



Cofinanciado pela
União Europeia

Financiado pela União Europeia. Os pontos de vista e as opiniões expressas são as do(s) autor(es) e não refletem necessariamente a posição da União Europeia ou da Agência de Execução Europeia da Educação e da Cultura (EACEA). Nem a União Europeia nem a EACEA podem ser tidos como responsáveis por essas opiniões.

IA e Tecnologia de Visão por Computador



Introduzir as 5 Grandes Ideias da Inteligência Artificial
utilizando a Internet das Coisas no ensino STEM

T2.4 Conceção de projetos IoT e desenvolvimento de recursos

29.08.2023 | EMPHASYS CENTRE
NÚMERO DO PROJECTO: 2022-1-FR01-KA220-SCH-000085611

Projetos IoT AI4STEM

Projeto: IA e Tecnologia de Visão por Computador

Copyright

© Direitos de autor do Consórcio AI4STEM
2022-1-FR01-KA220-SCH-000085611
Todos os direitos reservados.



Projetos IoT AI4STEM Projeto: IA e Tecnologia de Visão por Computador © 2023 pelo [Consórcio AI4STEM](#) está licenciado sob [Atribuição-NãoComercial-CompartilhaIgual 4.0 Internacional](#)

Índice

1. introdução ao projeto	4
1.1 O âmbito do projeto	4
1.2 Os grupos-alvo	4
1.3 Objetivo do presente documento.....	4
2. Glossário da unidade.....	4
3. Introdução à "IA na Visão"	5
3.1 Descrição	5
3.2 Objectivos e resultados da aprendizagem	5
3.3 Duração prevista da unidade	5
3.4 Atividade 1 - Introduzir a grande ideia da perceção através da IoT:	6
3.4.1 Descrição	6
3.4.2 Hardware	6
3.4.3 Configuração	7
3.4.4 Experiência 1	13
3.5 Atividade 2: Introduzir a ideia de Representação e Raciocínio	14
3.5.1 Descrição	14
3.5.2 Hardware	14
3.5.3 Configuração	15
3.5.4 Exercício: Treinar o modelo de IA para reconhecer diferentes formas	17
3.6 Atividade 3: Introduzir a ideia de aprendizagem através do treino de um modelo para o reconhecimento de rostos.....	18
3.6.1 Descrição	18
3.6.2 Hardware	19
3.6.3 Configuração	19
3.6.4 Código	20
3.6.4 Exercício: Testar se o modelo de IA o reconhece	21
3.7 Atividade 4: Introduzir a ideia de Interação Natural através da integração de um modelo treinado numa aplicação de IA	22
3.7.1 Descrição	22
3.7.2 Hardware	22
3.7.3 Configuração	23
3.7.4 Exercício: Sistema de alarme	25

3.8 Atividade 5: Introduzir a ideia de impacto social.....	25
3.8.1 Descrição.....	25
3.8.2 Hardware	26
3.8.3 Configuração	26
3.8.4 Introduzir a ideia de Interação Natural treinando um modelo de Reconhecimento de Rosto .	28
3.9 Material e recursos adicionais	28

1. introdução ao projeto

Para este projeto, serão utilizados o micro:bit e o HuskyLens para realizar diferentes actividades. O HuskyLens é um sensor de visão inteligente ou um módulo de câmara que combina inteligência artificial (IA) e tecnologia de visão por computador para realizar várias tarefas, como **reconhecimento de objectos**, **reconhecimento de rostos**, **localização**, **reconhecimento de cores** e muito mais. A integração da inteligência artificial (IA) com a câmara HuskyLens e o micro:bit é um projeto empolgante que permite criar aplicações interactivas e inteligentes.

1.1 O âmbito do projeto

O objetivo do projeto é utilizar o micro:bit e o HuskyLens para criar um modelo de Inteligência Artificial. O modelo será **treinado** através da câmara HuskyLens e dará um **resultado** através da ferramenta micro:bit.

1.2 Os grupos-alvo

O projeto visa sobretudo o envolvimento direto de educadores, principalmente do ensino primário e secundário.

1.3 Objetivo do presente documento

O objetivo deste documento é demonstrar como treinar e testar um modelo de IA utilizando a câmara HuskyLens e apresentar o resultado no micro:bit.

2. Glossário da unidade

Palavra	Definição
Micro:bit	O micro:bit é uma placa de computador programável, de bolso, concebida para o ensino. Possui uma matriz de LED, vários sensores e um microcontrolador
HuskyLens	O HuskyLens é um módulo de sensor de visão que pode reconhecer objectos, rostos e gestos. Está equipado com uma câmara e uma capacidade de processamento integrada para executar tarefas relacionadas com a IA

3. Introdução à "IA na Visão"

3.1 Descrição

Esta unidade apresentará aos alunos o processo de formação e teste de um modelo de IA com a câmara HuskyLens. Aprenderão a instalar e a configurar a câmara HuskyLens para **tarefas de reconhecimento de imagem**, a treinar o modelo de IA para reconhecer **objectos ou padrões específicos** e, por fim, a ligar a câmara a um micro:bit para apresentar os **resultados** do modelo de IA. Ao longo da unidade, os alunos adquirem uma compreensão prática da visão por computador, da aprendizagem automática e de como integrar tecnologias de IA em projectos do mundo real.

3.2 Objectivos e resultados da aprendizagem

Após a conclusão bem-sucedida desta unidade, os alunos devem ser capazes de

- Compreensão da visão por computador
- Formação de modelos de IA
- Configuração da HuskyLens
- Integração do Micro:bit
- Resolução de problemas
- Aplicação prática
- Operar o HuskyLens
- Recolher dados
- Treinar modelos de IA
- Interagir com o Micro:bit

3.3 Duração prevista da unidade

Trata-se de um projeto bastante extenso que necessita de várias horas para abordar corretamente todos os aspectos incluídos. A duração que se segue é indicativa e pode variar consoante a idade e o nível dos alunos.

Atividade 1: 90 minutos

Atividade 2: 90 minutos

Atividade 3: 45 minutos

Atividade 4: 45 minutos

Atividade 5: 45 minutos

3.4 Atividade 1 - Introduzir a grande ideia da percepção através da IoT:

3.4.1 Descrição

Nesta aula, os alunos vão explorar o mundo da inteligência artificial e da visão por computador, a fim de explorar o método de percepção, através do **rastreo de objectos em direto**, uma funcionalidade possível graças às capacidades de ponta da câmara HuskyLens e da plataforma micro:bit.

Utilizando a câmara HuskyLens e a plataforma micro:bit, os alunos descobrirão como a tecnologia pode ser utilizada para obter informações em tempo real e fornecer informações ao utilizador. O aspeto do rastreo em tempo real não só demonstra as capacidades em tempo real do sistema, como também realça a natureza dinâmica da tecnologia na resposta ao seu ambiente.

