

**ĐẠI HỌC QUỐC GIA TP.HỒ CHÍ MINH**  
**TRƯỜNG ĐẠI HỌC BÁCH KHOA**  
**KHOA ĐIỆN – ĐIỆN TỬ**  
**BỘ MÔN ĐIỆN TỬ**

-----o0o-----



**ĐỒ ÁN MÔN HỌC 2**

**THIẾT KẾ PHẦN MỀM**  
**ĐẾM SỐ NGƯỜI RA VÀO**  
**BẰNG CAMERA**

**GVHD: ThS Bùi Quốc Bảo**

**SVTH: Tạ Lục Gia Hoàng**

**MSSV: 41201241**

**TP. HỒ CHÍ MINH, THÁNG 6 NĂM 2016**

----- ☆ -----

----- ☆ -----

Số: \_\_\_\_\_/BKĐT  
Khoa: **Điện – Điện tử**  
Bộ Môn: **Điện Tử**

## NHIỆM VỤ ĐỒ ÁN MÔN HỌC 2

1. HỌ VÀ TÊN: TẠ LỤC GIA HOÀNG MSSV: 41201241
2. NGÀNH: **ĐIỆN TỬ - VIỄN THÔNG** LỚP : DD12DV01
3. Đề tài: THIẾT KẾ PHẦN MỀM ĐẾM SỐ NGƯỜI RA VÀO BẰNG CAMERA
4. Nhiệm vụ:  
*Sử dụng 1 camera thu hình ảnh và xử lý bằng một phần mềm.*  
*Lập trình bằng ngôn ngữ C*  
*Bước đầu làm quen với thư viện mã nguồn mở OpenCV*  
.....  
.....  
.....
5. Ngày giao nhiệm vụ đồ án môn học 2: 18/01/2016
6. Ngày hoàn thành nhiệm vụ: 09/06/2016
7. Họ và tên người hướng dẫn: ..... Phần hướng dẫn  
.....  
.....

Nội dung và yêu cầu ĐAMH2 đã được thông qua Bộ Môn.

*Tp.HCM, ngày..... tháng 06 năm 2016*  
**CHỦ NHIỆM BỘ MÔN**

**NGƯỜI HƯỚNG DẪN CHÍNH**

### PHẦN DÀNH CHO KHOA, BỘ MÔN:

Người duyệt (chấm sơ bộ):.....  
Đơn vị:.....  
Ngày bảo vệ : .....  
Điểm tổng kết: .....

## ***LỜI CẢM ƠN***

*Xin chân thành cảm ơn ThS. Bùi Quốc Bảo (Bộ môn Điện tử, Khoa Điện-Điện tử, Trường Đại học Bách Khoa – Đại học Quốc gia TP Hồ Chí Minh), các anh ở Vườn Ươm Doanh Nghiệp – Trường Đại học Bách Khoa – Đại học Quốc gia TP Hồ Chí Minh) đã tận tình giúp đỡ, hướng dẫn em hoàn thành tốt đề tài.*

*Tp. Hồ Chí Minh, ngày 03 tháng 06 năm 2016 .*

**Sinh viên**

**TẠ LỤC GIA HOÀNG**

## MỤC LỤC

1. GIỚI THIỆU .....	3
1.1 Tổng quan .....	3
1.3 Phương pháp thực hiện .....	4
2. LÝ THUYẾT .....	5
2.1 Một số Camera an ninh.....	5
2.2 OpenCV .....	5
2.3 Thuật toán trừ nền (Background Subtraction) .....	6
2.4 Theo vết chuyển động (Tracking and Motion) .....	9
3. CHỌN THIẾT BỊ PHẦN CỨNG .....	13
4. THIẾT KẾ VÀ THỰC HIỆN PHẦN MỀM.....	13
5. KẾT QUẢ THỰC HIỆN.....	14
6. KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN .....	15
6.1 Kết luận.....	15
6.2 Hướng phát triển .....	15
7. TÀI LIỆU THAM KHẢO.....	16

## DANH SÁCH HÌNH MINH HỌA

Hình 2.5.1: Average Background Method .....	7
Hình 2.5.2: Giá trị của 1 pixel theo thời gian.....	8
Hình 2.5.3: Codebook Background Method .....	9
Hình 2.6.1: Các bóng của 2 vật chuyển động .....	10
Hình 2.6.2: (A) Độ lớn các gradient và hướng. ....	11
(B) lược bỏ các gradient quá lớn.....	11
(C) gradient tổng .....	11
Hình 2.6.3: Tách các vật chuyển động để tính gradient tổng cho từng vật.....	12
Hình 3: Camera Logitech C920 .....	13
Hình 5: Kết quả chạy chương trình.....	14

## 1. GIỚI THIỆU

### 1.1 Tổng quan

Nghiên cứu thị trường là lĩnh vực yêu cầu phân tích dữ liệu. Ví dụ, muốn biết có bao nhiêu người vào một ngân hàng, ta phải đếm từng người một ra vào ngân hàng. Có rất nhiều phương pháp giải bài toán này. Đơn giản nhất có thể sử dụng công quét bằng tia hồng ngoại hoặc máy đếm cơ học, nhưng chúng không thể hoạt động ở nơi mật độ người quá đông. Trước khi có sự trợ giúp của các thuật toán xử lý ảnh, lựa chọn hàng đầu cho doanh nghiệp là các bộ đếm sử dụng cảm biến nhiệt để dò thân nhiệt con người. Các bộ đếm này khá chính xác, tuy nhiên có một số hạn chế như không dùng được với trần nhà cao, chỉ có thể dùng với cửa hẹp, khó kiểm tra độ chính xác của bộ đếm, độ chính xác giảm ở khu vực có nhiệt độ môi trường thay đổi.

