Hệ Thống nhận dạng thức ăn theo thời gian thực sử dụng Fisher Vector

1. Giới thiệu

Trong những năm gần đây, dịch vụ ghi lại thói quen ăn uống trên smartphone trở nên phổ biến.Chúng nhắc nhở người dùng về các thói quen sử dụng đồ ăn không tốt, và xu hướng sử dụng thực phẩm không tốt cho sức khỏe. Do đó, hữu ích cho cho phòng bệnh và chế độ ăn uống hợp lý.Tuy nhiên, hầu hết các dịch vụ này yêu cầu người dùng phải chọn các món ăn từ thực đơn bằng tay,điều này gây phiền hà và tốn thời gian cho hầu hết người dùng để tiếp túc sử dụng dịch vụ trong một thơi gian dài.

Hầu hết các hệ thống nhận dạng ảnh trên điện thoại hiện nay như Google Goggles yêu cầu người dùng gửi ảnh lên máy chủ mạnh hơn để xử lý, điều này làm cho quá trình tương tác chậm trễ, đòi hỏi chi phí truyền dẫn lên máy chủ, và phụ thuộc vào điều kiện mạng internet. Mặt khác, việc nhận dạng ảnh trực tiếp trên điện thoại nhiều hứa hẹn về tính sẵn có, độ trễ do truyền dẫn, chí phí duy trì máy chủ. Do sự phát triển nhanh chóng của điện thoại như hiện nay, chúng đã đủ mạnh để ghi lại và xử lý ảnh theo thời gian thực. Do đó, tận dụng sức mạnh tính toán trên điện thoại cũng như công nghệ nhận dạng ảnh nâng cao,trong demo này, chúng tôi đề xuất 1 hệ thống nhận dạng thực phẩm theo thời gian thực chạy trên điện thoại thông minh phổ biến. Để tăng độ chính xác và tốc độ xử lý, chúng tôi áp dụng Fisher Vector và SVM tuyến tính, và thực hiện tính toán đa luồng để sử dụng cpu 4 lõi hiệu quả.

Kể từ khi thực hiện nhận dạng, hệ thống đề xuất thực hiện thực hiện liên tục 1s/ 1 lần. Người dùng có thể di chuyển camera trên điện thoại 1 cách liên tục để có được ví trí tốt nhất mà hệ nhận dạng một cách chính xác mà không cần phải bấm nút chụp hình. Đây chính là lợi thế hệ thống nhận dạng theo thời gian thực trên điện thoại thông minh.

Trong các thí nghiệm, chúng tôi đã đạt được tỷ lệ phân loại là 79,2% cho 5 loại có độ chính xác cao nhất, vượt trội hơn so với các phương pháp chúng tôi đã có trước đó. Có 3 phương diện cho những cái mới trong hệ thống của chúng tối:

1. Hệ thống cho phép tương tác thời gian thực và ghi lại trên điện thoại

2. Sử dụng Fisher Vector trên điện thoại

3. Tự động điều chỉnh các khung giới hạn nhất định

1. Tổng quan về hệ thống

Mục đính cuối cùng của hệ thống là hỗ trợ người dùng ghi lại những thói quen ăn uống cũng như kiểm tra thói quen dùng thực phẩm của họ. Để làm điều đó dễ dàng, chúng tôi nhúng hệ thống nhận dạng lên múc đích của hệ thống. Luồng xử lý sẽ được thực hiện như sau:

1. Người dùng hướng camera điện thoại về phía thực phẩm trước khi ăn. Hệ thống liên tục đưa các frame xuống cho hệ thống xử lý ngầm
2. Người dũng vẽ các hình khối bao quanh từng Item trên màn hình. Hơn 2 hình khối có thể vẽ cùng lúc.
3. Việc nhận dạng thực phẩm sẽ được thực hiện cho từng khu vực nằm trong các khối bao quanh sử dụng HOG và màu sắc với SVM tuyến tính và Fisher Vector.
4. Kết quả hiện thị năm thực phẩm tỷ lệ nhận dạng chính xác nhất
5. Nếu thấy thực phẩm có trong danh sách, người dùng chọn bằng cách chạm tay vào và ước lượng bằng cách chỉ ra khối lượng của chúng. Nếu không thấy thực phẩm trong danh sách, người nên di chuyển camera và quay lại bước 3.
6. Tên, lượng colarie, thành phần dinh dưỡng hiện thị trên màn hình và được hệ thống ghi lại.

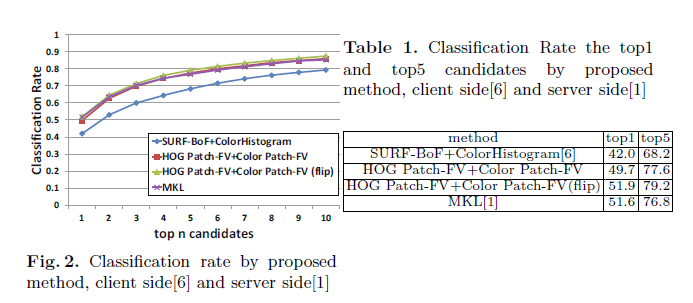
Thêm vào đó, người dùng có thể ghi lại bữa ăn của mình và chi tiết các món ăn với lượng calorie, thành phần dinh dưỡng có trong từng món ăn không chỉ trên màn hình mà còn cả trên website bằng cách gửi dữ liệu lên máy chủ.

1. Thực hiện và đánh giá

Đặc trưng của ảnh: Chúng tôi sử dụng HOG và color cho mã hóa Fisher Vector.