Na era dos rápidos avanços tecnológicos, a fusão da Internet das Coisas (IoT) com tecnologias de processamento de imagem de ponta abriu caminho para aplicações inovadoras. Uma dessas inovações é a integração da HuskyLens, uma poderosa câmara inteligente, com o Micro:bit. Juntos, formam um sistema dinâmico para o seguimento de objectos em tempo real, revolucionando a forma como percebemos e interagimos com o que nos rodeia.

Quando os alunos utilizam o sistema HuskyLens e Micro:bit, não só adquirem experiência prática com ferramentas de ponta, como também participam na narrativa contínua do progresso tecnológico.

3.4.2 Hardware

- Micro:bit microcontrolador
- Extensor IO para micro:bit
- HuskyLens



3.4.3 Configuração

3.4.3.1 Cablagem

- Ligar o micro:bit com o extensor IO para micro:bit.
- Em seguida, utilize a tabela seguinte para ligar o extensor IO para micro:bit à câmara HuskyLens.

Extensor IO para micro:bit PORT	HuskyLens PORT
SDA	T
SCL	R
GND	-
3V3	+

3.4.3.2 Atualização do firmware

Antes de utilizar este produto, é altamente recomendável instalar/atualizar o firmware do HuskyLens para acessar os recursos mais recentes e garantir a estabilidade ideal. Sugere-se a utilização do HuskyLens Uploader no Windows 10 para o carregamento do firmware, uma vez que oferece uma interface gráfica do utilizador (GUI) e uma operação fácil de utilizar.

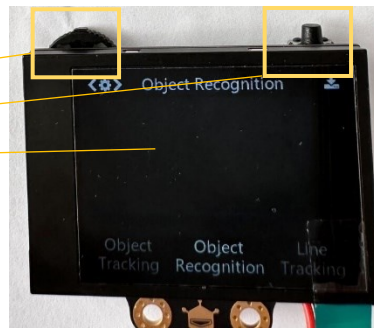
Utilize a seguinte ligação para navegar para o sítio Web do DFROBOT e encontrar a ligação para instalar o firmware: https://wiki.dfrobot.com/HUSKYLENS_V1.0_SKU_SEN0305_SEN0336

Depois de visitar o sítio Web acima mencionado, navegue até à secção "Update Firmware" (Atualizar firmware). Se tiver Windows, vá para a secção 4.1 e, se tiver Linux ou Mac, vá para a secção 4.2 e siga os passos para garantir que o firmware mais recente é atualizado.

Depois de seguir os passos apropriados, o **controlador USB para UART do HuskyLens Uploader** e o **firmware mais recente** serão instalados.

3.4.3.3 Descrição geral do quadro

- Botão de função
- Botão de
- Ecrã



aprendizagem

O HuskyLens está equipado com dois botões: o botão de função e o botão de aprendizagem. As suas funções principais são descritas de seguida:

- Utilize o botão de função para alternar entre diferentes funções, rodando-o para a esquerda ou para a direita.
- Prima brevemente o botão de aprendizagem para ensinar o dispositivo sobre um objeto específico. Manter premido o botão de aprendizagem permite a aprendizagem contínua do objeto a partir de vários ângulos e distâncias. Se o HuskyLens já tiver aprendido o objeto, uma breve pressão no botão de aprendizagem fará com que se esqueça.
- Para aceder ao menu de segundo nível (definições de parâmetros) dentro da função atual, prima e mantenha premido o botão de função. Utilize as rotações para a esquerda ou para a direita do botão ou uma breve pressão para ajustar os parâmetros relacionados.

HuskyCapacidades das lentes:

- Reconhecimento de rostos
- Seguimento de objectos
- Reconhecimento de objectos
- Seguimento de linha
- Reconhecimento de cores
- Reconhecimento de etiquetas
- Classificação de objectos

Versão mais recente do software:

- Navegar para Definições gerais -> Versão

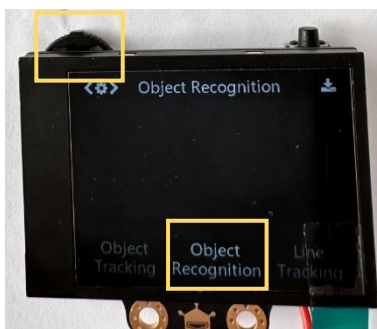
- Consulte o seguinte endereço para obter instruções sobre como atualizar para a versão mais recente (caso exista):

https://wiki.dfrobot.com/HUSKYLENS_V1.0_SKU_SEN0305_SEN0336

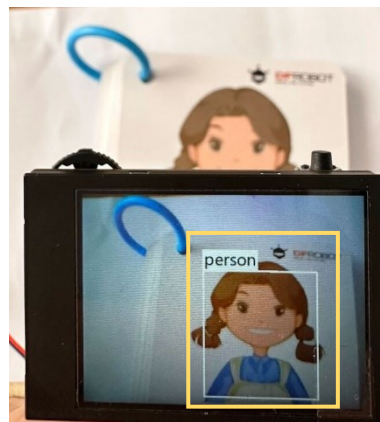
3.4.3.4 Começar

Depois de estabelecer a ligação entre o micro:bit e a HuskyLens, o micro:bit deve ser ligado ao PC através de um cabo. Uma vez feita a ligação, a câmara HuskyLens, juntamente com o micro:bit, liga-se.

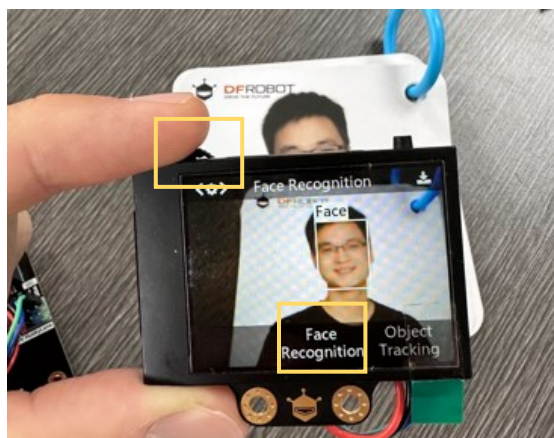
Em seguida, o aluno deve navegar para a secção Reconhecimento facial com a ajuda do botão que se encontra no canto superior esquerdo da câmara HuskyLens. Mova-o para a direita até encontrar a secção Reconhecimento facial.



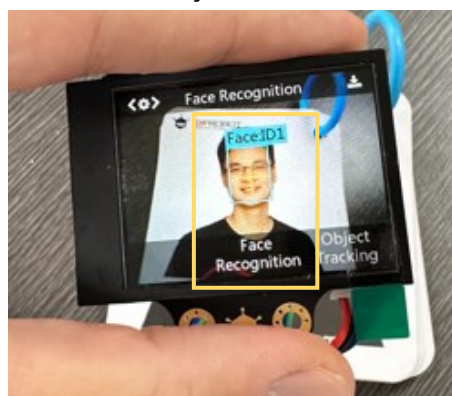
Depois de seleccionar o Reconhecimento de Rosto, aponte a câmara para um objeto ou pessoa para ver o resultado. A câmara HuskyLens pode detetar objectos através de uma entrada em tempo real e pode dar um resultado ao utilizador. Isto representa a perceção deste sistema.