Với sự phát triển của kỹ thuật máy tính, các thuật toán xử lý ảnh phức tạp bây giờ có thể giúp thực hiện việc đếm người bằng camera thu hình. Thị giác máy tính làm việc thông qua một thiết bị nhúng. Việc này làm giảm băng thông cần thiết vì chỉ dữ liệu đếm được gửi đi. Các thuật toán được phát triển để cung cấp cho việc đếm trở nên cực kỳ chính xác cả trong nhà và ngoài trời. Độ chính xác có thể đạt 98% trong các môi trường ánh sáng khác nhau. Áp dụng thêm trí tuệ nhân tạo và nhận dạng vật thể khiến cho độ chính xác ngày càng cao.

Camera đếm người có thể ứng dụng ở nhiều nơi như:

Đếm khách vào chơi trò chơi ở Công viên giải trí : Người quản lý có thể biết ngay tức thời hoặc báo cáo hằng ngày số lượng khách vào chơi trò chơi. So sánh số liệu này với lượng vé bán ra để giám sát nhân viên soát vé, tránh thất thoát. Phân tích báo cáo này biết được trò chơi nào nên tiếp tục đầu tư, trò nào ít thu hút để điều chỉnh hướng kinh doanh.

Đếm số người ra vào cho hệ thống bán lẻ di động : Báo cáo cho biết tình hình kinh doanh của từng Shop, từ số liệu này đưa ra hiệu quả thu hút khách hàng của các Shop bán lẻ trong ngày bình thường và trong thời gian có khuyến mãi.

Doanh trại quân đội kiểm soát lưu lượng ra vào : biết được số lượng người ra vào doanh trại là ổn định hay bất thường cho an ninh/an toàn. Số liệu được xem trực tiếp trên màn hình giúp Giám sát viên biết được tức thời để ứng phó.

Đếm số công nhân ra vào công trường : hiển thị số lượng Công nhân vào làm việc trong ngày ngay trên màn hình hiển thị ở cổng bảo vệ. Cho biết lượng công nhân vào công trường làm việc hằng ngày. Đặc biệt là có công nhân nào còn sót lại trong công trường sau giờ làm việc không, để ứng cứu nếu bị thương một mình mà không phát hiện hoặc có ý trộm cắp. Việc này dựa trên số liệu Ra thấp hơn Vào hằng ngày.

Phòng họp của thủy điện : Không cho 1 nhóm lớn hơn 5 người vào lối đi chung 1 lúc, nếu Camera đếm vượt quá 5 người sẽ báo động bằng đèn Alarm. Báo cáo số liệu ra vào hằng ngày cho quản lý.

Phòng tẩy rửa sát trùng của xưởng công nghệ cao : Mỗi phòng tẩy rửa sát trùng này chỉ cho 1 người vào 1 lúc, nếu vào 2 người trở lên sẽ ảnh hưởng chất lượng sát trùng. Camera quan sát đếm số người trong phòng này để cảnh báo Quản lý trực ca. Kết hợp đóng lối vào xưởng sản xuất.

### **1.2 Mục đích nghiên cứu**

Xuất phát từ nhu cầu thực tiễn trên cho thấy việc đếm người bằng camera và xử lý ảnh rất quan trọng trong cuộc sống ngày nay.

Đề tài này nghiên cứu một số thuật toán trong việc đếm người và làm quen với OpenCV, thư viện mã nguồn mở được sử dụng rộng rãi trên toàn thế giới.

### **1.3 Phương pháp thực hiện**

Sử dụng một vạch ảo trên khung ảnh để đếm mỗi khi có người đi qua vạch ảo.đó. Phương pháp này thường được gọi là line-crossing. Giả sử rằng vạch ảo nằm vuông góc hướng chuyển động của người.

Đề tài sử dụng:

- Một camera Logitech C920
- Công cụ lập trình: Visual Studio 2010
- Ngôn ngữ lập trình: C
- Thư viện mã nguồn mở OpenCV phiên bản 2.4.10

## 2. LÝ THUYẾT

### 2.1 Một số Camera an ninh

Mô tả tổng quan về lĩnh vực liên quan đến đề tài và những mục tiêu cần nghiên cứu. Từ đó giới thiệu nhiệm vụ cần đặt ra cho đề tài luận văn.

#### **Camera IP:**

Là loại camera quan sát có hình ảnh được số hóa, xử lý và mã hóa từ bên trong, sau đó truyền tải tín hiệu ảnh số qua một kết nối Ethernet về máy vi tính, cũng có thể là một thiết bị lưu trữ tín hiệu số như: hệ thống NAS, hệ thống server hoặc đầu ghi hình IP.

IP là viết tắt của Internet Protocol. Nghĩa là mỗi camera IP luôn có một địa chỉ IP riêng.

Camera IP có thể sử dụng 2 loại cảm biến CMOS và CCD, và có loại kết nối hồng ngoại hoặc wifi.

Camera IP: sử dụng cổng RJ45. (Để gắn vào hub mạng).

Truyền dẫn tín hiệu bằng cáp mạng.

#### **Camera Analog:**

Camera Analog là một camera sử dụng cảm biến CCD để thu nhận khung hình và sau đó hình ảnh được số hóa để xử lý.

Nhưng trước khi có thể truyền tải hình ảnh nó phải chuyển đổi thành tín hiệu Analog, sau đó truyền tải về thiết bị thu tín hiệu analog. Ví dụ như: Tivi, đầu ghi hình analog...

Không giống như camera IP, camera Analog không được tích hợp giao diện web, không thể truy cập kết nối trên internet một cách độc lập. Mà cần phải có thiết bị trung chuyển là đầu ghi hình Analog hoặc card ghi hình analog.

