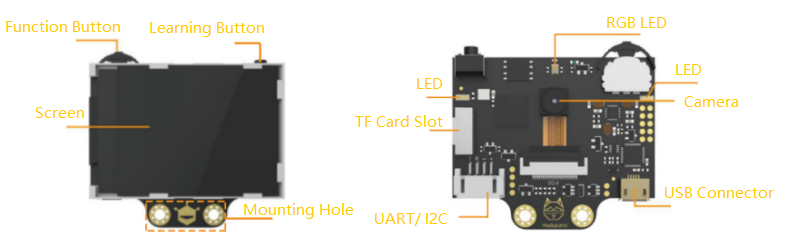
**📋 Modos de Funcionamento da HuskyLens (Menu de Modos)**

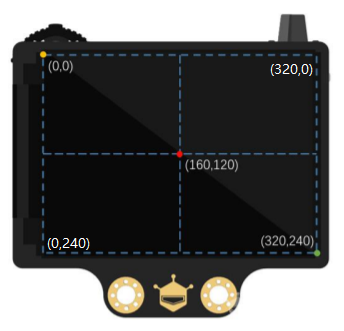
A HuskyLens possui **7 modos principais**, acessíveis diretamente pelo botão **“function button” (o pequeno botão lateral)**.

Cada modo ativa um algoritmo de visão computacional específico.



Uma imagem com circuito, Componente eletrónico, Engenharia eletrónica, Componente passivo de circuito

Os conteúdos gerados por IA podem estar incorretos.



**1. 🟥 Face Recognition (Reconhecimento Facial)**

* **Descrição**: Deteta e reconhece rostos humanos únicos.
* **Função**: Aprende rostos diferentes e atribui-lhes IDs.
* **Saída**: Coordenadas do rosto + ID + tamanho.
* **Uso**:
  + Acesso biométrico.
  + Contagem de pessoas.
  + Robôs que interagem com pessoas específicas.
* **Nota**: Requer boa iluminação e posicionamento frontal.

**2. 🟧 Object Tracking (Rastreamento de Objetos)**

* **Descrição**: Aprende um objeto visual (imagem/padrão) e rastreia-o em movimento.
* **Função**: Usa machine learning embutido para aprender e seguir.
* **Saída**: x, y, width, height, ID.
* **Uso**:
  + Seguidor de objeto com servo ou robô.
  + Interação com objetos físicos.
* **Nota**: É um dos modos mais usados para robótica educativa.

**3. 🟨 Object Recognition (Reconhecimento de Objetos)**

* **Descrição**: Aprende **vários objetos diferentes** e reconhece-os posteriormente.
* **Diferença face ao modo 2**: Pode distinguir entre **vários objetos aprendidos** com **diferentes IDs**.
* **Função**: Classificação de múltiplos objetos com base na imagem.
* **Uso**:
  + Jogos educativos.
  + Automação com identificação de peças.
* **Nota**: Ideal para aplicações tipo "classificador inteligente".

**4. 🟩 Line Tracking (Seguidor de Linha)**

* **Descrição**: Deteta linhas (pretas ou brancas) no solo e traça um **mapa do percurso**.
* **Função**: Usa IA para seguir trajetórias complexas (curvas, cruzamentos).
* **Saída**:
  + Coordenadas da linha.
  + Ângulo de direção.
* **Uso**:
  + Robôs seguidores de linha com aprendizagem automática do caminho.
* **Nota**: Pode gravar o percurso aprendido, útil para competições.

**5. 🟦 Color Recognition (Reconhecimento de Cor)**

* **Descrição**: Aprende uma ou mais cores específicas e reconhece-as no campo de visão.
* **Função**: Reconhece zonas de cor com coordenadas e ID.
* **Uso**:
  + Robôs que reagem a comandos por cor (ex: vermelho = parar).
  + Jogos com cartões coloridos.
* **Nota**: Sensível a iluminação. Cores no ecrã funcionam melhor que impressas.

**6. 🟪 Tag Recognition (Reconhecimento de Etiquetas / Tags)**

* **Descrição**: Deteta etiquetas especiais (tags) baseadas em padrões QR-like, próprios da HuskyLens.
* **Função**: Cada tag tem um ID único (1 a 8 por defeito).
* **Uso**:
  + Comandos físicos (ex: Tag 1 = avançar, Tag 2 = parar).
  + Navegação por checkpoints.
* **Nota**: Requer tags impressas fornecidas pela DFRobot ou geradas manualmente.