Para a definição de reconhecimento de rosto, tem a opção de guardar mais do que 1 rosto. Em primeiro lugar, seleccione a opção de reconhecimento de rosto e mantenha premido o botão de função, como se mostra abaixo.



Em seguida, clique na opção Aprender Múltiplos e depois em Guardar e regressar. Agora a função de aprendizagem múltipla está activada e podem ser digitalizados e guardados vários rostos na câmara HuskyLens.



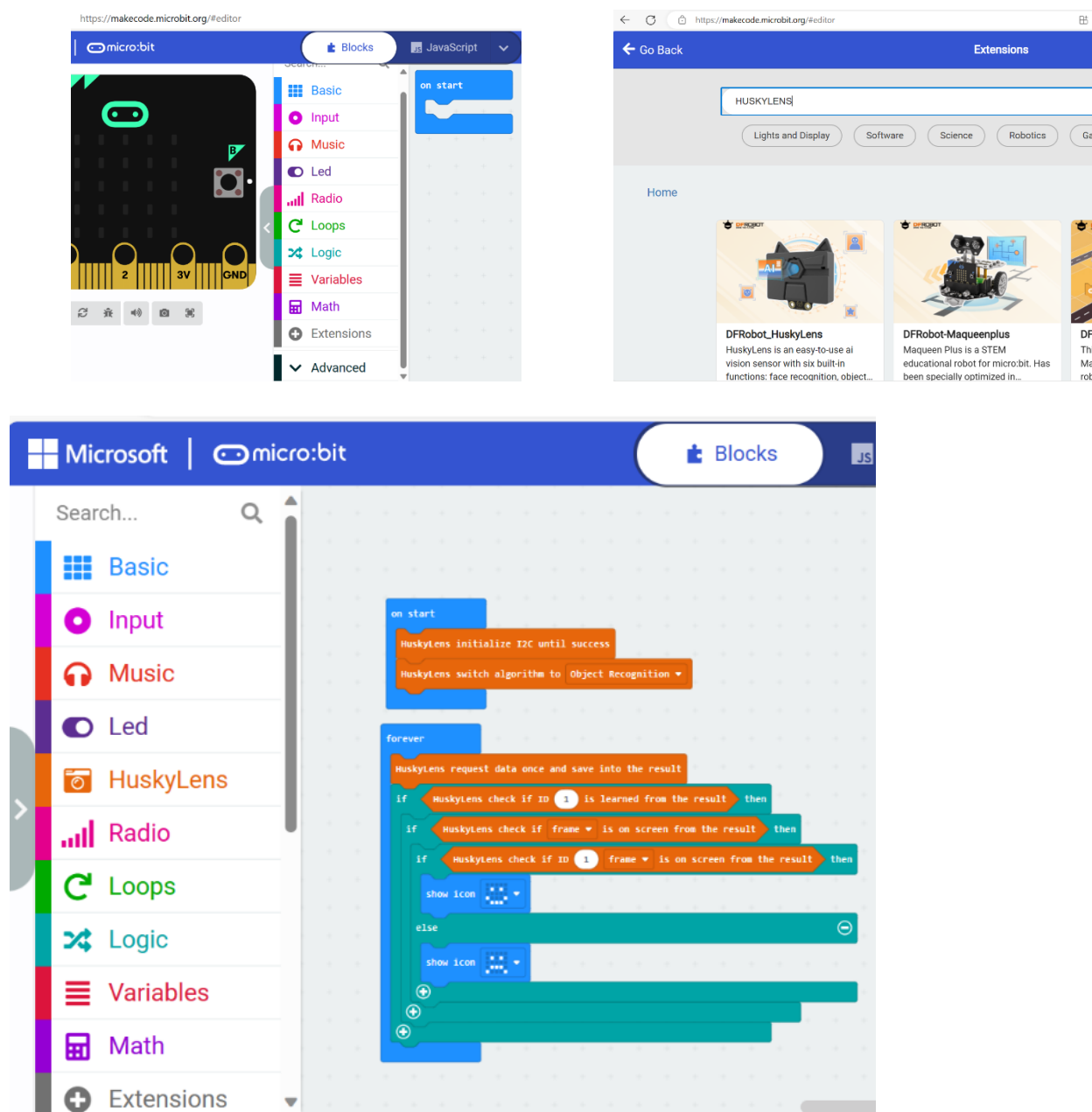
3.4.3.5 Código

Depois de estabelecerem a ligação entre o micro:bit e o HuskyLens, os alunos devem escrever código que forneça instruções à câmara HuskyLens, permitindo a criação de um sistema de reconhecimento de objectos. Este código guiará a câmara na captura e processamento de objectos para os reconhecer eficazmente.

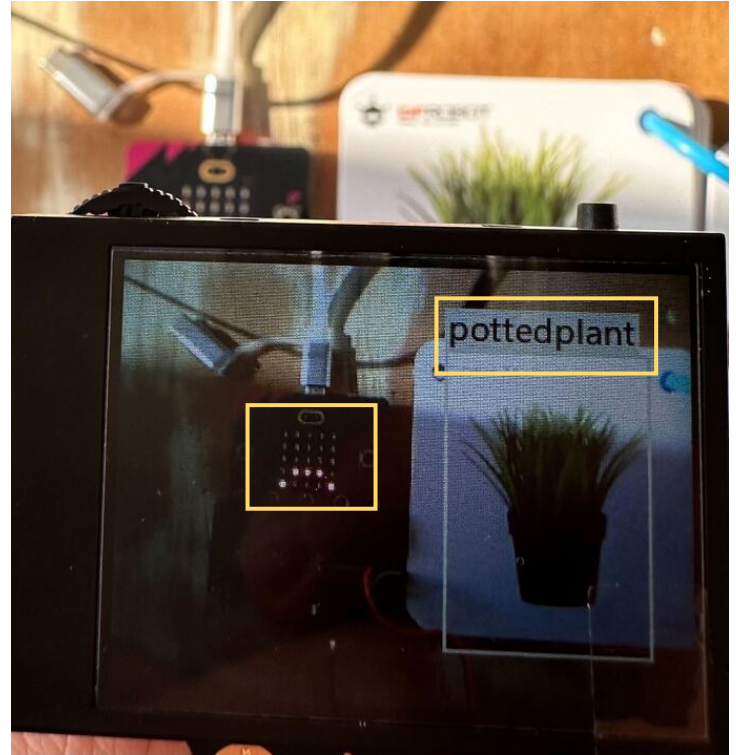
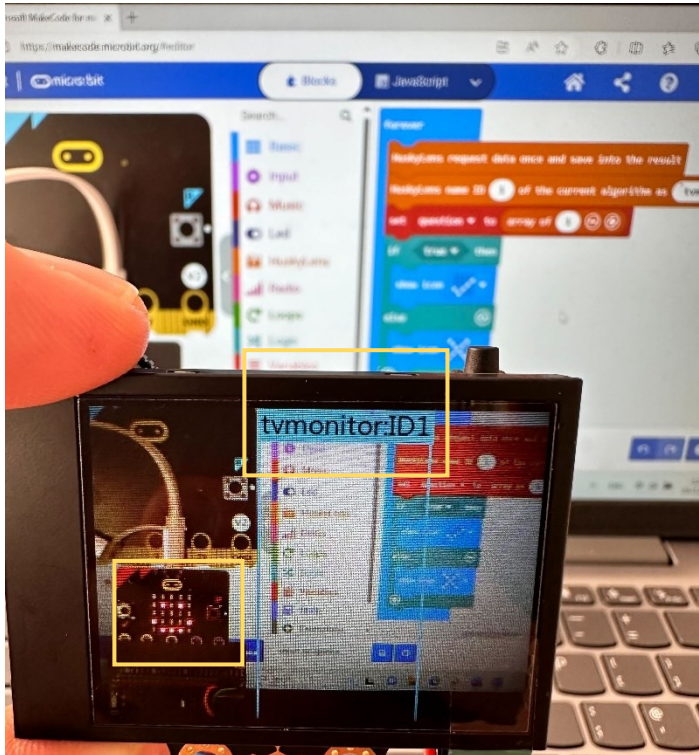
O aluno deve utilizar o software do seguinte sítio Web para escrever o código:
<https://makecode.microbit.org/>

