

# Báo cáo đồ án

## *Detect English Text*

### *In Photos*




#### 1. Thông tin sinh viên thực hiện

STT	MSSV	Họ Tên	Email
1	1312513	Nguyễn Nhật Tân	1312513@student.hcmus.edu.vn
2	1312681	Trần Tuấn Tú	1312681@student.hcmus.edu.vn

#### 2. Báo cáo tiến độ

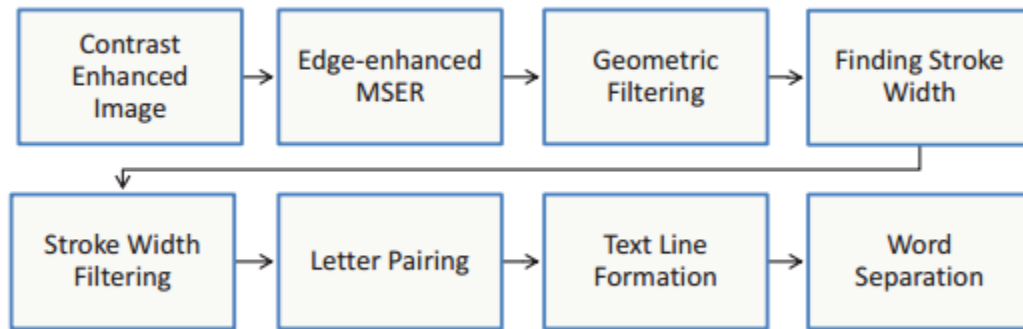
Thời gian	Kế hoạch công việc	Hoàn thành
16/10 – 23/10	Tìm hiểu và bổ sung tài liệu, các thuật toán liên quan	100%
24/10 – 31/10	Cài đặt chương trình định vị text trong ảnh	100%
1/11 – 6/11	Cài đặt chương trình đọc nội dung text đã xác định	100%
7/11 – 13/11	Cài đặt, tổng hợp và demo chương trình	100%
14/11 – 22/11	Cải tiến và khắc phục lỗi của chương trình	80%
23/11 – 30/11	Viết báo cáo và hoàn thiện chương trình	50%

#### 3. Tổ chức bài nộp

-  **Source code:** Đặt trong thư mục ComputerVisionProject
-  **Report** : Đặt trong thư mục gốc – Report-02.pdf
-  **Image** : Các ảnh dùng để test

#### 4. Báo cáo thuật toán chính dùng để cài đặt

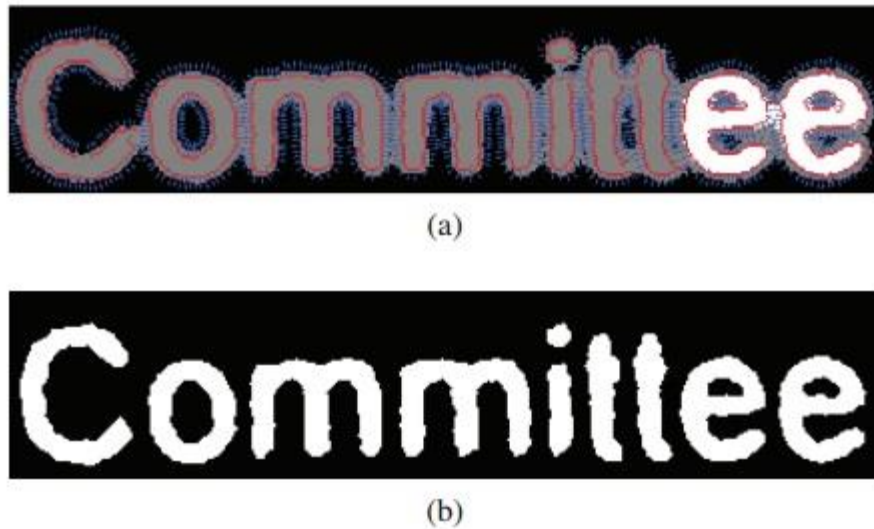
✚ Lưu đồ thuật toán áp dụng cho chương trình (Hình 4.1)