Sử dụng cổng BNC (để gắn vào đầu ghi hình analog) .

Truyền dẫn tín hiệu hình ảnh bằng cáp đồng trục

### 2.2 OpenCV

Open Source Computer Vision là một thư viện mã nguồn mở với các hàm xử lý ảnh dùng cho thị giác máy tính thời gian thực. Được dùng miễn phí cả trong học thuật và thương mại.

Giao diện C++, C, Python, Java và hỗ trợ cho Windows, Linux, Mac OS, iOS và Android.

OpenCV được viết tối ưu hóa bằng ngôn ngữ C chủ yếu dùng cho các ứng dụng thời gian thực.

Được ứng dụng rộng rãi trong nhiều lĩnh vực như khôi phục ảnh vệ tinh, bản đồ mạng, giảm nhiễu ảnh y tế, phân tích vật thể, hệ thống giám sát an ninh, hệ thống bảo hộ, ứng dụng quân đội...



### 2.3 Thuật toán trừ nền (Background Subtraction)

Làm thế nào để tách vật hoặc từng phần của vật ra khỏi các phần còn lại của ảnh. Trong video an ninh, phần lớn thời gian camera chỉ quan sát cùng một background (nền) giống với thời điểm trước đó, là những thông tin không đổi và không cần thiết. Những thông tin mà ta quan tâm chính là con người, xe cộ đi vào vùng theo dõi, hoặc khi có vật nào đó đi ra khỏi vùng theo dõi. Do đó cần phải cách ly các sự kiện đó ra khỏi background để dễ dàng xử lý.

#### Một số thuật toán:

**a) Background Subtraction:** là thuật toán đơn giản và cơ bản nhất trong việc xử lý ảnh áp dụng cho các ứng dụng video an ninh. Vì các camera thường đặt cố định, nên trước tiên ta cần “học” (lấy mẫu) một background, sau khi học, ảnh background này sẽ được lưu và so sánh với các ảnh hiện tại. Trừ ảnh hiện tại với background. Các phần khác nhau còn lại sau khi trừ là foreground (phần nổi).

Điểm yếu của Background Subtraction:

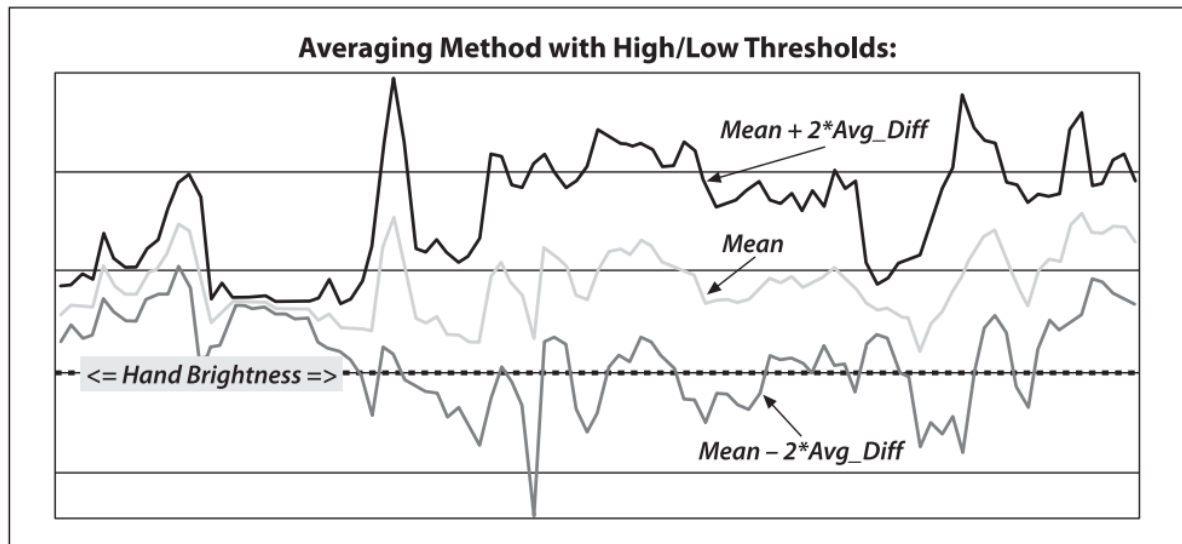
Chỉ hoạt động tốt cho các quang cảnh đơn giản, không đổi theo thời gian.

Không phân biệt được background với foreground trong các trường hợp thường gặp như

- ✓ khi ánh sáng thay đổi (vd: ánh nắng mặt trời thay đổi trong ngày, bật tắt đèn trong phòng...).
- ✓ Không cập nhật được thông tin mới khi background thay đổi (vd: một vật bị di chuyển sang vị trí khác và đặt cố định trong một thời gian dài, khi đó foreground sẽ bao gồm cả vị trí mới và cũ của vật đó).

**b) Averaging Method:**

Phương pháp này cho phép học giá trị trung bình và độ lệch chuẩn (hoặc độ lệch trung bình) của từng pixel khi lấy mẫu background. Mỗi pixel sau khi học sẽ có một giá trị ngưỡng trên và ngưỡng dưới. Nếu giá trị của pixel thay đổi nhưng không vượt ra 2 ngưỡng này thì vẫn xem như là background. Phương pháp này giải quyết được vấn đề ánh sáng thay đổi so với Background Subtraction.



*Hình 2.5.1: Average Background Method*

### c) Codebook Method:

Là một phương pháp nâng cao áp dụng cho các background chứa các vật có chuyển động phức tạp như lá cây đong đưa trong gió, cánh quạt xoay, sự lay động của tấm màn... Hoặc sự thay đổi ánh sáng do các đám mây di chuyển.