Para utilizar o código abaixo, deve ser adicionado um bloco HuskyLens. Para adicionar este bloco, seleccione o bloco "Extensions" e, em seguida, procure **DFRobot_HuskyLens** e

seleccione-o. Em seguida, o bloco HuskyLens será automaticamente adicionado à secção de blocos e pode então prosseguir com o código abaixo.



Resultado:



Explicação do código:

- A câmara HuskyLens usa o protocolo I2C. Prepara o micro:bit para falar com a câmara.
- Para utilizar a funcionalidade HuskyLens, comece por adicionar a extensão HuskyLens à sua paleta de códigos. Com esta extensão, pode criar código que muda facilmente o algoritmo para o modo de reconhecimento facial.
- Estamos a dizer ao HuskyLens para trabalhar no modo de reconhecimento de objectos. Isto significa que tentará reconhecer e diferenciar objectos.
- A parte seguinte do código configura um ciclo contínuo que funcionará enquanto o micro:bit estiver ligado.
- Em seguida, o micro:bit pede ao HuskyLens para fornecer informações sobre o que está a ver no momento.
- O programa verifica se o HuskyLens aprendeu algum objeto. O número 1 refere-se à etiqueta do objeto e, se tiver aprendido alguma coisa, o código dentro das chavetas será executado.
- O HuskyLens está a detetar um objeto na sua vista.
- Em seguida, verifica se um objeto específico (neste caso, o objeto 1) foi detectado.

- Se o HuskyLens reconhecer o objeto e este aparecer como um bloco, o micro:bit apresenta um ícone de cara feliz.
- Se o objeto não for reconhecido como um bloco, é apresentado um ícone de cara triste.

3.4.4 Experiência 1

Neste exercício, os alunos terão a oportunidade de aplicar a tecnologia do HuskyLens e do micro:bit para **reconhecer diferentes formas**. Este exercício desafia a sua compreensão do reconhecimento de objectos e permite-lhes explorar as aplicações práticas desta tecnologia.

O principal objetivo deste exercício é que os alunos criem um programa que utilize a câmara HuskyLens para identificar e distinguir diferentes formas, tais como um círculo, um triângulo, um retângulo e, em seguida, apresentar os emojis ou símbolos correspondentes no ecrã do micro:bit.

3.5 Atividade 2: Introduzir a ideia de Representação e Raciocínio

3.5.1 Descrição

Nesta atividade, os alunos vão explorar o mundo da inteligência artificial e da visão por computador, a fim de explorar o método de reconhecimento facial. Esta atividade introduzirá a ideia de **treinar** um modelo de IA para realizar tarefas específicas ou tomar decisões inteligentes. A fase de treino é crucial para um modelo de IA porque é durante esta fase que o modelo **aprende** e se adapta à tarefa ou ao problema que foi concebido para resolver. A formação é essencialmente o processo de ensinar o modelo de IA expondo-o a um grande conjunto de dados, permitindo que o modelo aprenda padrões, correlações e regras a partir da informação fornecida.

Nesta atividade, é instalada uma câmara HuskyLens que permite ao modelo de IA recolher informações diretamente. Durante o processo de formação, o modelo de IA ajusta os seus parâmetros internos através de uma otimização iterativa, aumentando a sua capacidade de fazer previsões ou classificações precisas.

A plataforma micro:bit, que funciona como o cérebro da operação, torna-se a interface através da qual os alunos podem observar e interagir com o modelo de IA treinado. Este aspeto prático da atividade permite que os alunos não só compreendam os fundamentos teóricos da aprendizagem automática, mas também apreciem as implicações práticas da utilização desta tecnologia em cenários do mundo real.

Utilizando a câmara HuskyLens e a plataforma micro:bit, os alunos descobrirão como a tecnologia pode ser utilizada para treinar o modelo de IA com rostos humanos familiares. No final da atividade, os alunos não só compreendem os fundamentos do reconhecimento de rostos e objectos e do treino de IA, como também adquirem uma apreciação mais ampla do potencial e dos desafios da integração da IA na vida quotidiana.

3.5.2 Hardware

- Micro:bit
- Extensor IO para micro:bit
- HuskyLens
- Imagens de objectos



3.5.3 Configuração

3.5.3.1 Cablagem

- Ligar o micro:bit com o extensor IO para micro:bit.
- Em seguida, utilize a tabela seguinte para ligar o extensor IO para micro:bit à câmara HuskyLens.

