transforma a informação fornecida por voz para texto usando o microfone do robô, usando a biblioteca SpeechRecognition, biblioteca que reconhece a fala através de diversos engines e APIs, online e offline

Fala do Robô

Para a comunicação do robô foi utilizado a biblioteca *gtts* que transforma texto em fala através da função construída, é uma biblioteca em Python e uma ferramenta CLI para interagir com a API que converte o texto em fala do Google Tradutor. Na função construída esta grava os dados num ficheiro mp3, que através da biblioteca os é capaz de reproduzir este ficheiro usando o comando `mpg123`, reprodutor de áudio para sistemas Linux.

Movimento do robô conforme a posição da face

Para o movimento do robô conforme a posição da face foi utilizado o OpenCV para a deteção de um rosto, e foi utilizado para o movimento o código controlador do robô já feito pela empresa, e a cada 2 segundos é calculado a posição da face da pessoa e conforme essa posição, o robô movimenta-se com uma determinada direção e intensidade. Existem quatro funções utilizadas para o movimento, uma para cada direção, ou seja, `CamDown` para baixo, `CamUp` para cima, `CamLeft` para a esquerda e `CamRight` para a direita, cada função obedece a 3 parâmetros, a distância, a velocidade que é utilizada para o movimento e o controlador para efetuar o movimento do servo, a equação utilizada para calcular a coordenada X ou Y é uma subtração entre o X ou Y antigo e a distância.

Guardar novas faces no modelo

O programa inicia solicitando ao utilizador que forneça o seu nome para dar início ao processo de registo. Em seguida, a câmara é ativada para captar imagens durante os próximos 200 frames. Durante esse período, o robô utiliza técnicas de deteção facial para identificar a presença de um rosto.

Ao detetar um rosto, o robô realiza o recorte da região facial e armazena essas imagens numa pasta específica, utilizando o nome fornecido pelo utilizador como identificador. Essas 100 imagens recortadas, capturadas ao longo dos 200 frames, são preservadas e organizadas nessa pasta.

Esse processo permite que o programa registre novas pessoas por meio da captura de múltiplas imagens faciais, proporcionando uma ampla variedade de poses e expressões para melhorar a precisão do sistema de reconhecimento facial.

Além do recorte e armazenamento das imagens, é importante ressaltar que o programa pode realizar outras etapas no processo de registo, como pré-processamento das imagens para garantir uma melhor qualidade, aplicação de técnicas de normalização e padronização. Essa abordagem detalhada e sistemática do registo de novas pessoas no modelo visa criar uma base de dados robusta e diversificada, o que contribui para um melhor desempenho do sistema de reconhecimento facial em termos de precisão e confiabilidade.

Treinar o modelo de reconhecimento

Para treinar o modelo de reconhecimento, o robô abre cada ficheiro correspondente a uma pessoa e guarda as 100 fotos dessa mesma pessoa num array junto com o id correspondente à pessoa, depois treina o modelo e guarda-o como ficheiro .yml junto com o projeto.

Reconhecimento facial

Para o reconhecimento facial foi utilizado o algoritmo já criado do OpenCV chamado LPBH (Local Binary Pattern Histograms), que é o algoritmo mais simples de reconhecimento facial, e possui excelentes resultados. O LPBH é uma variação do algoritmo LBP (Local Binary Patterns), que é um método popular para a descrição de texturas em imagens. No caso do LPBH, a técnica é adaptada para a tarefa de reconhecimento facial, capturando padrões locais de textura na imagem do rosto.

A ideia central do algoritmo LPBH é representar cada pixel da imagem da face como um número binário, comparando-o com seus vizinhos próximos. Essa comparação resulta em uma sequência de 0s e 1s, que é convertida em um número decimal. Esses números decimais são então usados para construir um histograma, que representa a distribuição dos padrões locais na imagem do rosto.

Durante a fase de treinamento do algoritmo, várias imagens de rostos são utilizadas para criar um modelo de referência que representa as características faciais de cada indivíduo. Esse modelo é construído a partir dos histogramas dos padrões locais das regiões do rosto. Na fase de reconhecimento, o algoritmo LPBH compara o histograma do padrão local extraído da imagem do rosto a ser reconhecido com os modelos de referência previamente armazenados. A similaridade entre os histogramas é medida e o rosto é reconhecido com base nessa comparação.

No programa para reconhecer a pessoa é aberto o modelo já treinado pelo programa que treina e guarda, e é criado um array com o nome das pessoas que se encontram no modelo conforme o nome das pastas criado, depois com a camara é identificada a face da pessoa, e são recortados 10 frames com a face do utilizador aonde o modelo reconhece com esses frames retornando um array com os 10 resultados feitos, é calculado a moda desse array para assim identificar a pessoa.

Testes realizados

Foram realizados testes a nove pessoas voluntárias para testar se a deteção e o reconhecimento funcionavam em diferentes pessoas. Antes de ser realizado o teste, foi dado um termo de consentimento livre e esclarecido, em que as pessoas assinavam para dar o consentimento a fazer os testes com os dados respetivos, visto que guardaria fotos das faces das pessoas no robô.