Phương pháp này sẽ so sánh giá trị hiện tại với giá trị trước đó của mỗi pixel. Nếu giá trị mới gần với giá trị cũ sẽ xem như là nhiễu loạn nhỏ và vẫn thuộc background. Trong thực tế ta thường chuyển ảnh RGB sang YUV hoặc HSV và chỉ xử lý trên trục V (cường độ sáng).

Mỗi pixel sẽ có một codebook theo dõi các giá trị trước của background. Khái niệm codebook: là một tập dạng các box (hộp) bao quanh giá trị mới của pixel, đỉnh và đáy box là ngưỡng trên và ngưỡng dưới của giá trị (độ sáng) mới của pixel đó.

Khi giá trị mới của pixel thay đổi, đỉnh và đáy của box tăng hoặc giảm để bao phủ giá trị mới đó. Nếu giá trị pixel thay đổi quá lớn thì một box mới sẽ được tạo ra tại giá trị đó và tiếp tục bao phủ các giá trị mới tiếp theo của pixel (xem hình 2.5.2), đồ thị trên biểu diễn giá trị của 1 pixel theo thời gian, đồ thị dưới mô tả các box xuất hiện và bao phủ giá trị của pixel).

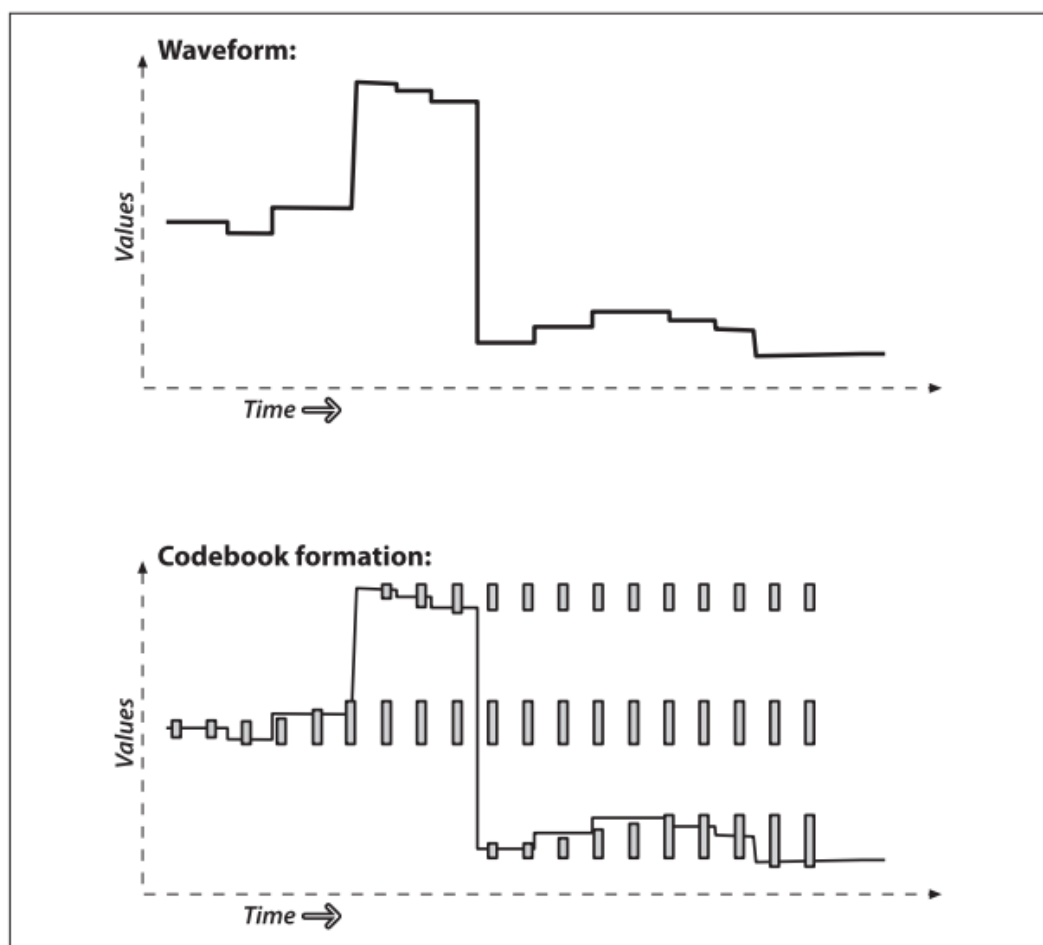


Figure 9-3. Codebooks are just "boxes" delimiting intensity values: a box is formed to cover a new value and slowly grows to cover nearby values; if values are too far away then a new box is formed

Hình 2.5.2: Giá trị của 1 pixel theo thời gian

Codebook được tạo khi chương trình bắt đầu chạy.

