

OpenCV+OpenCvSharp打造智能考勤系统，实现高效人脸录入和精准考勤识别

作者：微信公众号：【架构师老卢】

11-28 9:54

1334



概述：该考勤系统基于OpenCV和OpenCvSharp实现，包含员工人脸录入和上下班考勤人脸识别。员工人脸特征通过ORB方法提取并存储，考勤时通过相似度计算识别员工。系统灵活、可扩展，提高考勤效率，确保准确性。

实现基于OpenCV和OpenCvSharp的考勤系统，包括员工人脸录入和上下班考勤人脸识别。以下是详细步骤和示例代码：

步骤1：安装OpenCvSharp

确保在项目中已安装OpenCvSharp库。通过NuGet包管理器或包管理控制台执行以下命令：

```
1 Install-Package OpenCvSharp4
```

步骤2：编写代码

```
1 using System;
2 using System.Collections.Generic;
3 using OpenCvSharp;
4 using OpenCvSharp.CPlusPlus; // 或者使用OpenCvSharp4
5
6 class Program
7 {
8     // 全局变量用于存储员工的人脸特征
9     static Dictionary<string, List<float>> employeeFaceFeatures = new Dictionary<string, List<float>>();
10
11     static void Main()
12     {
13         // 步骤3: 员工人脸录入
14         EmployeeFaceEnrollment("Employee1", "path/to/employee1.jpg");
15         EmployeeFaceEnrollment("Employee2", "path/to/employee2.jpg");
16
17         // 步骤4: 上下班考勤人脸识别
18         FaceRecognition("path/to/attendance_face.jpg");
19     }
20
21     // 步骤3: 员工人脸录入的方法
22     static void EmployeeFaceEnrollment(string employeeName, string imagePath)
23     {
24         Mat faceImage = Cv2.ImRead(imagePath, ImreadModes.Color);
25
26         // 提取人脸特征
27         List<float> faceFeature = ExtractFaceFeature(faceImage);
28
29         // 存储人脸特征到全局变量中
30         employeeFaceFeatures[employeeName] = faceFeature;
31
32         Console.WriteLine($"{employeeName}的人脸特征已录入。");
33     }
34
35     // 步骤4: 上下班考勤人脸识别的方法
36     static void FaceRecognition(string attendanceImagePath)
37     {
38         Mat attendanceFaceImage = Cv2.ImRead(attendanceImagePath, ImreadModes.Color);
39
40         // 提取考勤人脸的特征
41         List<float> attendanceFaceFeature = ExtractFaceFeature(attendanceFaceImage);
42
43         // 与员工人脸特征进行比对
44         string recognizedEmployee = RecognizeEmployee(attendanceFaceFeature);
45
46         // 输出考勤结果
47         if (!string.IsNullOrEmpty(recognizedEmployee))
48         {
49             Console.WriteLine($"识别到员工: {recognizedEmployee}, 考勤成功。");
50         }
51         else
52         {
53             Console.WriteLine("未识别到员工, 考勤失败。");
54         }
55     }
56
57     // 提取人脸特征的方法
58     static List<float> ExtractFaceFeature(Mat faceImage)
59     {
60         // 使用OpenCV的方法提取人脸特征，例如人脸识别模型
61         // 这里简单地使用ORB方法提取特征向量
62         using (var orb = new ORB())
63         {
64             KeyPoint[] keyPoints;
65             Mat descriptors = new Mat();
66             orb.DetectAndCompute(faceImage, null, out keyPoints, descriptors);
67
68             // 返回特征向量
69             return descriptors.ToFloatArray();
70         }
71     }
72
73     // 识别员工的方法
74     static string RecognizeEmployee(List<float> attendanceFaceFeature)
75     {
76         foreach (var employee in employeeFaceFeatures)
77         {
78             double similarity = CalculateSimilarity(employee.Value, attendanceFaceFeature);
79
80             // 设置相似度阈值，可以根据实际情况调整
81             double similarityThreshold = 0.7;
82
83             if (similarity > similarityThreshold)
84             {
85                 return employee.Key;
86             }
87         }
88
89         return null;
90     }
91
92     // 计算相似度的方法
93     static double CalculateSimilarity(List<float> featureVector1, List<float> featureVector2)
94     {
95         // 使用OpenCV的方法计算相似度，例如欧氏距离、余弦相似度等
96         // 这里简单地使用余弦相似度计算
97         double dotProduct = 0;
98         double magnitude1 = 0;
99         double magnitude2 = 0;
100
101         for (int i = 0; i < featureVector1.Count; i++)
102         {
103             dotProduct += featureVector1[i] * featureVector2[i];
104             magnitude1 += Math.Pow(featureVector1[i], 2);
105             magnitude2 += Math.Pow(featureVector2[i], 2);
106         }
107
108         if (magnitude1 == 0 || magnitude2 == 0)
109             return 0;
110
111         return dotProduct / (Math.Sqrt(magnitude1) * Math.Sqrt(magnitude2));
112     }
113 }
```

请注意：

- 步骤3中的图片路径需要替换为实际的员工人脸图像路径。

- 步骤4中的图片路径需要替换为实际的考勤人脸图像路径。
- 步骤3中的`ExtractFaceFeature`方法需要根据实际需求选择合适的人脸特征提取方法。
- 步骤4中的`RecognizeEmployee`方法根据实际情况调整相似度阈值。
- 这只是一个简单的示例，实际中可能需要更复杂的人脸识别模型和数据库存储方式。确保你的项目引用了正确版本的OpenCvSharp库。