Extensor IO para micro:bit PORT	HuskyLens PORT
SDA	T
SCL	R
GND	-
3V3	+

3.5.3.2 Começar

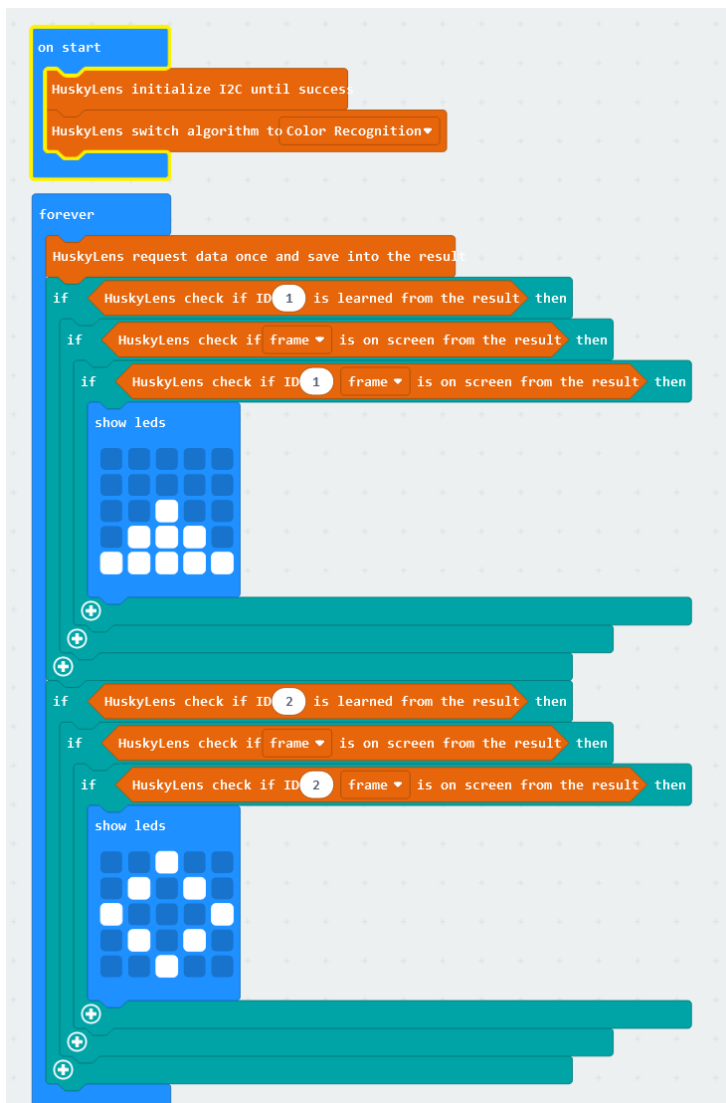
Depois de estabelecer a ligação entre o micro:bit e a HuskyLens, o micro:bit deve ser ligado ao PC através de um cabo. Uma vez feita a ligação, a câmara HuskyLens, juntamente com o micro:bit, liga-se.

Em seguida, o aluno deve navegar para a secção Reconhecimento facial com a ajuda do botão que se encontra no canto superior esquerdo da câmara HuskyLens. Mova-o para a direita até encontrar a secção Face Recognition (Reconhecimento de rostos).



3.5.3.3 Código

Depois de estabelecerem a ligação entre o micro:bit e o HuskyLens, os alunos devem escrever código que forneça instruções à câmara HuskyLens, permitindo a criação de um sistema de reconhecimento de cores. Este código guiará a câmara na captura e processamento de cores para reconhecer formas de forma eficaz.

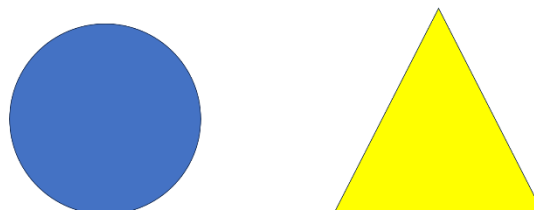
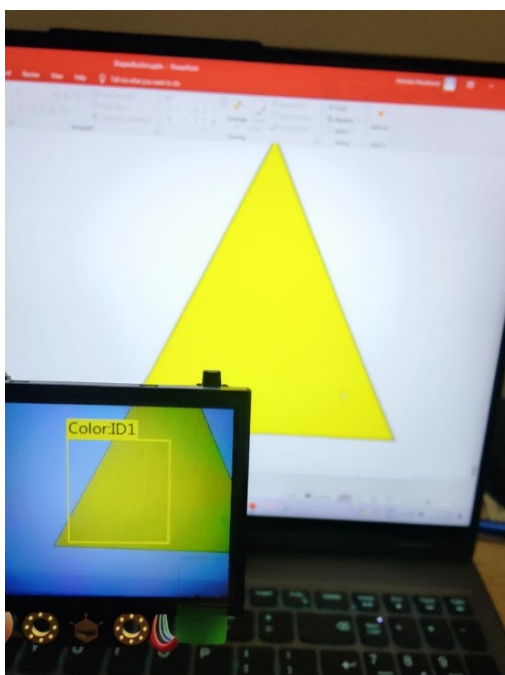


Explicação do código:

Para utilizar a funcionalidade HuskyLens, comece por adicionar a extensão HuskyLens à sua paleta de códigos. Com esta extensão, pode criar código que muda facilmente o algoritmo para o modo de reconhecimento de cores.

Inicializa a comunicação I2C entre o microcontrolador e o HuskyLens. Continuará a tentar até que a inicialização seja bem sucedida. Define o HuskyLens para usar o algoritmo de reconhecimento de cores. Solicita dados do HuskyLens uma vez e armazena-os no resultado. Assegura que a cor com ID 1 seja aprendida pelo HuskyLens. Se a cor ID 1 for reconhecida, é apresentado um padrão LED **triangular** no ecrã do microcontrolador.

Assegura que a cor com ID 2 é reconhecida pelo HuskyLens. Se a cor ID 2 for reconhecida, apresenta um padrão LED **circular** no ecrã do microcontrolador.



3.5.4 Exercício: Treinar o modelo de IA para reconhecer diferentes formas

Neste exercício, os alunos terão a oportunidade de aplicar a tecnologia do HuskyLens e do micro:bit para **reconhecer formas**. Este exercício desafia a sua compreensão da fase de treino do modelo de IA.

3.6 Atividade 3: Introduzir a ideia de aprendizagem através do treino de um modelo para o reconhecimento de rostos

3.6.1 Descrição

Nesta atividade, os alunos mergulharão na fase dinâmica que se segue à fase de formação do modelo de IA - a **fase de aprendizagem**. Esta fase marca a aplicação prática dos conhecimentos adquiridos pelo modelo e a avaliação da sua capacidade de fornecer resultados exactos em cenários em tempo real, especificamente no contexto do reconhecimento facial.

A fase de aprendizagem envolve colocar o modelo de IA treinado à prova, avaliar o seu desempenho e garantir que consegue identificar e responder corretamente aos rostos familiares com que foi treinado. O principal objetivo é determinar se o modelo é capaz de produzir o resultado certo, um passo fundamental para afirmar que o modelo de IA está a funcionar como pretendido.