HOG: Histogram of Oriented Gradients được đề xuất bởi N.DALAL. Mô tả HOG là rất đơn giản, nó nhanh hơn, nhiều hơn so với các mô tả như SIFT,SURF. Đây là điều quan trọng để thực hiện nhận dạng theo thời gian thực trên điện thoại. Ngoài ra nó có thể trích xuất đặc trưng với mật độ nhiều hơn. Kết quả là nó cải thiện độ chính xác của việc nhận dạng. Chúng tôi trích xuất HOG như các features local. Chúng tôi chia các local patch thành các khối 2\*2 và trích xuất gradient histogram theo 8 hướng từ mỗi khối. Cuối cùng, chúng tôi trích xuất 32-dim đặc trưng HOG Patch,PCA áp dụng để giảm từ 32-dim thành 24-dim. Đặc trưng màu sắc cũng được trích xuất thành 24-dim.

Fisher Vector: Fisher Vector ít lỗi lượng tử hóa hơn so với BoF. Trong khi, Bof cần 1 từ điển lớn để cải thiện độ chính xác thì Fisher Vector chỉ cần dùng từ điển bé. Mặt khác, dùng từ điển lớn sẽ có chi phí tìm kiếm lớn hơn. Fisher Vector có thể có độ nhận dạng chính xác cao khi sử dụng tử điển bé và thuật toán ít phức tạp hơn. Đây chính là lợi thế cho thiết bị di động. Vì vậy, việc áp dụng Fisher Vector như là một phương pháp mã hóa tốt hơn về nhận dạng chính xác và thời gian xử lý cho nhận dạng đối tượng trên điện thoại. Nhưng các hệ thống nhận dạng trên điện thoại cho đến nay

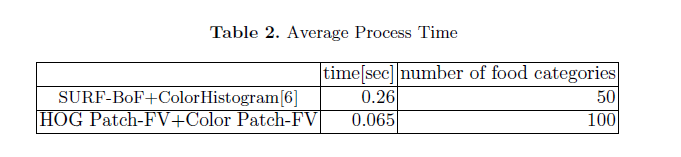


Để cải thiện độ chính xác, chúng tôi áp dụng chuẩn hóa mạnh(α=0.5) và chuẩn hóa L2

Phân loại: Để phân loại, chúng tôi sử dụng SVM tuyến tính. Chúng tôi tiến hành học máy SVM trước đó, Tất cả các tham số sử dụng cho bước nhận dạng được nạp vào bộ nhớ chính.Fisher có thể mang lại kết quả nhận dạng tốt hơn với bộ từ điển nhỏ sơn so với BoF. Chúng tôi cũng thực hiện cài đặt số chiều của vector đặc trưng nhỏ hơn bởi PCA. Kết quả là không gian bộ nhớ cho Fisher Vector nhỏ hơn so với BOF thông thường.

Thực nghiệm: Để triển khai hệ thống, chúng tôi chuẩn bị cơ sở dữ liệu của hơn 100 loại thực phẩm, với mỗi loại có hơn 100 ảnh mẫu. Tất cả các ảnh mẫu đều được đánh dấu bởi khunng giới hạn. Tổng số ảnh thực tế là 12.905.

Trong thí nghiệm này, chúng tôi so sánh 2 hệ thống nhận dạng đã có trước đó, Một là hệ thống client-side của Kawano và một là hệ thống client-side của Matsuda. Chúng tôi so sánh độ nhận diện chính xác với các hệ thống nhận dạng thực phẩm đã có trước đó. Phương pháp đề xuất của chúng tôi là (HOG Patch-FV+Color Patch-FV) and (HOG Patch- FV+Color Patch-FV(flip)) , phương thức nhận dạng client-side (SURF-BoF+color histogram) và phương thức nhận dạng phía máy chủ. Lưu rằng "flip" ở đây là thêm vào bộ huấn luyện những ảnh xoay theo chiều ngang và xoay ngược.



HOG Patch-FV+Color Patch-FV đạt được 49.7% và 77.6% tỷ lệ phân loại với top1 và top5 tương ướng. Trong trường hợp thêm vào các ảnh "flip" thì tỷ lệ top1 và top5 tương ứng là 51.9% và 79.2%. Cách tiếp cận của chúng tôi là tốt hơn so với hệ thống nhận dạng của Matsuda, cái mà có chi phí cao hơn chạy ở phía máy chủ. Vì vậy, chúng tôi cho thấy hiệu quả từ phương pháp nhận dạng của chúng tôi và có thể thực hiện chúng trên điện thoại thông minh. Tiếp theo, chúng tôi tính toán thời gian thực hiện bằng cách thực hiện 20 lần và tính toán thời gian trung bình của chúng. Chúng tôi sử dụng Galaxy Note 2 (1.6GHz 4 cores, 4 threads, Android 4.1). Chúng tôi cũng cho thấy thời gian nhận dạng của 50 loại thực phẩm khác nhau. Cách tiếp cận của chúng tôi mất 0.065s cho 100 loại,Mặt khác, Phương pháp kia mất 0.26s cho 50 loại. Vì vậy phương pháp chúng tôi đề xuất nhanh hơn nhiều so với phương pháp kia.Và kết quả chỉ ra rằng phương pháp đề xuất này hoàn toàn phù hợp với hệ thống nhận dạng thời gian thực.Qua các thực nghiệm về tính chính xác và thời gian xử lý, chúng tôi đã cho thấy sự hiểu quả phương pháp chúng tôi đề xuất. Vì vậy chúng tôi cho thấy rằng việc nhận dạng anh nhanh chóng trên điện thoại là hoàn toàn có thể.