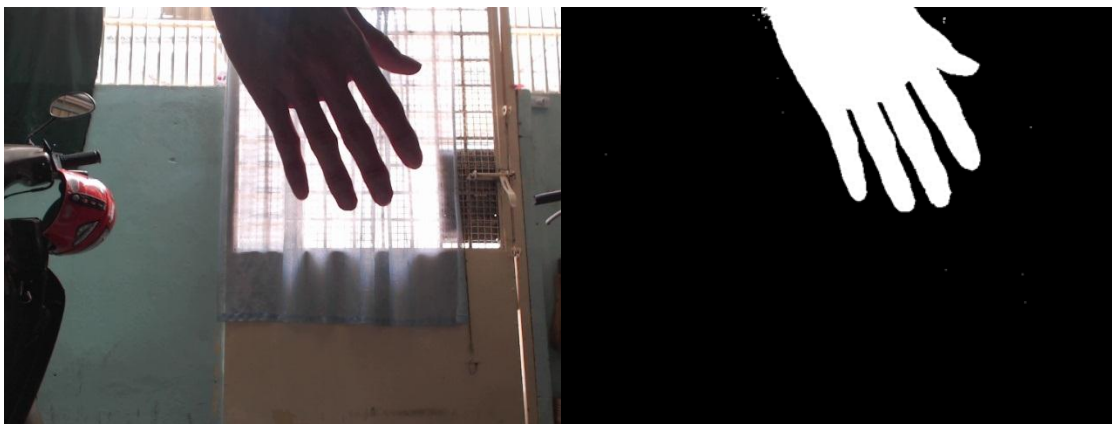
Tuần hoàn sau một khoảng thời gian ngắn (vài giây hoặc vài phút) codebook tự động cập nhật các giá trị mới, tạo thêm box nếu giá trị của pixel thay đổi lớn.

Tuần hoàn sau một khoảng thời gian dài hơn (vài phút) codebook tự động xóa các giá trị cũ, xóa các box có giá trị không đổi.

Khi giá trị mới của pixel nằm ngoài các box thì xem như thuộc foreground.

Phương pháp này cũng giải quyết được trường hợp một vật bị di chuyển sang vị trí khác và đặt cố định trong một thời gian dài bằng cách cập nhật codebook sau một khoảng thời gian ngắn, ta được một background mới.

Phương pháp codebook làm việc tốt trong nhiều điều kiện. Thời gian học cũng nhanh so với các phương pháp khác. Tuy nhiên vẫn bị hạn chế đối với các quang cảnh bị thay đổi toàn cục như ánh nắng ngoài trời hoặc khi bật tắt đèn trong phòng. Với mỗi trường hợp trên cần phải áp dụng các codebook với thuật toán khác nhau cho từng trường hợp cụ thể.



Hình 2.5.3: Codebook Background Method

### So sánh phương pháp Average và Codebook :

Phương pháp codebook lọc nhiễu và cho foreground có hình dạng tốt hơn so với phương pháp average

## 2.4 Theo vết chuyển động (Tracking and Motion)

Sau khi tạo được ảnh foreground chứa các vật quan tâm. Vấn đề thường đặt ra tiếp theo là theo dõi chuyển động của các vật đó sau mỗi frame trong một video. Có rất nhiều thuật toán theo dõi chuyển động. Trong đồ án này chỉ giới thiệu thuật toán Khuôn Mẫu Chuyển Động (Motion Templates).

### Khuôn mẫu chuyển động (Motion Templates):

Motion templates là một thuật toán theo dõi chuyển động tổng thể và nhận dạng động tác rất hiệu quả. Thuật toán sử dụng một ảnh lịch sử chuyển động MHI (Motion History Image) để lưu các bóng (silhouettes) là ảnh của vật ở các frame trước. OpenCV đã cung cấp sẵn cho ta một số hàm để hiện thực thuật toán trên:

```
/* Updates motion history image given motion silhouette */
CVAPI(void)    cvUpdateMotionHistory( const CvArr* silhouette, CvArr* mhi,
                                     double timestamp, double duration );

/* Calculates gradient of the motion history image and fills
   a mask indicating where the gradient is valid */
CVAPI(void)    cvCalcMotionGradient( const CvArr* mhi, CvArr* mask,
                                     CvArr* orientation,
                                     double delta1, double delta2,
                                     int aperture_size CV_DEFAULT(3));

/* Calculates average motion direction within a selected motion region
   (region can be selected by setting ROIs and/or by composing a valid gradient mask
   with the region mask) */
CVAPI(double)  cvCalcGlobalOrientation( const CvArr* orientation, const CvArr* mask,
                                       const CvArr* mhi, double timestamp,
                                       double duration );

/* Splits a motion history image into a few parts corresponding to separate
   independent motions
   (e.g. left hand, right hand) */
CVAPI(CvSeq*)  cvSegmentMotion( const CvArr* mhi, CvArr* seg_mask,
                               CvmemStorage* storage,
                               double timestamp, double seg_thresh );
```

Tất cả các hàm trên chỉ xử lý ảnh đơn sắc (single-channel image).

Hàm `cvUpdateMotionHistory()` tạo ra ảnh MHI từ các ảnh silhouette hiện tại. `timestamp` là thời gian thực của hệ thống (ms) và `duration` là thời gian tồn tại mhi của các bóng cũ trên.

```
double timestamp = (double)clock() / CLOCKS_PER_SEC; // get current time in seconds
```

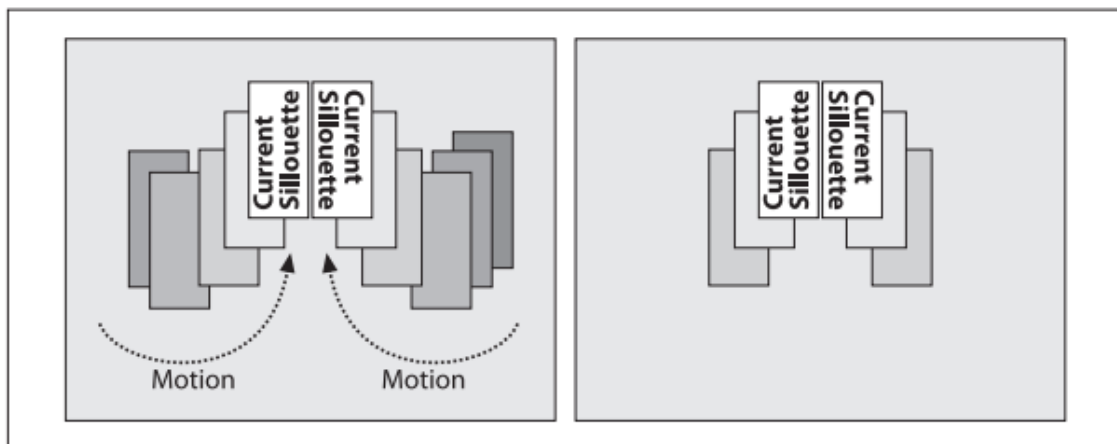


Figure 10-14. Motion template silhouettes for two moving objects (left); silhouettes older than a specified duration are set to 0 (right)

Khi đã có một tập các bóng chuyển động của vật và thời gian chuyển động, ta dễ dàng tính được hướng chuyển động của vật bằng việc tính gradient của ảnh mhi.

