

ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN
KHOA CÔNG NGHỆ THÔNG TIN



PROTOTYPE

ĐỀ TÀI: BABY FACE GENERATOR

❖ **Họ và tên các thành viên:**

1/ Dương Đăng Khoa MSSV: 1512251
2/ Đặng Thành Phát MSSV: 1512390

❖ **Thông tin giảng viên:**

GVLT: Trần Thái Sơn
GVTH: Võ Hoài Việt

Ngày 27 tháng 04 năm 2018

MỤC LỤC

1	Thông tin nhóm	3
2	Mô tả bài toán	3
3	Tổng quan yêu cầu	4
4	Sơ lược thuật toán	5
5	Bản mẫu (Prototype)	6
6	Tham khảo	8

1

Thông tin nhóm

MSSV	Họ Tên	Email	Điện thoại
1512251	Dương Đăng Khoa	khoadangduong63@gmail.com	0961604198
1512390	Đặng Thành Phát	dtphat.itus@gmail.com	0964427065

2

Mô tả bài toán

- Bài toán Baby Face Generator:
 - + Input: Người dùng nhập vào 2 ảnh của cha và của mẹ, trên mỗi bức ảnh chỉ có duy nhất một khuôn mặt.
 - + Output: Chương trình sẽ phát sinh ra khuôn mặt người con có nét giống với khuôn mặt của cha và mẹ
- Môi trường hoạt động: Windows
- Ngôn ngữ lập trình: C/C++
- Thư viện nguồn mở kèm theo: OpenCV, Dlib. Link download thư viện:
 - + OpenCV: https://sourceforge.net/projects/opencvlibrary/files/opencv-win/3.4.1/opencv-3.4.1-vc14_vc15.exe/download
 - + Dlib: <https://sourceforge.net/projects/dclib/files/dlib/v19.7/>

3

Tổng quan yêu cầu

3.1.1. *Đặc tả yêu cầu chức năng*

- Bước 1: Chương trình đầu tiên sẽ yêu cầu nhập đường dẫn đến ảnh của người cha
- Bước 2: Kế đến yêu cầu nhập đường dẫn đến hình ảnh của người mẹ
- Bước 3: Nhập hệ số alpha để biến đổi (hệ số $\alpha \cdot 100\%$ là tỉ lệ phần trăm giống cha, và $(1 - \alpha) \cdot 100\%$ là tỉ lệ phần trăm giống mẹ với $0 < \alpha < 1$)
- Bước 4: Xử lý
- Bước 5: Chương trình xuất ra hình ảnh lai ghép giữa cha và mẹ theo tỉ lệ khuôn mặt của đứa trẻ

3.1.2. *Đặc tả yêu cầu phi chức năng*

- Chương trình xuất ra hình ảnh lai ghép giữa cha và mẹ theo tỉ lệ khuôn mặt của người trưởng thành

4

Sơ lược thuật toán

- **Bước 1:** Đưa vào một ảnh của một đứa bé bất kì làm để làm khung cho khuôn mặt của đứa bé sau này phát sinh ra.
- **Bước 2:**
 - Kiểm tra bức ảnh đó có nhiều hơn 1 khuôn 1 khuôn mặt hay không. Nếu có 2 khuôn mặt trở lên thì xuất ra thông báo, kết thúc chương trình.
 - Ngược lại thì tiếp tục chương trình, lấy tọa độ điểm đặt trung khuôn mặt
- **Bước 3:** Nhập ảnh của bố đứa trẻ.
- **Bước 4:**
 - Kiểm tra bức ảnh đó có nhiều hơn 1 khuôn 1 khuôn mặt hay không. Nếu có 2 khuôn mặt trở lên thì xuất ra thông báo, kết thúc chương trình
 - Ngược lại thì tiếp tục chương trình, lấy tọa độ điểm đặt trung khuôn mặt
- **Bước 5:** Nhập ảnh của mẹ đứa trẻ.
- **Bước 6:**
 - Kiểm tra bức ảnh đó có nhiều hơn 1 khuôn 1 khuôn mặt hay không. Nếu có 2 khuôn mặt trở lên thì xuất ra thông báo, kết thúc chương trình
 - Ngược lại thì tiếp tục chương trình, lấy tọa độ điểm đặt trung khuôn mặt
- **Bước 7:** Phát sinh thêm điểm cho khuôn mặt `getMorePoints()` để đủ tập dữ liệu gồm 80 điểm cho ảnh bố, ảnh mẹ, ảnh con.
- **Bước 8:** Nhập hệ số alpha là tỉ lệ phần trăm biến đổi.
- **Bước 9:** Bắt đầu vòng lặp đọc file Trigulation.txt để xác định các chỉ mục các điểm trên hình và chia tam giác cho 3 ảnh bố, mẹ và ảnh con biến đổi.
 - Đọc từng chỉ mục của các đỉnh tam giác vào 3 hình để lưu biến điểm tam giác
 - Gọi hàm `morphBabyFromParents()` để gán lại điểm tam giác cho ảnh con biến đổi
 - Khi kết thúc thuật toán ta sẽ được ảnh của người con ở tỉ lệ trưởng thành
- **Bước 10:** Làm tương tự các bước trên để phát sinh ảnh của người con ở tỉ lệ đứa trẻ
- **Bước 11:** Xuất 4 ảnh ra màn hình lần lượt là: ảnh bố, ảnh mẹ, ảnh con (trưởng thành) và ảnh con (sơ sinh)

