

CHƯƠNG TRÌNH DÒ TÌM CHUYỂN ĐỘNG CỦA MÔI CÀI ĐẶT TRÊN FPGA

LIP TRACKING PROGRAM IMPLEMENTED ON FPGA

Võ Thị Thu Hồng, Lê Quốc Bảo Trí, Nguyễn Ngọc Tài, Lê Trung Hiếu

Trường Đại học Bách khoa, TP Hồ Chí Minh; votthuhong@hcmut.edu.vn

Tóm tắt - Việc dò tìm chuyển động của môi là giai đoạn đầu tiên mang tính quyết định đối với hệ thống nhận dạng thính thị (AVSR). Đây là quá trình phân tách đường viền môi từ các chuỗi video của khuôn mặt. Đã có nhiều giải thuật rất hữu hiệu được đề xuất để cho kết quả phân đoạn chính xác vùng môi. Do độ phức tạp trong tính toán nên đa số các giải thuật này đều được thực hiện hoàn toàn bằng phần mềm. Trong bài viết này, chúng tôi trình bày mô hình mới sử dụng phần cứng để dò tìm chuyển động môi, cài đặt trên FPGA. Công cụ toán học ở đây kết hợp giải thuật phân cụm K-Means và giải thuật tìm đường đi ngắn nhất của Dijkstra. Kết quả thử nghiệm thích hợp với khuôn mặt có sự xuất hiện của râu, răng và xử lý tốt trong trường hợp màu môi không khác biệt nhiều so với vùng ngoài môi. Cấu trúc phần cứng được thiết kế pipeline cho phép tăng đáng kể tốc độ xử lý so với phần mềm (hơn 700 frames/s).

Từ khóa - dò chuyển động môi; phân đoạn vùng môi; FPGA; phân cụm K-Means; giải thuật Dijkstra; pipeline.

1. Đặt vấn đề

Tự động nhận dạng tiếng nói (ASR: Automatic Speech Recognition) luôn là đề tài được các nhà nghiên cứu trên thế giới quan tâm từ nhiều năm nay. Để nâng cao độ chính xác, người ta còn phối hợp thêm kỹ thuật “đọc” bằng hình ảnh dùng phương pháp nhận dạng tiếng nói thính thị AVSR. Phương pháp này chủ yếu dựa trên hình ảnh thay đổi của môi để bổ sung cho các âm từ bị trùng lặp khi phát âm, hoặc bị biến dạng âm sắc, hoặc bị suy hao do môi trường. Kỹ thuật nhận dạng AVSR được ứng dụng nhiều để hỗ trợ cho người khiếm thính, dùng trong an ninh hay bảo mật. Đối với AVSR, việc dò theo chuyển động của môi đóng vai trò quyết định trong thu thập dữ liệu nhận dạng tiếng. Tuy đây không phải là đề tài mới, nhưng thường xuyên được cập nhật để cải tiến do khả năng ứng dụng thực tiễn cao. Nhiều công trình và bài báo liên quan đã được thực hiện và công bố rộng rãi trên thế giới. Tại Việt Nam cũng có một số các công trình nghiên cứu trong lĩnh vực này (chủ yếu của ngành công nghệ thông tin) và hầu hết được thực hiện bằng phần mềm trên máy tính.

Dò đường viền môi là tổng hợp của việc phân đoạn vùng môi trên các frame ảnh liên tiếp. Các phương pháp phân đoạn vùng môi đều dựa trên đặc tính ảnh (image-based) hay đặc tính mô hình (model-based) của môi để phân tích. Đặc tính ảnh thường sử dụng trực tiếp thông tin màu đặc trưng của môi trên khuôn mặt. Giải thuật dựa trên đặc tính này [1], [2] thường dễ thực hiện và ít tốn thời gian. Nhược điểm của phương pháp này là dễ bị tác động bởi điều kiện độ sáng khác nhau. Đặc tính mô hình bao gồm các thông tin đã biết về hình dạng thông thường của môi. Giải thuật được tiếp cận nhiều nhất là đường viền động quanh co (Snake/Active Contour) [3], [4]. Giải thuật này cho kết quả tốt hơn nhưng tốc độ chậm hơn do tốn nhiều thời gian cho các vòng lặp (để tối thiểu hàm năng lượng).