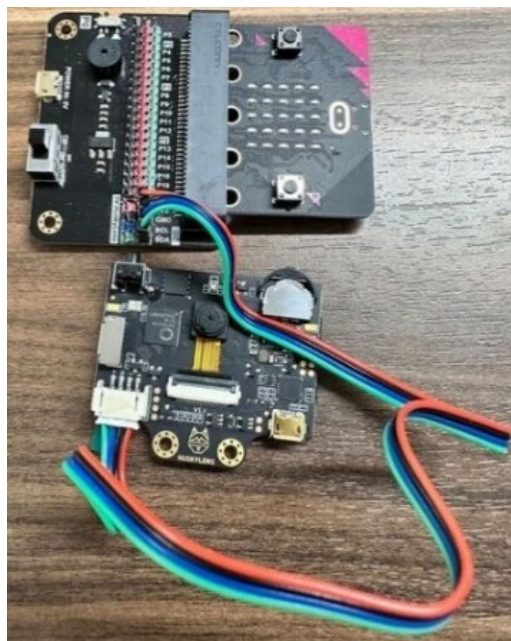
A plataforma micro:bit serve de interface através da qual os alunos interagem com o modelo de IA. Como utilizadores, os alunos têm a tarefa de verificar se o modelo reconhece corretamente o rosto treinado captado pela câmara HuskyLens. Este aspeto prático da atividade fornece uma demonstração tangível e prática de como os modelos de IA passam da aprendizagem teórica para a aplicação no mundo real.

O processo envolve a captura de imagens utilizando a câmara HuskyLens e permitindo que o modelo de IA analise e identifique os rostos nas imagens. O micro:bit funciona então como mecanismo de feedback, apresentando o resultado do processo de reconhecimento. Se o modelo de IA identificar com êxito o rosto treinado, isso serve como indicador de que a fase de aprendizagem foi eficaz e que o modelo está a funcionar corretamente.

Esta atividade não só reforça os conceitos técnicos relacionados com o reconhecimento facial e a IA, como também realça a importância da validação e dos testes reais no desenvolvimento e implementação de sistemas de IA. Os alunos adquirem conhecimentos sobre os desafios e as considerações envolvidas para garantir a fiabilidade e a precisão dos modelos de IA em aplicações práticas.

3.6.2 Hardware

- Micro:bit
- Extensor IO para micro:bit
- HuskyLens
- Imagens a cores

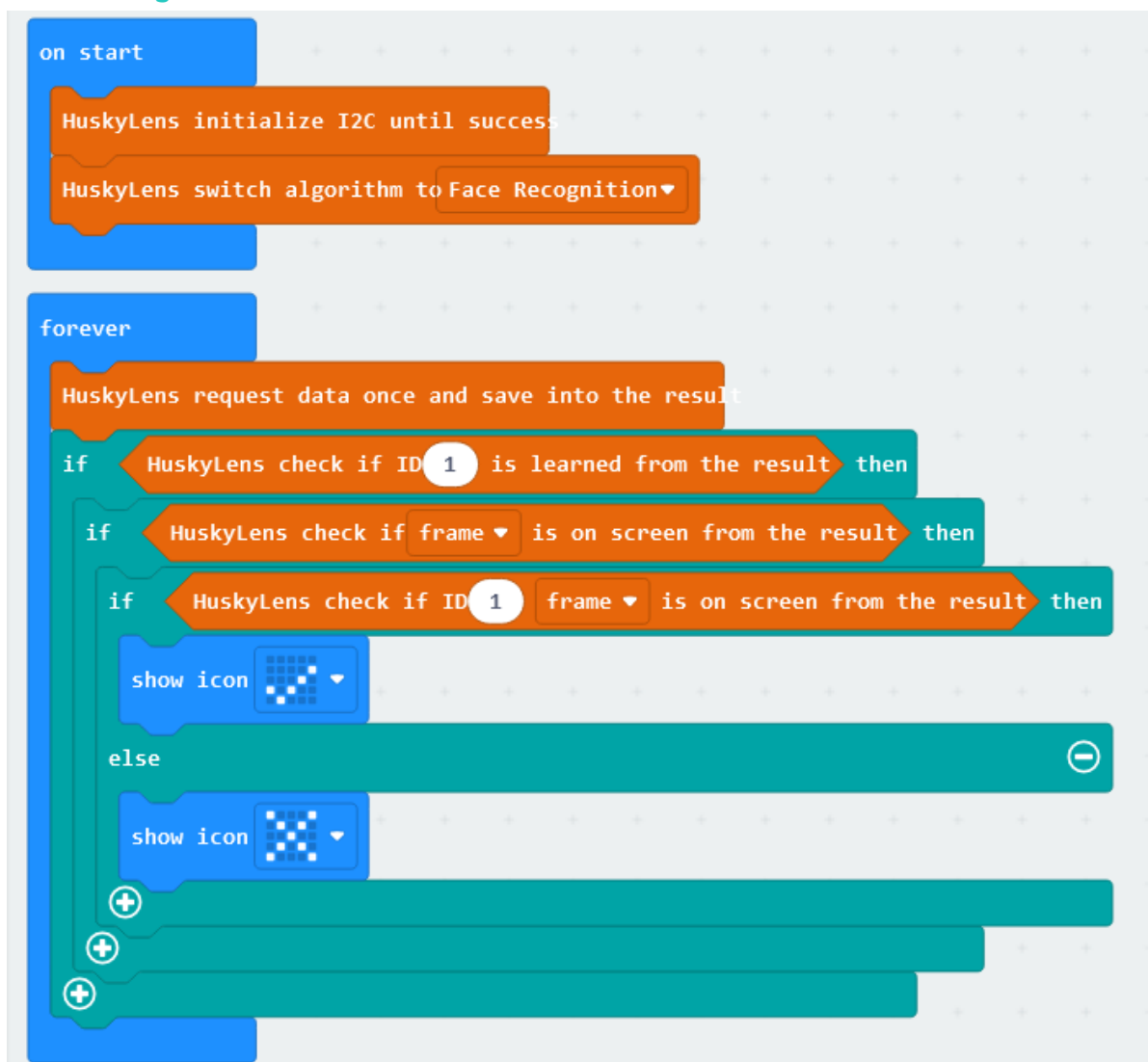


3.6.3 Configuração

- Ligar o micro:bit com o extensor IO para micro:bit.
- Em seguida, utilize a tabela seguinte para ligar o extensor IO para micro:bit à câmara HuskyLens.

Extensor IO para micro:bit PORT	HuskyLens PORT
SDA	T
SCL	R
GND	-
3V3	+

3.6.4 Código



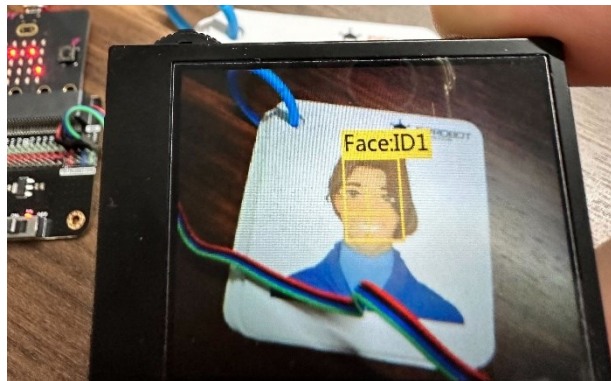
Explicação do código:

Para utilizar a funcionalidade HuskyLens, comece por adicionar a extensão HuskyLens à sua paleta de códigos. Com esta extensão, pode criar código que muda facilmente o algoritmo para o modo de reconhecimento facial.