**7. ⬛ Object Classification (Classificação de Objetos com IA)**

* **Descrição**: Usa um modelo de **aprendizagem profunda (deep learning)** embarcado para **classificar objetos complexos**.
* **Função**: Aprende diferentes tipos de objetos com base em muitos exemplos.
* **Uso**:
  + Reconhecimento mais robusto com múltiplas imagens por classe.
  + Classificação por aparência geral (ex: maçã vs banana).
* **Nota**: Requer treino mais intensivo, mas é o modo com maior capacidade de generalização.

**⚙️ Outras Opções de Menu (Acessíveis via interface na câmara)**

Além dos modos de operação, a HuskyLens inclui opções de configuração acessíveis pressionando **longamente o botão "function"**:

* **Save/Load Data** – Guarda ou carrega dados aprendidos.
* **Adjust Screen Brightness** – Ajuste de brilho do visor.
* **Firmware Update** – Atualização via USB.
* **Change Baudrate** – Ajusta velocidade de comunicação UART.
* **Enable/Disable Learning Mode** – Bloqueia novos aprendizados acidentais.

**📡 Comunicação**

Todos os modos permitem comunicação com o microcontrolador via:

* **UART** (Serial – ex: SoftwareSerial)
* **I2C** (endereçável e integrável em redes de sensores)

| **Opção** | **Descrição Técnica** | **Observações e Sugestões Práticas** |
| --- | --- | --- |
| 🔍 **Learn Multiple** | Permite aprender **vários rostos**, cada um com um ID diferente. | Ideal para sistemas multiutilizador. Cada rosto recebe um ID único. |
| 🎚️ **Detect Threshold** | Define a **sensibilidade da deteção inicial** de rostos. | Valor mais alto = deteção mais exigente. Afeta se um rosto é detetado ou não. Valores típicos: 30–100. |
| 📐 **NMS Threshold** | Define a **threshold do Non-Maximum Suppression**. | Controla a sobreposição entre detecções. Mais baixo = menos duplicações. Valor típico: 0.3–0.6. |
| 🧠 **Recognize Threshold** | Define a **exigência para considerar um rosto como conhecido (match)**. | Valor alto = mais rigoroso (menos falsos positivos). Valor típico: 0.6–0.9. |

**🟥 Face Recognition – Opções Avançadas (via botão esquerdo)**

Ao pressionar o **botão esquerdo (grande)** no modo *Face Recognition*, surge o **menu contextual do modo**, com as seguintes opções:

**📘 Explicação técnica resumida:**

**🔸 1. Learn Multiple**

* Liga o modo de aprendizagem contínua de vários rostos.
* Cada novo rosto que mostres será armazenado com um novo ID (ID: 1, 2, 3, ...).
* A aprendizagem é confirmada quando aparece a box com o ID no visor.
* Para treinar um novo rosto:
  1. Ativar esta opção.
  2. Apontar para novo rosto.
  3. Pressionar botão direito (pequeno) para aprender.

**🔸 2. Detect Threshold**

* Threshold para **detetar se há um rosto na imagem**, mesmo antes de reconhecer.
* Valor baixo: mais permissivo (mas pode gerar ruído).
* Valor alto: mais seletivo, deteta apenas rostos claros e bem centrados.

**🔸 3. NMS Threshold**

(*Non-Maximum Suppression*)

* Usado para remover **detecções redundantes (sobrepostas)**.
* Se estiver muito alto (>0.7), podem aparecer várias caixas sobre o mesmo rosto.
* Se muito baixo (<0.3), pode ignorar rostos que deviam ser reconhecidos.
* Sugestão: manter entre **0.4 e 0.6**.

**🔸 4. Recognize Threshold**

* Define a “confiança” mínima para que o sistema diga: "é o João!".
* Se for baixa (<0.5), o sistema aceita rostos parecidos.
* Se for alta (>0.8), só reconhece com muita certeza (pode falhar em alguns ângulos).
* Ideal: **0.6 a 0.75** para uso realista.