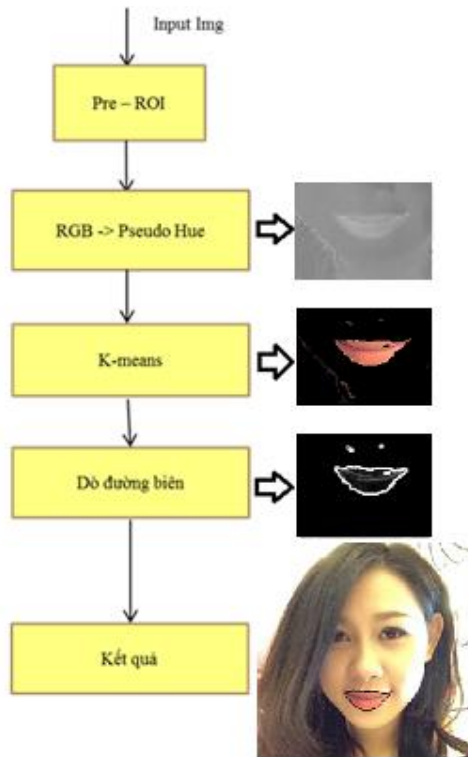
Abstract - The first and crucial stage in audio visual speech recognition (AVSR) system is lip tracking. It is the process of partitioning lip contour from facial video sequences. Many efficient algorithms have been developed to achieve good and accurate segmentation. Most of these have to be performed entirely by software because of their complex computation. In this paper, we present a novel hardware- based lip tracking model, implemented on DE2 FPGA (Field Programmable Gate Array) Board of Altera. Our mathematical tool is the combination of K-Means clustering and shortest path Dijkstra algorithms. The experimental results have proved that our proposed technique is well adapted to face with the presence of beard, teeth and even to lips with weak color contrast. The pipeline design in our hardware structure considerably enhances the processing speed (more than 700 FPS) in comparison with software. (FPS: Frames per second).

Key words - Lip tracking; lip segmentation; FPGA; K-Means clustering; Dijkstra; pipeline.

Mặt khác để đạt kết quả như mong muốn thì đường bao ban đầu phải được khởi tạo thích hợp. Hiện nay, người ta thường chọn giải pháp kết hợp nhiều phương pháp xử lý để cho độ chính xác cao hơn [5], [6].

Trong bài viết này, chúng tôi trình bày hướng thiết kế được viết theo ngôn ngữ lập trình phân cứng (HDL: Hardware Description Language) cài đặt trên chip FPGA. Phương pháp được chọn để xử lý dựa trên thông tin màu, đặc tính và vị trí hình học của môi trên khuôn mặt. Việc lập trình trên FPGA tuy phức tạp hơn nhiều so với lập trình trên máy tính (xét cho cùng một thuật toán) nhưng tốc độ làm việc của FPGA cao hơn và tiêu thụ năng lượng ít hơn. Các dòng FPGA hiện nay đều thích hợp với các ứng dụng xử lý ảnh trong thời gian thực. Đây là điều mà hầu như máy tính khó đáp ứng được. Trước khi dò đường viền môi, chúng tôi chọn thuật toán phân cụm K-means để tách vùng môi và ngoài môi dựa trên độ sai lệch về màu. Kết quả sau K-means sẽ phân định frame hình thành hai tập hợp, tập các vùng “được cho” là vùng môi và tập các vùng ngoài môi. Trong bước xử lý tiếp theo, giải thuật tìm đường đi ngắn nhất của Dijkstra được sử dụng để tìm đường viền môi, đồng thời cũng có tác dụng loại bỏ các thành phần thừa ở vùng chung quanh môi mà K-means chưa lọc hết được. Trình tự xử lý trên từng frame được cho ở Hình 1.

Nội dung bài viết gồm 6 phần. Phần 1 giới thiệu ý tưởng khoa học của đề tài, tổng quan về quá trình nghiên cứu trong và ngoài nước, đề ra giải pháp dò đường biên của môi áp dụng trên phần cứng (viết bằng ngôn ngữ Verilog). Phần 2 là phần tiền xử lý tín hiệu đầu vào, chuyển đổi không gian màu thích hợp. Phần 3 trình bày giải thuật phân cụm K-means và ứng dụng để phân biệt vùng môi và các vùng khác ngoài môi trên frame ảnh. Phần 4 nêu giải thuật tìm đường đi ngắn nhất được kết hợp để dò theo đường viền môi. Phần 5 và 6 là kết quả thử nghiệm và kết luận.



