

// Histórico de alterações:

// 2021-08-01: Criado por Mauro Braga.

// 2021-08-02: Adicionado comentários explicativos e seção de histórico de alterações.

import org.opencv.core.Core;

import org.opencv.core.Mat;

import org.opencv.core.MatOfRect;

import org.opencv.core.Rect;

import org.opencv.core.Scalar;

import org.opencv.core.Size;

import org.opencv.imgcodecs.Imgcodecs;

import org.opencv.imgproc.Imgproc;

import org.opencv.objdetect.CascadeClassifier;

public class Condominio {

// Variáveis para controle do visitante

private CascadeClassifier faceCascade; // Classificador facial

private boolean visitanteDesordeiro; // Indica se o visitante é desordeiro

private boolean visitanteSupervisionado; // Indica se o visitante está sob supervisão

private boolean multaEnviada; // Indica se a multa foi enviada ao morador

// Construtor da classe

public Condominio() {

 faceCascade = new CascadeClassifier();

 faceCascade.load("haarcascade_frontalface_default.xml"); // Caminho para o arquivo XML
do classificador facial

 visitanteDesordeiro = false;

 visitanteSupervisionado = false;

 multaEnviada = false;

```
}
```

```
// Método para analisar a imagem e detectar faces
```

```
public void analisarImagem(String morador, String visitante, String imagePath) {
```

```
    Mat imagem = Imgcodecs.imread(imagePath); // Carrega a imagem
```

```
    // Converte a imagem para escala de cinza para melhorar o desempenho do  
    reconhecimento facial
```

```
    Mat imagemCinza = new Mat();
```

```
    Imgproc.cvtColor(imagem, imagemCinza, Imgproc.COLOR_BGR2GRAY);
```

```
    // Redimensiona a imagem para acelerar o processo de detecção facial
```

```
    Imgproc.resize(imagemCinza, imagemCinza, new Size(640, 480));
```

```
    // Detecta faces na imagem
```

```
    MatOfRect faces = new MatOfRect();
```

```
    faceCascade.detectMultiScale(imagemCinza, faces);
```

```
    if (faces.toArray().length > 0) {
```

```
        // Face detectada
```

```
        if (visitanteDesordeiro && !visitanteSupervisionado) {
```

```
            System.out.println("ALERTA: O visitante " + visitante + " é desordeiro! Necessita  
assinatura do morador para liberar.");
```

```
            visitanteSupervisionado = true;
```

```
        } else if (visitanteDesordeiro && visitanteSupervisionado) {
```

```
            System.out.println("O visitante " + visitante + " está sob supervisão do sistema de  
segurança.");
```

```
        }
```

```
    // Processa ações do visitante
```

```
    // ...
```

```
// Atualiza os status de desordeiro, supervisionado e multa enviada
visitanteDesordeiro = true;
visitanteSupervisionado = true;
multaEnviada = false;
} else {
    // Nenhuma face detectada
    visitanteDesordeiro = false;
    visitanteSupervisionado = false;
}
}
}
```