Hàm `cvCalcMotionGradient()` dùng để thực hiện việc này. `delta1` và `delta2` cường độ gradient nhỏ nhất và lớn nhất cho phép. Các điểm khác 0 trên ảnh `mask` là vị trí gradient hợp lệ tìm được. Ảnh `orientation` lưu độ lớn gradient tại điểm đó.

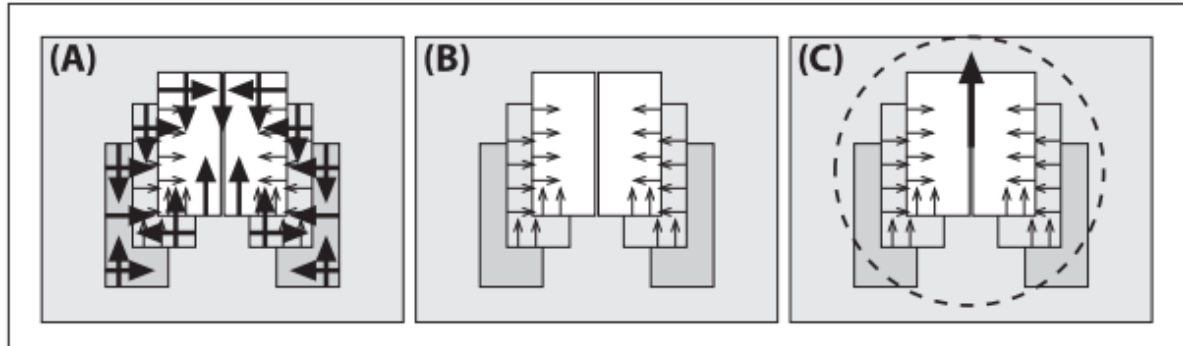


Figure 10-15. Motion gradients of the mhi image: (A) gradient magnitudes and directions; (B) large gradients are eliminated; (C) overall direction of motion is found

Hình 2.6.2: (A) Độ lớn các gradient và hướng. (B) lược bỏ các gradient quá lớn. (C) gradient tổng

Hàm `cvCalcGlobalOrientation()` tính gradient tổng của hướng chuyển động. Tham số `orientation`, `mask` là kết quả trả về của hàm `cvCalcMotionGradient()`, `mhi`, `timestamp`, `duration` từ hàm `cvUpdateMotionHistory()`. Hàm trả về góc của hướng chuyển động (đơn vị degree)

Vì hàm `cvCalcGlobalOrientation()` chỉ trả về 1 tổng gradient cho toàn ảnh, nên ta thường tách các vật chuyển động thành các ảnh nhỏ rồi dùng `cvCalcGlobalOrientation()` để tìm gradient tổng cho từng vật.

Hàm `cvSegmentMotion()` thực hiện việc cách ly các vật chuyển động và trả về đường bao các bóng của vật chuyển động. Tham số `seg_thresh` là thời gian tồn tại của các bóng cũ, tránh trường hợp lấy các bóng quá cũ nằm chồng lên bóng của vật khác.