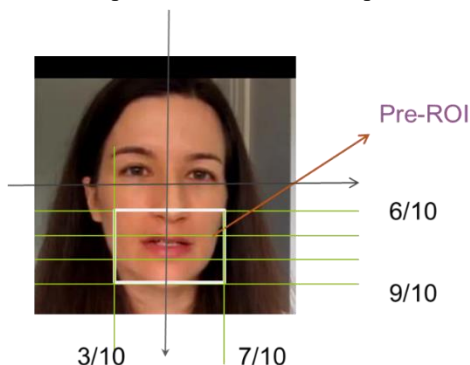
Hình 1. Trình tự xử lý

2. Giai đoạn tiền xử lý

Dữ liệu video vào được tách thành chuỗi frame màu RGB. Kích thước mỗi frame ảnh là 256 x 256, thể hiện toàn khuôn mặt, từ đỉnh đầu đến dưới vai. Để tăng tốc độ và tiết kiệm tài nguyên trên kit DE2, chỉ riêng vùng Pre-ROI (Hình 2) mỗi frame ảnh được xử lý. Vùng Pre-ROI được chọn tự động để thu hẹp phạm vi xử lý, nhưng vẫn đủ lớn để đảm bảo vẫn chứa vùng môi trong trường hợp nhân vật trong video nghiêng lắc đầu khi phát âm. Màu RGB trên vùng khảo sát sẽ được chuyển sang không gian màu thích hợp để làm nổi vùng môi so với những vùng khác. Hệ màu RGB tuy thông dụng nhưng chứa đồng thời thông tin màu và độ sáng. N. Eveno đã đề nghị hệ màu pseudo hue [7] phù hợp với màu môi người và tương đối ổn định với các thành phần ánh sáng khác nhau. Công thức chuyển đổi được cho ở (1).

$$H(x, y) = \frac{R(x, y)}{R(x, y) + G(x, y)} \quad (1)$$

Với $H(x, y)$ là giá trị màu pseudo hue, $R(x, y)$ và $G(x, y)$ lần lượt là thành phần Red và Green của pixel ở vị trí (x, y) .



Hình 2. Vùng Pre-Roi

Kết quả ngõ ra của ma trận ảnh trong không gian màu chuyển đổi quyết định nhiều đến chất lượng xử lý của phần sau. Do đó, biểu thức toán trong (1) không thể được thực hiện theo phép chia thông thường vì sẽ gây sai số lớn khi cài đặt lên phần cứng. Giải thuật chia sẽ được thực hiện như sau:

+ Gán giá trị tử số là R, độ lớn xử lý là 18bits.

+ Gán giá trị mẫu số là (R+G), độ lớn xử lý là 18 bits.

+ Vòng lặp: dịch trái R ($\ll i+1$), so sánh $R(\ll i+1)$ và $(R+G) (\ll i)$. Kết quả là $d_0[i]$ (với $i = 9, 8, \dots, 2$), nếu:

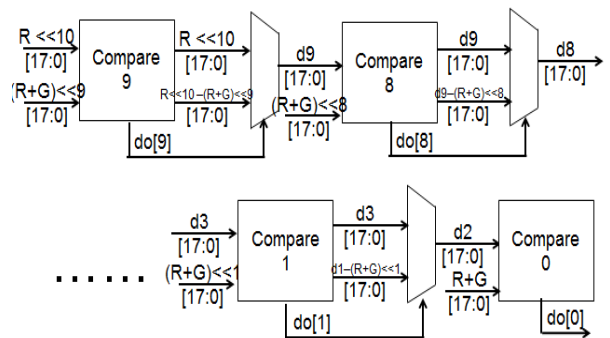
$$\begin{cases} R > (R+G) \rightarrow d_0[i] = 1 \text{ và } R = R - (R+G) \\ R < (R+G) \rightarrow d_0[i] = 0 \text{ và } R = R \end{cases}$$

$$(R+G) = (R+G) \ll i$$

$$+P_Hue = d_0[9:0]$$

Giải thuật tính $H(x, y)$ trên phần cứng:

RGBtoPseudoHue



Hình 3. Sơ đồ thiết kế phần cứng cho pseudo hue

Hình 3 hiển thị sơ đồ thiết kế phần cứng giải thuật tính giá trị pseudo hue cho FPGA. Khối RGBtoPseudoHue được thiết kế sử dụng phép toán dịch bit, so sánh và các bộ mux để thực hiện phép chia theo (1).

3. Giải thuật phân cụm K-means

Giải thuật K-means là một trong những giải thuật phân cụm được sử dụng phổ biến trong phân đoạn ảnh [8], [9] hiện nay do tính toán đơn giản, nhưng hiệu quả. Đặc biệt là trong trường hợp số lượng cần phân cụm ít. Giải thuật K-means được áp dụng trên ảnh pseudo hue để phân vùng môi và ngoài môi.