**🟧 Opções avançadas do modo Object Tracking**

*(Acessíveis ao manter pressionado o botão esquerdo — Function Button)*

| **Opção** | **Descrição técnica** | **Aplicação e sugestão prática** |
| --- | --- | --- |
| ✅ **Save & Return** | Guarda as configurações feitas e volta ao modo de operação normal. | Confirma qualquer alteração feita no menu. |
| 💾 **Auto Save** | Se ativo, os objetos aprendidos e configurações são guardados automaticamente. | Evita perder aprendizagem ao desligar. Recomenda-se manter **ON**. |
| 📚 **Learn Enable** | Ativa/desativa a capacidade de aprender o objeto com o botão direito. | Útil para evitar aprendizagens acidentais depois do treino inicial. |
| 🧠 **Multi Core** | Usa os dois núcleos do processador da HuskyLens para desempenho melhorado. | Pode melhorar a resposta em seguimento rápido. Recomenda-se manter **ON**. |
| 🟥 **Frame Ratio** | Define a proporção da caixa de deteção: largura/altura. | Ex: 1.0 = quadrado; 2.0 = retângulo largo. Ajusta-se ao formato do objeto a seguir. |
| 📐 **Frame Size** | Define o **tamanho mínimo** da bounding box detetável. | Evita deteção de objetos muito pequenos (ruído). |
| 🎚 **Threshold** | Sensibilidade da deteção. Mais alto = mais rigoroso, menos falsos positivos. | Começa com valor médio (ex: 60–100). |
| 🚫 **False Positive** | Ajusta o filtro para evitar deteções erradas. | Alta = mais rigoroso, mas pode perder deteções válidas. |
| ✅ **True Positive** | Ajusta a confiança mínima para aceitar uma deteção como válida. | Alta = mais seletivo. Valor típico: 0.6–0.8. |

**🧪 Sugestão de calibração prática**

| **Parâmetro** | **Valor sugerido** | **Observação** |
| --- | --- | --- |
| Auto Save | ON | Guarda tudo sem intervenção |
| Learn Enable | ON (para treino) → OFF (depois) | Para evitar sobrescrita |
| Multi Core | ON | Melhor desempenho |
| Frame Ratio | 1.0 | Objeto quadrado; ajustar se necessário |
| Frame Size | 80–120 | Evita ruído de pequenos objetos |
| Threshold | 80 | Boa relação sensibilidade/precisão |
| False Positive | 3 | Filtro moderado |
| True Positive | 0.7 | Confiança média-alta |

**⚙️ Navegação rápida**

* **Botão esquerdo (grande)** → abrir menu.
* **Rodar botão ou tocar** → navegar pelas opções.
* **Botão direito (pequeno)** → confirmar/alternar valores.
* **Guardar com "Save & Return"** no fim.

**🟨 Modo: Object Recognition – Opções Reais do Menu Interno**

\*(acessíveis ao manter pressionado o botão esquerdo — **Function Button**)

| **🛠️ Opção** | **📋 Descrição técnica** | **💡 Aplicação prática** |
| --- | --- | --- |
| ✅ **Save & Return** | Guarda as configurações e regressa ao modo de operação normal. | Usa sempre após ajustes. |
| 📚 **Learn Multiple** | Permite marcar **vários objetos diferentes**, cada um com um **ID único**. | Ativa reconhecimento multiobjeto. |
| 🧠 **Recognize Threshold** | Define o nível mínimo de **confiança** para aceitar um objeto como reconhecido (0.0 a 1.0). | 0.6–0.8 é equilibrado; mais alto = mais rigoroso. |
| 📐 **NMS Threshold** | Ajusta o valor de **Non-Maximum Suppression** – elimina deteções sobrepostas com o mesmo ID. | 0.3–0.6 funciona bem na maioria dos casos. |

**🧪 Sugestões práticas de calibração**

| **Parâmetro** | **Valor sugerido** | **Justificação** |
| --- | --- | --- |
| Learn Multiple | ON | Para reconhecer mais de 1 objeto |
| Recognize Threshold | 0.7 | Boa relação entre precisão e tolerância |
| NMS Threshold | 0.5 | Elimina sobreposições duplicadas |
| Save & Return | — | Usar sempre após ajustes |

**🟩 Modo: Line Tracking – Opções de Menu Interno**

*(acessíveis ao manter pressionado o botão esquerdo — Function Button)*

| **🛠️ Opção** | **📋 Descrição técnica** | **💡 Aplicação prática** |
| --- | --- | --- |
| ✅ **Save & Return** | Guarda as configurações e regressa ao modo de deteção. | Usa sempre após ajustes. |
| 📚 **Learn Multiple** | Permite aprender **vários segmentos de linha** como parte do mesmo trajeto. | Essencial para linhas com curvas/interseções. |
| 💡 **LED Switch** | Liga/desliga os LEDs frontais da HuskyLens. | Útil para melhorar contraste com iluminação fraca. |

**🧪 Dicas práticas de utilização**

* **Learn Multiple** deve estar **ativo** quando quiseres ensinar linhas com várias curvas ou segmentos.
* O **LED Switch** deve ser **ligado** em ambientes com sombras ou pouca luz — melhora muito a deteção da linha.
* Após configurar, termina sempre com **Save & Return** para guardar tudo.