Este código solicitará dados de entrada, exigindo que o aluno utilize a câmara HuskyLens para capturar o rosto de uma pessoa. O utilizador treinou o modelo de IA com um rosto específico na atividade anterior.

Quando a câmara detecta a cara reconhecida, o micro:bit responde com uma cara sorridente e alegre ou com um sinal de resposta correcta, enquanto uma cara aleatória ou não reconhecida resulta numa cara triste ou num sinal errado. Este processo envolvente faz a ponte entre a tecnologia e a interação humana, tornando-o uma experiência de aprendizagem fascinante.

- A câmara HuskyLens utiliza o protocolo I2C. Prepara o micro:bit para falar com a câmara.
- Estamos a dizer ao HuskyLens para trabalhar no modo de reconhecimento de cores. Isto significa que tentará reconhecer e diferenciar objectos com base nas suas cores.
- A parte seguinte do código configura um ciclo contínuo que irá correr enquanto o micro:bit estiver ligado.
- Em seguida, o micro:bit pede ao HuskyLens que forneça informações sobre o que está a ver no momento.
- O programa verifica se o HuskyLens aprendeu algum objeto. O número 1 refere-se à etiqueta do objeto e, se tiver aprendido alguma coisa, o código dentro das chavetas será executado.
- O HuskyLens está a detetar um objeto na sua vista.
- Em seguida, verifica se um objeto específico (neste caso, o objeto 1) foi detectado.
- Se o HuskyLens reconhecer o objeto e este aparecer como um bloco, o micro:bit apresenta um ícone de cara feliz.
- Se o objeto não for reconhecido como um bloco, é apresentado um ícone de cara triste.



3.6.4 Exercício: Testar se o modelo de IA o reconhece

Utilizar o sistema HuskyLens e micro:bit para treinar o modelo de IA com um novo rosto com o nome da pessoa e, em seguida, capturar o rosto da nova pessoa com a câmara HuskyLens para ver se consegue reconhecê-lo e se tem a etiqueta correcta para essa pessoa.

Este exercício testará se o utilizador pode treinar e testar o modelo de IA.

3.7 Atividade 4: Introduzir a ideia de Interação Natural através da integração de um modelo treinado numa aplicação de IA

3.7.1 Descrição

Nesta envolvente continuação do percurso de exploração da IA, os alunos passarão à **fase de teste**, um passo fundamental que se segue à fase de aprendizagem. A fase de teste serve como uma avaliação abrangente do desempenho do modelo de IA, avaliando a sua capacidade de generalizar e responder com precisão a vários casos. Através da lente do reconhecimento facial, utilizando a câmara HuskyLens e a plataforma micro:bit, os alunos irão aprofundar os meandros da fase de teste.

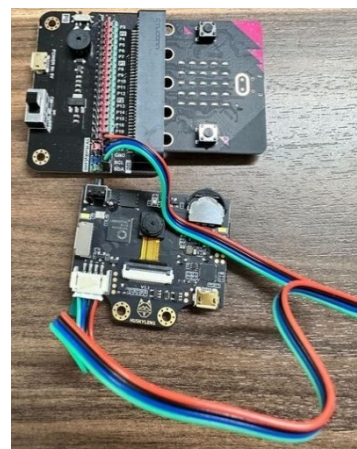
A fase de teste é o passo seguinte à fase de aprendizagem. Nesta fase, o modelo de IA é testado em vários casos para verificar se está a funcionar corretamente. Após esta fase, podem ser efectuados os ajustamentos adequados. A fase de teste é um ponto de controlo essencial no desenvolvimento de um modelo de IA, uma vez que simula cenários do mundo real e avalia a adaptabilidade e fiabilidade do modelo. Após a fase de aprendizagem, durante a qual o modelo se familiarizou com rostos específicos, a fase de teste introduz variabilidade, capturando rostos humanos familiares e desconhecidos.

Nesta atividade, os alunos captam vários rostos utilizando a câmara HuskyLens, incluindo aqueles para os quais o modelo de IA foi treinado (rostos familiares) e outros que não foram encontrados durante as fases de treino e aprendizagem (rostos desconhecidos). O micro:bit apresentará então o resultado do processo de reconhecimento de cada rosto capturado.

Utilizando a câmara HuskyLens e a plataforma micro:bit, os alunos descobrirão como a tecnologia pode ser utilizada para testar o modelo de IA com rostos humanos familiares e desconhecidos.

3.7.2 Hardware

- Micro:bit
- Extensor IO para micro:bit
- HuskyLens
- Imagens a cores



3.7.3 Configuração

- Ligar o micro:bit com o extensor IO para micro:bit.
- Em seguida, utilize a tabela seguinte para ligar o extensor IO para micro:bit à câmara HuskyLens.

Extensor IO para micro:bit PORT	HuskyLens PORT
SDA	T
SCL	R
GND	-
3V3	+

3.7.3.1 Código

Depois de estabelecerem a ligação entre o micro:bit e o HuskyLens, os alunos devem escrever código que forneça instruções à câmara HuskyLens, permitindo a criação de um sistema de reconhecimento facial. Este código guiará a câmara na captura e processamento de rostos para os reconhecer eficazmente.



O aluno deve utilizar o software do seguinte sítio Web para escrever o código:
<https://makecode.microbit.org/>

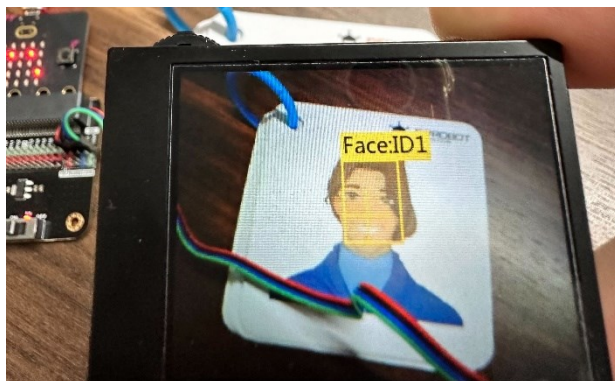
Explicação do código:

Para utilizar a funcionalidade HuskyLens, comece por adicionar a extensão HuskyLens à sua paleta de códigos. Com esta extensão, pode criar código que muda facilmente o algoritmo para o modo de reconhecimento facial.

Este código solicitará dados de entrada, exigindo que o aluno utilize a câmara HuskyLens para capturar o rosto de uma pessoa. O utilizador treinou o modelo de IA com um rosto específico na atividade anterior.