Giải thuật K-means được tính toán sử dụng các phép toán cộng, trừ và bộ so sánh các giá trị khoảng cách. Giá trị trọng tâm ban đầu (centroid) của vùng ngoài môi B_0 -sam được tính toán dựa trên giá trị trung bình của các pixel trên hàng đầu của vùng Pre-ROI. Từ đó, tính giá trị trọng tâm ban đầu của vùng môi là:

$$F_0\text{-sam} = B_0\text{-sam} + dev \quad (2)$$

Với dev là độ lệch, có giá trị phụ thuộc độ tương phản giữa môi và da.

B-sam (Background sample) và F-sam (Foreground sample) được cập nhật sau mỗi hàng quét của K-means. Tại hàng thứ x, ta có:

$$B_sam = \frac{\sum_{y=0}^{n_B-1} g_B(x, y)}{n_B} \quad (3)$$

$$F_sam = \frac{\sum_{y=0}^{n_F-1} g_F(x, y)}{n_F} \quad (4)$$

Trong đó, (g_B, n_B) và (g_F, n_F) lần lượt là giá trị pixel và số lượng pixel của vùng ngoài môi và vùng môi.

Giải thuật:

Vòng lặp: i hàng, j cột ($i, j = 0, \dots, 255$)

$Sub_B = |Phue(i, j) - B_sam|$;

$Sub_F = |Phue(i, j) - F_sam|$;

Nếu $Sub_B > Sub_F$ % vùng môi

$n_F = n_F + 1$;

$g_F = g_F + Phue(i, j)$;

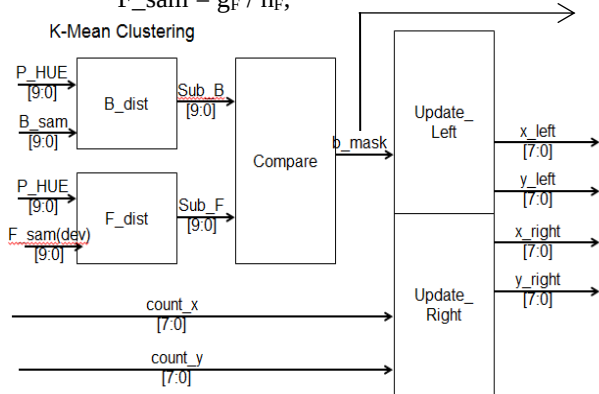
Nếu $Sub_B < Sub_F$ % vùng da

$n_B = n_B + 1$;

$g_B = g_B + a(i, j)$;

$B_sam = g_B / n_B$;

$F_sam = g_F / n_F$;



Hình 4. Sơ đồ phần cứng giải thuật K-Means

Hình 4 cho sơ đồ phần cứng K-means clustering. Trong đó, b-mask là ma trận ảnh ngõ ra của K-means. Ngoài ra, khối update left và update right được thiết kế để tính toán các giá trị mép trái và phải của vùng môi. Vì môi có dạng hình ellipse, do đó 2 mép được xác định bằng cách tính khoảng cách dài nhất từ trung tâm môi sang 2 biên (Hình 5). Các giá trị này sẽ được sử dụng trong khối tìm đường đi ngắn nhất được trình bày ở phần sau.

4. Dò đường biên

Dùng kỹ thuật phân đoạn ảnh Livewire (hay Intelligent Scissors) [10], [11] để dò đường viền môi. Kỹ thuật này dựa trên giải thuật tìm đường đi ngắn nhất của Dijkstra. Trong nghiên cứu này, chúng tôi sử dụng các tọa độ tìm được ở ngõ ra khối K-means để xác định 4 vùng khảo sát mới (I, II, III và IV) trong Pre-ROI như Hình 5.

Ảnh Mask sau K-means được chuyển thành gradient:

$$G = \frac{1}{2} (|G_x| + |G_y|) \quad (5)$$

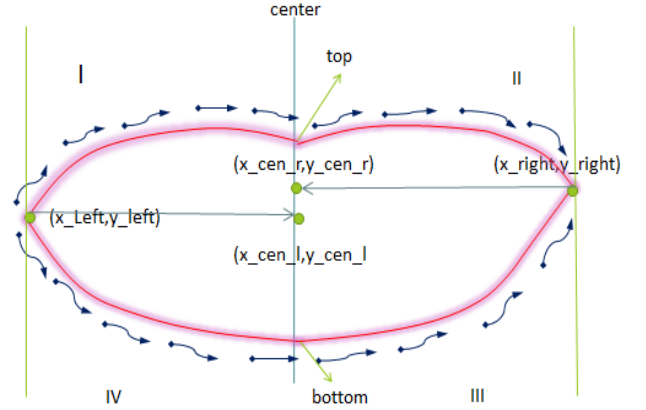
Với G_x và G_y lần lượt là gradient theo phương x và y của ma trận ảnh Mask.