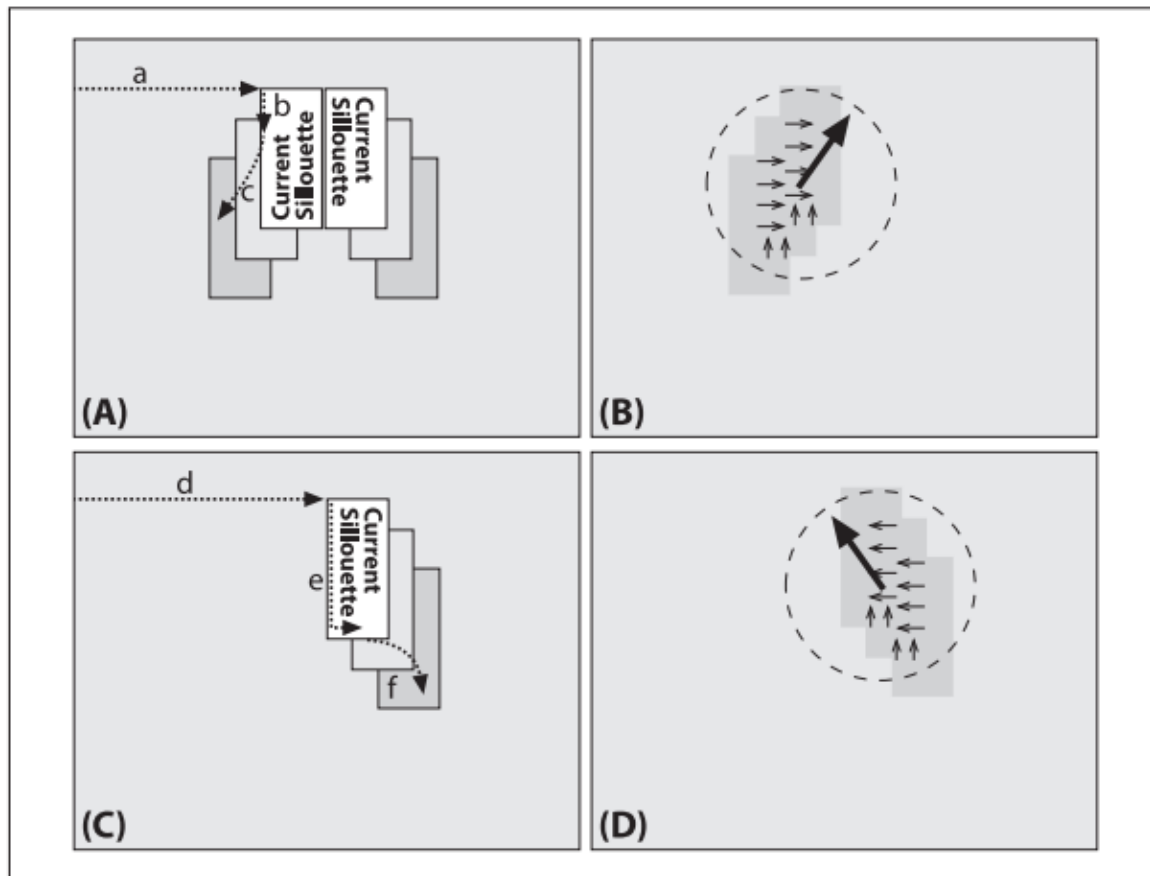


Figure 10-16. Segmenting local regions of motion in the mhi image: (A) scan the mhi image for current silhouettes (a) and, when found, go around the perimeter looking for other recent silhouettes (b); when a recent silhouette is found, perform downward-stepping flood fills (c) to isolate local motion; (B) use the gradients found within the isolated local motion region to compute local motion; (C) remove the previously found region and search for the next current silhouette region (d), scan along it (e), and perform downward-stepping flood fill on it (f); (D) compute motion within the newly isolated region and continue the process (A)-(C) until no current silhouette remains

Hình 2.6.3: Tách các vật chuyển động để tính gradient tổng cho từng vật



### 3. CHỌN THIẾT BỊ PHẦN CỨNG

#### Camera Logitech C920:

- Kích thước: 22x15x5 (cmxcmxcm)
- Kết nối USB2.0
- Hỗ trợ định dạng nén 'H264' và 'MJPG'
- Tích hợp microphone
- Tự động lấy nét, theo dõi khuôn mặt, phát hiện chuyển động
- Chế độ ghi video full HD 1920x1080 pixels.



Hình 3:

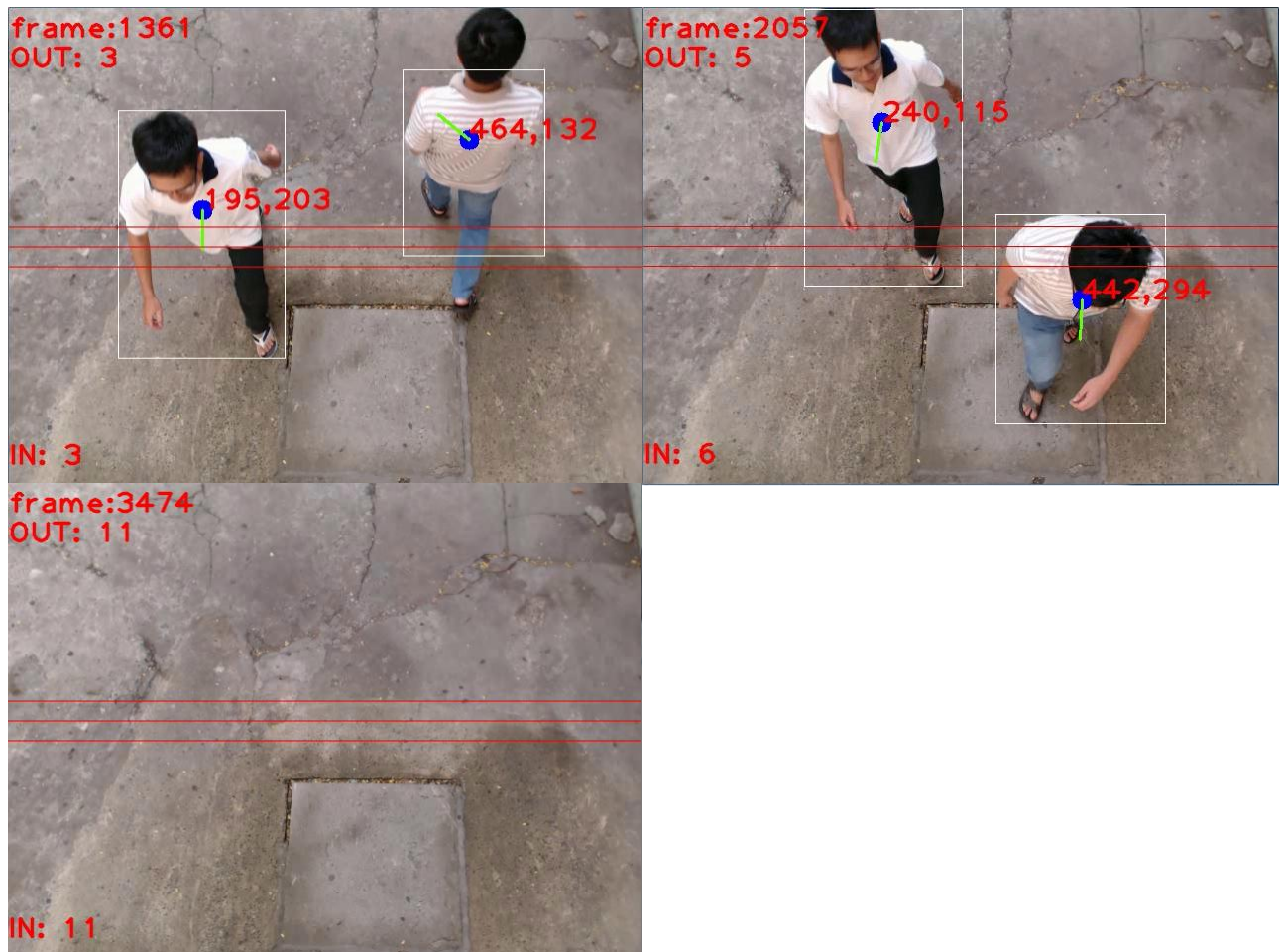
### 4. THIẾT KẾ VÀ THỰC HIỆN PHẦN MỀM

- **Yêu cầu đặt ra cho phần mềm**
  - Phân biệt, tách được vật đang chuyển động và nền.
  - Lọc nhiễu, giảm bóng
  - Tính được hướng chuyển động của vật.
  - Đếm được số người ra hoặc vào.
  - Tốc độ xử lý nhanh.
  - Có thể đếm thông qua một camera hoặc một file .avi
  - Kết quả xử lý được lưu thành một file .avi.
  - Có các phím tắt điều khiển chương trình.
- **Thuật toán xử lý:** bao gồm việc lặp lại 3 bước sau. Với mỗi frame thu được sẽ được chuyển sang ảnh HSV và chỉ xử lý trên mặt phẳng ảnh V (cường độ sáng).
  - Bước 1:
    - Dùng phương pháp trừ nền, sử dụng phương pháp codebook, để tách các vật chuyển động.
    - Ảnh sau khi tách được lọc nhiễu, giảm bóng và loại bỏ các phần ảnh nhỏ hơn kích thước 1 người trung bình
    - Khi chương trình khởi động codebook được tạo bằng việc lấy mẫu khoảng 100 frame đầu tiên hoặc lấy mẫu trong vài giây.
    - Tuần hoàn sau một khoảng thời gian chương trình tự động cập nhật và xóa các dữ liệu trong codebook.
  - Bước 2:
    - Sử dụng thuật toán Motion Templates để tính toán các thành phần chuyển động của ảnh để xác định hướng di chuyển của vật là vào hay ra
    - Với mỗi vật chuyển động sẽ được đánh dấu bởi một hình chữ nhật, một chấm sáng tại tâm của vật và một vạch ngắn thể hiện hướng di chuyển của vật.
  - Bước 3:
    - Dựa vào kết quả hướng di chuyển của vật ở bước 2. Ta tính được hướng số lượng người đi vào và ra khi tâm của vật đi qua một vạch nằm ngang ở giữa khung hình



## 5. KẾT QUẢ THỰC HIỆN

Chương trình chạy thử trên một đoạn video.avi có 3485 frame, kích thước 640x480, tốc độ 24fps. Camera đặt cố định tại lầu 1 cách mặt đất khoảng 4-5 mét, ngoài trời, ống kính đặt hướng xuống mặt đất. Có 2 người liên tục ra, vào bên dưới camera. Trời nhiều mây, không nắng. Kết quả đếm chính xác được 11 lần vào, 11 lần ra.



Hình 5: Kết quả chạy chương trình

## 6. KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

### 6.1 Kết luận

#### **Ưu điểm:**

- Đếm được nhanh, chính xác số người ra và vào, đặc biệt những nơi có dòng người đông.

#### **Khuyết điểm:**

- Chương trình còn phụ thuộc vào điều kiện ở nơi lắp đặt camera như ánh sáng môi trường. Nên cần phải chỉnh lại các thông số cho phù hợp với mỗi nơi.
- Chưa xử lý được các trường hợp người đi sát, công nhau, các vật có kích thước giống kích thước con người

### 6.2 Hướng phát triển

Đề tài có thể phát triển thêm một số ứng dụng nâng cao khác như:

- Đếm mật độ người tại một khu vực.
- Đếm lưu lượng xe trên các tuyến đường cao tốc, có thể tách riêng từng loại xe khác nhau.
- Sử dụng hệ thống 3D có thể vừa đếm vừa phân biệt được trẻ em, người lớn, giới tính

## 7. TÀI LIỆU THAM KHẢO

- [1] Gary Bradski and Adrian Kaehler, “Learning OpenCV”, O’Reilly, 8/2008
- [2] Tạ Thị Ái Nhi, “Phát hiện và theo vết người từ dữ liệu Video”, Tóm tắt Luận văn Thạc Sĩ Kỹ thuật, Đại học Đà Nẵng, 2012
- [3] Nguyễn Văn Long, “Ứng dụng xử lý ảnh trong thực tế với thư viện OpenCV”,
- [4] James Davis & Gary Bradski, “Real-time Motion Template Gradients using Intel CVLib, MIT Media Lab, 1999
- [5] Gagan, “Motion Detection”, <https://singhgaganpreet.wordpress.com/tag/cvboundingrect/>, 2012
- [6] “Camera IP là gì? Camera Analog là gì?”,  
<http://www.vienthonghoanggia.com/camera-ip-la-gi-camera-analog-la-gi-.html>
- [7] Rhondasw, “People counting with top-mounted camera” , <http://www.computer-vision-software.com/blog/2010/03/people-counting-with-top-mounted-camera/>
- [8] “How the Camera Counting System Works”, <http://videoturnstile.com/camera-counting.html>, 2011
- [9] Greg Herno , “How does a video based people counter”,  
<http://www.vinteoinc.com/blog/2015/12/12/how-does-a-video-based-people-counter-work>, 2015