Estes são os dados de cada participante, 11% são do género feminino e 89% dos participantes são do género masculino, 22% têm nacionalidade brasileira e 78% têm nacionalidade portuguesa. Na tabela conseguimos observar as diversas idades de cada participante.

Idades	21	49	24	49	50	23	34	58	59
---------------	----	----	----	----	----	----	----	----	----



Para além desses dados, os voluntários responderam a um pequeno questionário sobre a experiência e dar a sua opinião e avaliação sobre o projeto, e alguns dos resultados foram os seguintes:

Como avalia a experiência geral do projeto?

Experiência muito boa, o robot tem um bom feedback de frases
Muito interessante, parece um trabalho bem complexo
Experiência interessante, com um bom reconhecimento de imagem. A deteção de voz e a tradução de texto para voz e de voz para texto funciona bem
Projeto muito inovador
Desafiante e com expectativas para um futuro mais seguro
No ponto de vista da experiência, a exploração é interessante

Como descreve a experiência em uma ou mais palavras?

Interessante com potencial
Incrível
Setup um pouco lento, mas funciona bastante bem
Interessante e Divertida
Boa experiência no âmbito da robótica
Desafiante
Boa experiência

Há algum comentário ou sugestão que possa nos ajudar a melhorar a experiência do utilizador e o robô?

Receber um feedback de quando o robot está a ouvir e podemos falar para evitar dizer mais que uma vez
Me parece um excelente trabalho
Resultados bastantes satisfatórios, mas recomendaria a utilização de algoritmos mais dinâmicos para reconhecimento de faces
Maior velocidade de processamento para o robô
Melhorar o diálogo com o robô
Melhorar o fluxo da conversa com o assistente

Também foi pedido aos participantes para avaliarem a experiência de zero a cinco e este foi o resultado:



Assim com os resultados obtidos dos participantes, posso concluir que o projeto correu bem e foi o que esperava, e graças aos testes e às sugestões dos participantes consegui melhorar o meu projeto.

Problemas e Alternativas

Ao longo deste projeto, deparei-me com diversos problemas que vão para além do que consigo resolver. Alguns desses problemas incluíram a limitação de velocidade de processamento do robô, devido à capacidade limitada da Raspberry Pi em controlar todas as operações. Por exemplo, durante a detecção de faces em movimento, a câmara do robô era capaz de alcançar apenas 3 a 5 frames por segundo (FPS), o que dificultava o acompanhamento adequado de movimentos rápidos por parte do robô.

Outro desafio significativo foi a versão desatualizada do sistema operativo, o que impossibilitou a instalação da biblioteca "face_recognition", uma biblioteca muito mais eficaz e rápida em comparação ao algoritmo alternativo implementado. Solucionar esse problema exigiu um investimento considerável de tempo no projeto.

Além disso, encontrei dificuldades relacionadas aos arquivos de controlo do robô, que estavam escritos em Python 2.3, uma linguagem que caiu em desuso. Para superar essa questão, foi necessário realizar a conversão de todos os arquivos do robô de Python 2 para Python 3.5, a fim de utilizá-los na minha aplicação.

Outro obstáculo foi a escassez de documentação disponível sobre o robô. A empresa responsável pelo desenvolvimento do robô disponibilizava apenas um fórum no seu site com alguns tópicos, e os próprios arquivos do robô em Python 2 possuíam poucos comentários, dificultando a compreensão e o suporte adequado.

Em suma, enfrentei uma série de desafios ao longo deste projeto. A limitação de velocidade de processamento, a versão desatualizada do sistema operativo, a incompatibilidade dos arquivos de controle e a escassez de documentação foram fatores que demandaram tempo e esforço adicionais para encontrar soluções adequadas. No entanto, apesar desses contratemplos, essas dificuldades também proporcionaram oportunidades para desenvolver habilidades de resolução de problemas e aprimorar a minha capacidade de adaptação a circunstâncias adversas.

Conclusão

Concluindo, posso classificar este projeto como bem-sucedido, visto que alcancei todos os objetivos indicados no início do relatório apesar dos problemas enfrentados. Adquiri diversos conhecimentos em relação aos sistemas Linux, à biblioteca OpenCV e à área visão por computador.

A integração da aplicação no robô Q.bo One representa um passo significativo no desenvolvimento de robôs autônomos e interativos. O projeto possibilitou explorar e aplicar conceitos de inteligência artificial num cenário real, contribuindo para o avanço da área de robótica e interação humano-robô.

Em suma, o projeto final de Engenharia Informática, com o tema de desenvolvimento de um robô autônomo capaz de realizar o rastreamento facial de pessoas em ambientes movimentados, demonstrou a aplicação prática e eficácia das técnicas de inteligência artificial, bem como a integração de hardware e software para alcançar os objetivos propostos. Os resultados obtidos abrem portas para novas possibilidades no campo da robótica e mostram o potencial de avanços futuros na interação humano-robô.