Lập bản đồ trọng số trên mỗi pixel trong vùng khảo sát theo công thức sau:

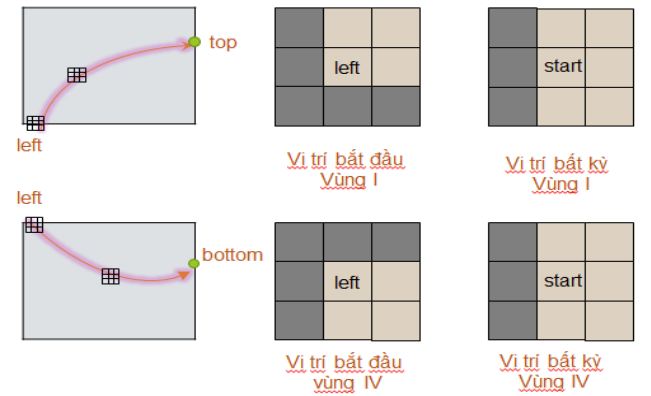
$$W_{(x,y)} = G_{\max} - G_{(x,y)} \quad (6)$$

Trong (6), $G_{\max}$ là giá trị gradient lớn nhất trên vùng khảo sát, $G_{(x,y)}$ là giá trị gradient của pixel tại vị trí (x,y) . Những điểm trên đường viền môi sẽ tương ứng với những

điểm $W_{(x,y)}$ cho đường đi ngắn nhất. Khối dò đường được thiết kế để xử lý lần 1 cho 2 vùng nửa môi trái và lần 2 cho nửa bên còn lại. Mỗi bước dò đường được thực hiện trên một cửa sổ 9 pixel, các ô bôi đen trên Hình 6 là ô không cần xử lý (được gán giá trị $W_{(x,y)} = G_{\max}$).



Hình 5. Vùng khảo sát của giải thuật dò đường



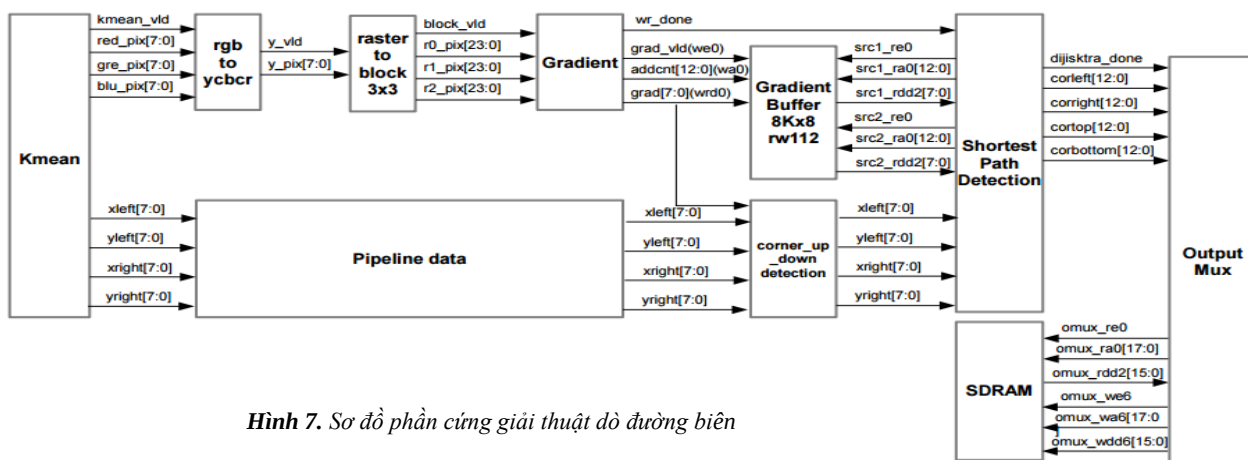
Hình 6. Giải thuật dò đường nửa môi trái

Sơ đồ cấu trúc phần cứng của khối tìm đường viền môi được hiển thị trên Hình 7. Khối này gồm có 4 khối chính: Khối tính toán giá trị Gradient của từng điểm ảnh trên ma trận b-mask, khối Buffer để lưu trữ các giá trị Gradient, khối tìm đường đi ngắn nhất (shortest path detection) dựa trên giải thuật Dijkstra, và cuối cùng là khối Mux để kết hợp ảnh gốc và Mask đường viền môi.