**Exemplo de processo:**

1. Entra em *Line Tracking*.
2. Mantém pressionado o botão esquerdo → entra no menu.
3. Liga Learn Multiple e os LEDs, se necessário.
4. Sai com Save & Return.
5. Aponta a câmara para a linha e pressiona o botão direito para aprender o trajeto.

**🟦 Modo: Color Recognition – Opções de Menu Interno (reais)**

*(acessíveis ao manter pressionado o botão esquerdo — Function Button)*

| **🛠️ Opção** | **📋 Descrição técnica** | **💡 Aplicação prática** |
| --- | --- | --- |
| ✅ **Save & Return** | Guarda os parâmetros atuais e regressa ao modo de operação. | Usa sempre após configurar. |
| 📚 **Learn Multiple** | Permite aprender e identificar **várias cores diferentes**, cada uma com um **ID próprio**. | Ex: vermelho = ID1, azul = ID2, verde = ID3. |
| 🎚 **Start Range** | Define a **tolerância de início** para deteção da cor durante a aprendizagem. | Afeta quão cedo a HuskyLens começa a identificar a cor. |
| 🎚 **Accept Range** | Define a **tolerância de reconhecimento** após a aprendizagem — maior = aceita tons mais semelhantes. | Ajusta a sensibilidade para reconhecer tons próximos. |

**🧪 Sugestões práticas de calibração**

| **Parâmetro** | **Valor sugerido** | **Justificação** |
| --- | --- | --- |
| Learn Multiple | ON | Para usar várias cores com IDs únicos |
| Start Range | 10–30 | Evita deteções acidentais muito cedo |
| Accept Range | 20–50 | Mais alto = mais tolerante a variações |
| Save & Return | — | Confirmar sempre no fim |

**🎯 Dica:**

O **Accept Range** funciona como um "filtro de semelhança". Se estiver baixo, só a cor exata é reconhecida. Se estiver alto, tons parecidos (ex: vermelho alaranjado) também serão aceites.

**🟪 Modo: Tag Recognition – Opções Reais de Menu**

*(acessíveis ao manter pressionado o botão esquerdo — Function Button)*

| **🛠️ Opção** | **📋 Descrição técnica** | **💡 Aplicação prática** |
| --- | --- | --- |
| ✅ **Save & Return** | Guarda os parâmetros ajustados e regressa ao modo de deteção. | Confirmar após qualquer alteração. |
| 📚 **Learn Multiple** | Permite reconhecer **várias tags diferentes** em simultâneo, cada uma com o seu **ID**. | Ex: Tag 1 = avançar, Tag 2 = parar, etc. |

**🧠 Notas importantes:**

* **Não é necessário aprender manualmente as tags.**
  + A HuskyLens já reconhece as **tags padrão da DFRobot**, que têm IDs entre 1 e 8.
  + O modo Tag Recognition identifica-as diretamente e mostra:
    - **Moldura colorida**
    - **ID** da tag no ecrã
    - **Posição (X/Y)** e tamanho

**⚙️ *General Settings*** da HuskyLens (firmware **v0.4.6 Stable**):

**⚙️ General Settings – Opções disponíveis**

| **Opção** | **Descrição** |
| --- | --- |
| ✅ **Save & Return** | Guarda alterações feitas no menu e regressa ao modo normal |
| 🔄 **Protocol Type** | Seleciona entre **UART** ou **I2C** para comunicação serial |
| 💡 **Screen Brightness** | Ajusta o brilho do visor LCD |
| 🕒 **Menu Auto Hide** | Ativa/desativa o fecho automático do menu após inatividade |
| 🔦 **LED Switch** | Liga/desliga os LEDs brancos frontais |
| 🔆 **LED Brightness** | Ajusta o brilho dos LEDs frontais |
| 🌈 **RGB Switch** | Liga/desliga os LEDs RGB embutidos (visualização de estado) |
| 🌈 **RGB Brightness** | Ajusta o brilho dos LEDs RGB |
| 🏭 **Factory Reset** | Restaura todos os parâmetros aos valores de fábrica |
| 🔢 **Version** | Mostra a versão do firmware atual (**v0.4.6 Stable**) |
| 🌍 **Language** | Define o idioma da interface (ex: English, 中文) |