5

Bản mẫu (Prototype)

- Hình ảnh demo chương trình

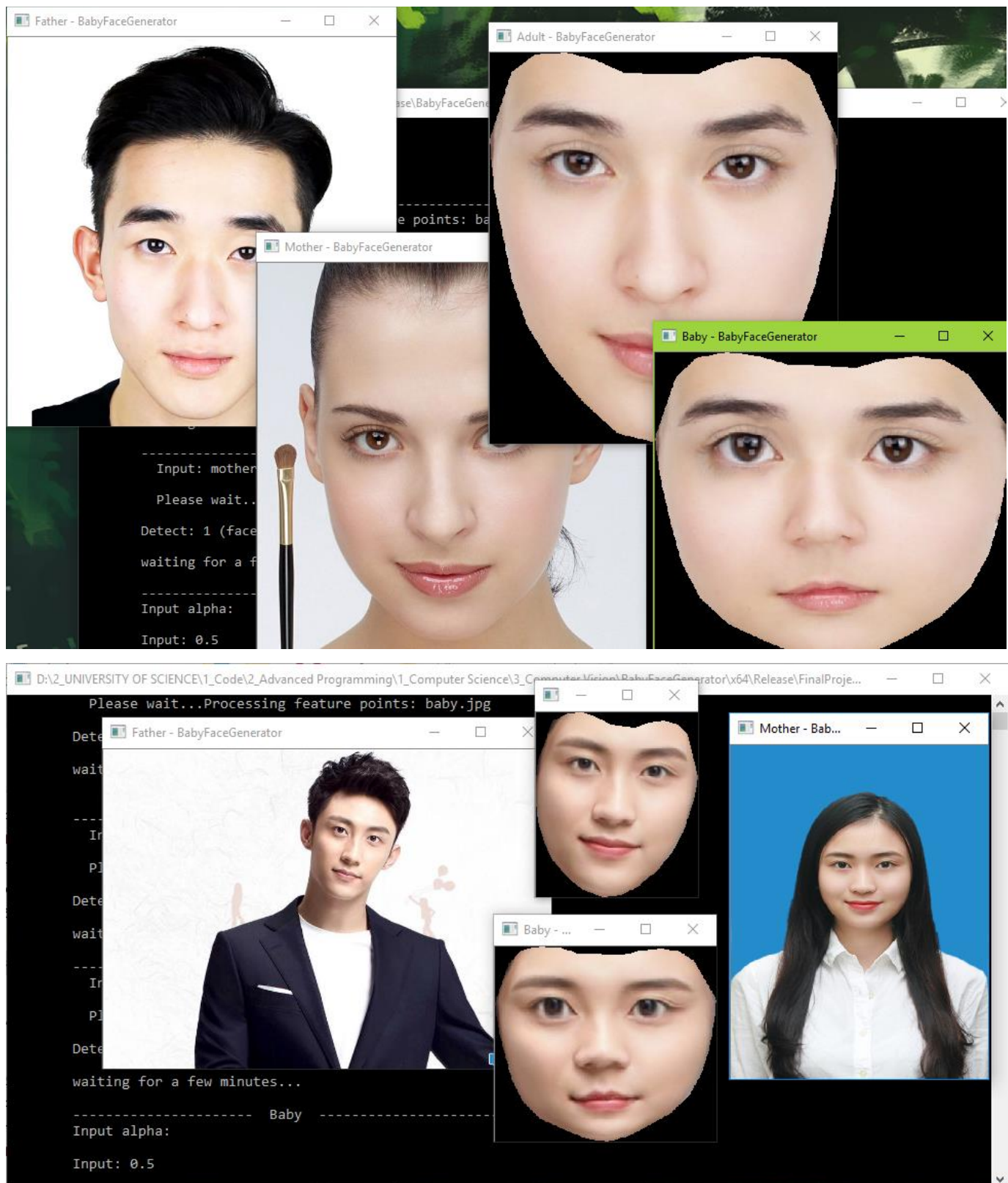
```

----- Processing -----
Please wait...Processing feature points: baby.jpg
Detect: 1 (face)
waiting for a few minutes...

----- Father -----
Input: father.jpg
Please wait...Processing feature points: father.jpg
Detect: 1 (face)
waiting for a few minutes...

----- Mother -----
Input: mother1.jpg
Please wait...Processing feature points: mother1.jpg
Detect: 1 (face)
waiting for a few minutes...

----- Baby -----
Input alpha:
Input: 0.50_
    
```



6

Tham khảo

- [1] <https://www.learnopencv.com/face-morph-using-opencv-cpp-python/>
- [2] <https://ccrma.stanford.edu/~jacobliu/368Report/>
- [3] <http://vision.gel.ulaval.ca/~jflalonde/cours/4105/h16/tps/results/tp3/JIZHA16/index.html>
- [4] <https://opencv.org/>
- [5] <http://dlib.net/>
- [6] <https://cmake.org/>