Đầu tiên, b-mask từ ngõ ra khối Kmeans được chuyển từ không gian màu RGB sang YCbCr. Thành phần Y được dùng để tính Gradient. Do dữ liệu ảnh chạy theo dạng Raster, trong khi khối tính Gradient theo phương pháp Sobel lại tính toán theo từng khối 3x3, nên khối Raster to Block 3x3 được đưa vào để thực hiện công việc đó. Sau đó, các giá trị Gradient được ghi vào Buffer.

Khối tìm đường đi ngắn nhất sử dụng các giá trị mép trái, mép phải từ ngõ ra khối Kmeans, đọc các giá trị Gradient từ Buffer (khi wr_done bật lên 1) để tìm đường đi ngắn nhất. Kết quả, chúng ta sẽ có một ma trận Mask đường viền môi.

Khối Output Mux sẽ đọc ảnh gốc từ SDRAM, kết hợp ảnh gốc với ma trận Mask đường viền môi để cho ảnh kết quả. Cuối cùng đọc ảnh sau khi xử lý lên máy tính để so sánh và đối chiếu kết quả trên ảnh gốc.



Hình 7. Sơ đồ phân cứng giải thuật dò đường biên

5. Kết quả thử nghiệm

Phần cứng được cài đặt trên board DE2 của Altera. Kết nối dữ liệu vào/ra giữa máy tính và board DE2 thông qua giao tiếp RS232 và sau đó chứa trong SDRAM. Các nguồn dữ liệu này được lấy trên kho dữ liệu miễn phí trực tuyến và vẫn giữ nguyên chất lượng ban đầu, chỉ thay đổi thành kích thước 256x256. Thử nghiệm cho kết quả tốt ngay cả khi màu môi không tương phản nhiều so với vùng ngoài môi. Giải thuật áp dụng cho các trường hợp nhân vật là nam, nữ, có râu hay không có râu, các trạng thái mở miệng, có răng, nghiêng lắc đầu khi nói (Hình 8). So với các giải thuật phần mềm trước đây trên máy tính, giải thuật của chúng tôi xử lý được khi có sự xuất hiện của lưỡi (màu sắc của lưỡi thường bị phần mềm nhầm với môi) và không cần khống chế tốc độ nghiêng, lắc của nhân vật khi phát âm (vì mỗi frame được xử lý hoàn toàn độc lập nhau). Với tần số 50MHz trên kit DE2 và kích thước ảnh 256x256, tốc độ của phần cứng cho phép xử lý hơn 700 frames/s, nhanh hơn rất nhiều so với các phần mềm cùng chức năng, thực hiện trên máy tính. Do giải thuật xử lý dựa trên đặc tính màu ảnh nên sai số chỉ xảy ra khi vùng môi và vùng biên không có ranh giới (không lấy được gradient), vì ánh sáng làm mất một phần đường viền môi. Kết quả tổng hợp trên phần cứng (Bảng 1) cho thấy nguồn tài nguyên trên kit được sử dụng tương đối hiệu quả và có khả năng mở rộng thêm các bộ xử lý song song để tăng tốc độ làm việc.

Bảng 1. Kết quả tổng hợp trên phần cứng

Synthesis Tool	Quartus 32-bit V 11.1 Web Edition
Device	Cyclone II EP2C35F484C6
Logic Elements	2972/33,216
Register	2377
RAM blocks(M4K)	46/105
SRAM	512Kbytes(256Kx16bits)
Fmax	134MHZ

6. Kết luận

Bài báo trình bày một phương pháp dò đường viền môi cái tiến và cấu trúc phần cứng tương ứng. Phương pháp này kết hợp giải thuật phân cụm K-means và giải thuật tìm đường đi ngắn nhất của Dijkstra để phân biệt vùng môi với các vùng khác trên frame ảnh, các tọa độ mép trái và mép phải của môi được định vị tự động. Phần cứng được thiết kế theo cấu trúc Pipeline và được hiện thực trên FPGA (Cyclone II của Altera) nên đạt được tốc độ xử lý nhanh hơn so với phần mềm, áp dụng tốt trong xử lý video thời gian thực. Với kết quả tổng