Quando a câmara detecta um rosto reconhecido, o micro:bit responde com um sorriso alegre e um som de felicidade, enquanto um rosto aleatório ou não reconhecido resulta num rosto triste e num som de tristeza. Este processo envolvente faz a ponte entre a tecnologia e a interação humana, tornando-o uma experiência de aprendizagem fascinante.

- A câmara HuskyLens utiliza o protocolo I2C. Prepara o micro:bit para falar com a câmara.
- Estamos a dizer ao HuskyLens para trabalhar no modo de reconhecimento de cores. Isto significa que tentará reconhecer e diferenciar objectos com base nas suas cores.
- A parte seguinte do código configura um ciclo contínuo que funcionará enquanto o micro:bit estiver ligado.
- Depois, o micro:bit pede ao HuskyLens para fornecer informações sobre o que está a ver no momento.
- O programa verifica se o HuskyLens aprendeu algum objeto. O número 1 refere-se à etiqueta do objeto e, se tiver aprendido alguma coisa, o código dentro das chavetas será executado.
- O HuskyLens está a detetar um objeto na sua vista.
- Em seguida, verifica se foi detectado um objeto específico (neste caso, o objeto 1).
- Se o HuskyLens reconhecer o objeto e este aparecer como um bloco, o micro:bit apresenta um ícone de cara feliz.
- Se o objeto não for reconhecido como um bloco, é apresentado um ícone de cara triste.



3.7.4 Exercício: Sistema de alarme

Utilizar o sistema HuskyLens e o micro:bit para treinar o modelo de IA com um novo rosto com o nome da pessoa e, em seguida, capturar o rosto da nova pessoa com a câmara HuskyLens para ver se consegue reconhecê-lo e se a câmara detecta uma pessoa que não está no sistema para emitir um som. O utilizador criará um sistema de alarme para este exercício.

3.8 Atividade 5: Introduzir a ideia de impacto social

3.8.1 Descrição

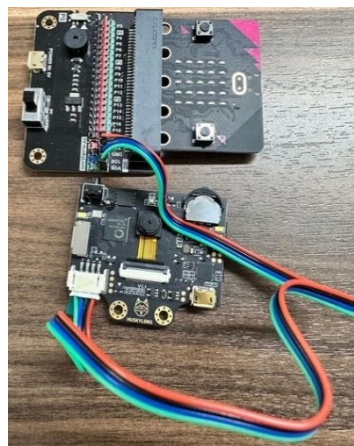
Esta aula introduz o conceito de impacto social através do treino de um modelo de reconhecimento de vários rostos. Os alunos irão explorar a forma de codificar e processar dados faciais para treinar um modelo capaz de reconhecer vários rostos em simultâneo.

Aprenderão a implementar esta tecnologia e discutirão as implicações sociais dos sistemas de IA capazes de detetar e identificar rostos em várias situações. A capacidade da IA para efetuar o reconhecimento de rostos tem impactos sociais significativos, incluindo maior segurança, preocupações com a privacidade e considerações éticas relativas à vigilância e à utilização de dados.

Esta lição tem como objetivo proporcionar aos alunos uma compreensão abrangente dos aspectos técnicos e sociais da tecnologia de reconhecimento facial.

3.8.2 Hardware

- Micro:bit
- Extensor IO para micro:bit
- HuskyLens
- Imagens com rostos



3.8.3 Configuração

- Ligar o micro:bit com o extensor IO para micro:bit.
- Em seguida, utilize a tabela seguinte para ligar o extensor IO para micro:bit à câmara HuskyLens.

Extensor IO para micro:bit PORT	HuskyLens PORT
SDA	T
SCL	R
GND	-
3V3	+

3.8.3.1 Código

Depois de estabelecerem a ligação entre o micro:bit e o HuskyLens, os alunos devem escrever código que forneça instruções à câmara HuskyLens, permitindo a criação de um sistema de reconhecimento facial. Este código guiará a câmara na captura e processamento de rostos para os reconhecer eficazmente. É importante para esta atividade utilizar a configuração para capturar várias imagens, de modo a criar a atividade seguinte.

O aluno deve utilizar o software do seguinte sítio Web para escrever o código:
<https://makecode.microbit.org/>

```

on start
  HuskyLens initialize I2C until success
  HuskyLens switch algorithm to Face Recognition

forever
  HuskyLens request data once and save into the result
  if HuskyLens check if ID 1 is learned from the result and HuskyLens check if ID 2 is learned from the result and HuskyLens check if ID 3 is learned from the result then
    if HuskyLens check if frame is on screen from the result then
      if HuskyLens check if ID 1 frame is on screen from the result then
        show string '1'
        pause (ms) 1000
        clear screen
      else if HuskyLens check if ID 2 frame is on screen from the result then
        show string '2'
        pause (ms) 1000
        clear screen
      else if HuskyLens check if ID 3 frame is on screen from the result then
        show string '3'
        pause (ms) 1000
        clear screen
      else
        show string '0'
        pause (ms) 1000
        clear screen
  
```

Explicação do código:

Para utilizar a funcionalidade HuskyLens, comece por adicionar a extensão HuskyLens à sua paleta de códigos. Com esta extensão, pode criar código que muda facilmente o algoritmo para o modo de reconhecimento facial.

Inicializa a comunicação I2C entre o microcontrolador e o HuskyLens. Continuará a tentar até que a inicialização seja bem sucedida. Define o HuskyLens para usar o algoritmo de Reconhecimento de Rosto. Solicita dados do HuskyLens uma vez e armazena-os no resultado. Assegura que as faces com IDs 1, 2 e 3 sejam aprendidas pelo HuskyLens.

O código é executado continuamente num ciclo, solicitando dados das HuskyLens e verificando se alguma das três faces é reconhecida. Dependendo da face detectada, apresenta o número correspondente (1, 2 ou 3) no ecrã do microcontrolador. Se não for reconhecida nenhuma face, é apresentado "0". Isto fornece feedback em tempo real com base no reconhecimento de rostos, tornando a interação interessante e informativa para o utilizador.

3.8.4 Introduzir a ideia de Interação Natural treinando um modelo de Reconhecimento de Rosto

Os alunos vão treinar o HuskyLens com um rosto e escrever código para o detetar e reproduzir um som.

Para esta atividade, é importante ter em conta os seguintes passos: Utilizar a opção Reconhecimento de Rosto, premir longamente o botão de função em Reconhecimento de Rosto, seleccionar Múltiplo e gravar 3 rostos.

3.9 Material e recursos adicionais

Tipo de recurso	Título	Tópico	Ligação
software	Código de fabrico	Codificação em bloco	https://makecode.microbit.org/
Artigo	Utilização de Microbit e Huskylens	Como utilizar o Huskylens	https://www.instructables.com/Microbit-Visual-Object-Tracking-With-Huskylens/