hợp trên phần cứng, tần số Fmax cho thấy giải thuật có thể chạy được trên kit có xung clock tối đa 134 MHz (khi đó tốc độ xử lý frames/s sẽ tăng lên thêm rất nhiều). Với nhiều nguồn dữ liệu khác nhau trên internet, giải thuật xử lý được với những trường hợp độ sáng khác nhau, tỷ lệ khuôn mặt khác nhau và khi độ tương phản giữa vùng môi và da không cao (do giá trị trọng tâm trong phép tính K-means được cập nhật liên tục). Đối với kỹ thuật Livewire khi phân đoạn ảnh, điểm xuất phát và kết thúc không cần chọn bằng tay theo cách làm thông thường như trên các phần mềm xử lý ảnh trên máy tính mà được định vị tự động. Kết quả thử nghiệm của chúng tôi chứng minh rằng việc sử dụng phương pháp đề nghị trong việc dò đường viền môi cho kết quả tốt về mặt chất lượng hình ảnh và hiệu suất dò đường viền môi khá cao (85%).

Lời cảm ơn

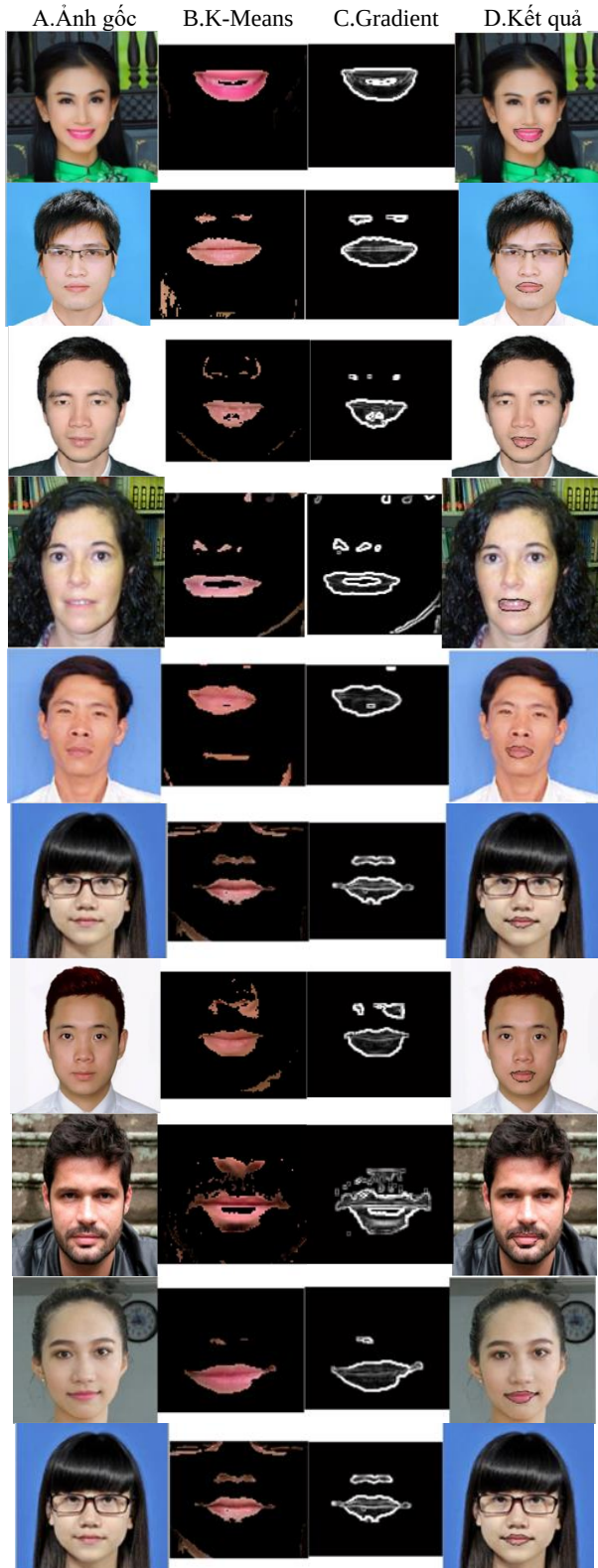
Nghiên cứu này được tài trợ bởi Trường Đại học Bách khoa – Đại học Quốc gia tp. Hồ Chí Minh, trong khuôn khổ đề tài khoa học cấp trường: “Chương trình dò tìm chuyển động của môi, cài đặt trên FPGA”, mã số T- ĐĐT- 2014- 21.

TÀI LIỆU THAM KHẢO

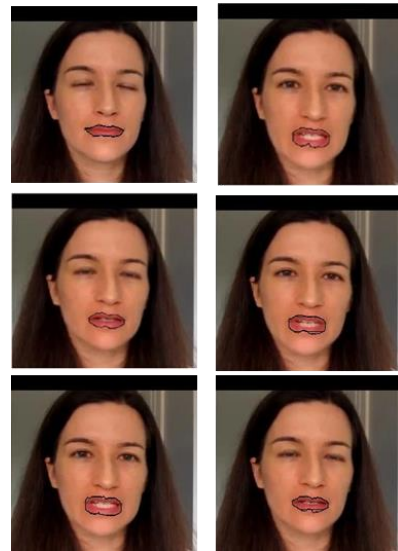
- [1] C.B.O.Lopes, A.L.Goncalves, J.scharcanski and C.R.Jung, “ Color based lip extraction applied to voice activity detection”, 18th IEEE International Conference on image processing, 2011
- [2] Erol Ozgur, Berkay Yilmaz, Harun Karabalkan, Hakan Erdogan, Mustafa Unel, “ Lip segmentation using adaptive color space training”, Faculty of Engineering and Natural Sciences, Sabanci University, Istanbul Turkey
- [3] Mark Banard, Eun -Jung Holden, Robyn Owens, “Lip tracking using pattern matching snakes”, ACCV 2002: The 5th Asian Conference on Computer Vision, pp. 23 – 25, Jan. 2002.
- [4] Xin Liu and Yiu-ming Cheung, “A robust lip tracking algorithm using localized color active contours and deformable models”, ICASSP, page 1197- 1200, 2011
- [5] Tian,Y., Kanade, T., Cohn, J., “ Robust lip tracking by combining shape, color and motion” In: Proc. ACCV, pp.1040 – 1045, 2000
- [6] Salah Werda, Walid Mahdi and Abdelmajid Ben Hamadou, “Automatic hybrid approach for lip POI localization: Application for lip-reading system”, ICTA’07, April 12-14, Hammamet, Tunisia
- [7] N.Eveno, A.Caplier, P.-Y.Coulon, “ A new color transformation for lip segmentation”, Laboratoire des images et signaux, Institut National Polytechnique De Grenoble
- [8] Ms.Chinki Chandhok, Mrs. Soni Chaturvedi, Dr.A.A khurshid, “An approach to image segmentation using K-means clustering algorithm”, IIJT, Vol.1, Aug. 2012
- [9] Vijay Jumb, Mandar Sohani, Avinash Shrivas, ”Color image segmentation using K-means clustering and Otsu’s adaptive thresholding”, IIITEE, Vol. 3, Feb. 2014

- [10] Eric N.Mortensen, William A. Barrett,"Intelligent scissors for image composition", unpublished
- [11] Xianghua Li, Hyo-Haeng Lee, Kwang-Seok Hong," Leaf contour extraction based on an intelligent scissor algorithm with complex background", 2nd International Conference on Future Computers in Education, lecture Notes in Information Technology, Vols. 23-24, 2002

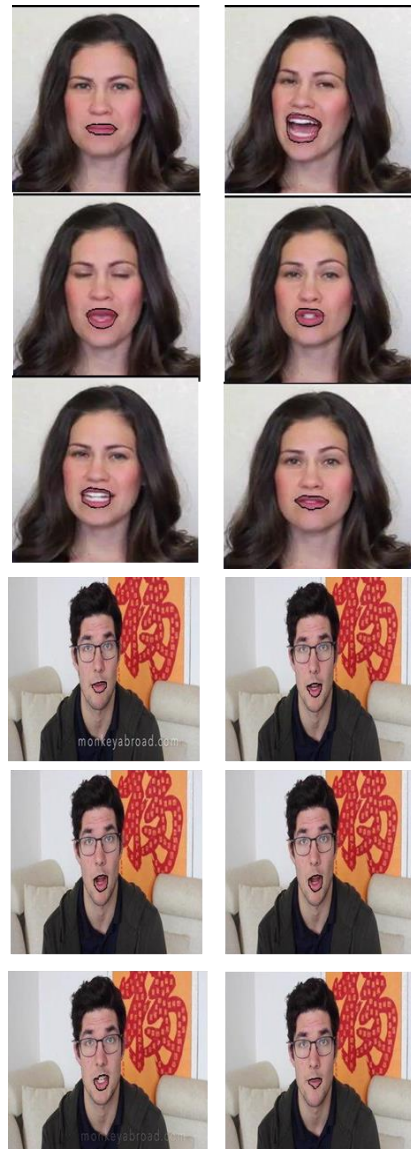
Hình 8. Kết quả thử nghiệm



Kết quả thử nghiệm trên video



Kết quả thử nghiệm trên video



(BBT nhận bài: 22/07/2015, phản biện xong: 29/12/2015)