Hình 4.1

#### ✚ Edge Enhanced MSER:

- ✓ Yếu tố quan trọng để xác định vùng chữ trong ảnh bao sự tương phản giữa vùng text với nền ảnh, và sự đồng nhất về màu sắc, điểm ảnh, kích thước, tỉ lệ, sự thay đổi về độ sáng của chúng
- ✓ Phương pháp sử dụng ở đây là MSER (Maximally Stable Extremal Regions). Tuy nhiên thuật toán này lại chưa tối ưu trong việc định vị vùng text với ảnh có độ phân giải thấp. Cho nên ở đây chúng ta sử dụng MSER kết hợp với Canny Edge
- ✓ Ban đầu ta chuyển về ảnh nhị phân, sau đó dùng thuật toán phát hiện cạnh Canny Edge để xác định bao vùng cực trị (Extremal Regions)
- ✓ Tiếp theo là loại bỏ các điểm ảnh xác định bởi MSER nằm ngoài bao vùng xác định bởi Canny Edge theo hướng dọc (tức hướng từ vùng chữ màu sáng ra vùng nền màu tối)



Hình 4.2

- ✓ Phân tích kết quả (Hình 4.2):
  - (a) Thuật toán Canny edges phát hiện các cạnh (màu đỏ) ứng với khoảng bao vùng cực trị
  - (b) Hình ảnh sau khi đã loại bỏ các điểm ảnh MSER nằm ngoài vùng bao theo hướng dốc
- ✓ **Ưu điểm:** tăng hiệu quả của bộ lọc hình học (Geometric Filtering) và giúp MSER hoạt động với các ảnh mờ tốt hơn

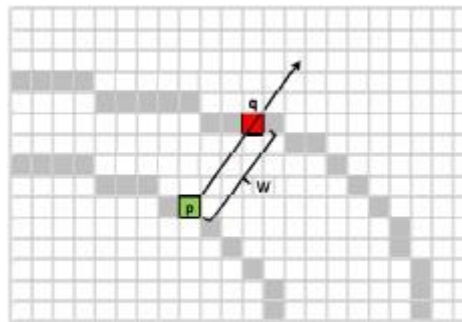
### Geometric Filtering

- ✓ Bước này ta xác định và loại bỏ những đối tượng hay vùng không phải là chữ
- ✓ Hầu hết các chữ cái đều có tỉ lệ (aspect ratio) xấp xỉ 1, ta sẽ loại bỏ tất cả các đối tượng có tỉ lệ rất lớn hoặc rất bé. Đồng thời ta cũng lựa chọn một ngưỡng tỉ lệ để đảm bảo rằng các chữ cái có dạng thon dài như "l" hay "i" sẽ không bị loại bỏ. Đồng thời loại bỏ các đối tượng có nhiều lỗ, vì đó không phải là ứng viên phù hợp.
- ✓ Aspect ratio của một chữ cái mà tỉ lệ giữa chiều dài và chiều rộng của chữ cái đó, thông thường các chữ cái thường có tỉ lệ chiều dài và chiều rộng xấp xỉ

nhau, tuy nhiên có trường hợp ngoại lệ là “l” và “i” có tỉ lệ lớn, ta sẽ chọn ngưỡng riêng cho 2 đối tượng này

#### ✚ Finding Stroke Width by Distance Transform

- ✓ Bước này ta sẽ tiến hành tìm độ dày (stroke width) của các nét chữ. Việc tìm độ dày của nét chữ là một trong những bước rất quan trọng trong việc nhận dạng chữ cái



Hình 4.3

- ✓ **Stroke width** (Hình 4.3): Độ dày của nét chữ ở điểm p chính là w ứng với khoảng cách từ p đến q
- ✓ Gán nhãn độ dày cho các điểm ảnh nằm bên trong bao vùng đã được xác định bởi thuật toán Canny edges. Bắt đầu với những điểm ảnh nằm ở chính giữa nét chữ, ta sẽ gán giá trị độ dày cho điểm ảnh bằng khoảng cách Euclide của nó đến điểm ảnh gần nhất nằm trên cạnh Canny (những điểm ảnh nằm ở rìa). Cứ như vậy tính toán lan truyền từ những điểm nằm giữa đến rìa

- ✓ Thuật toán chính để cài đặt (Công đoạn Finding Stroke Width) – trích [4]

---

**Algorithm 1** Finding stroke width
 

---

**Input:** binary image BW

**Output:** stroke width image SW

$D := \text{DistanceTransform}(BW);$

$D := \text{round}(D);$

**for**  $p = \text{each foreground pixel in } D$  **do**

$pVal := D(p);$

$\text{LookUp}(p) := p\text{'s } 8 \text{ neighbors whose value } < pVal;$

**end for**

{LookUp can be efficiently computed without FOR loop.}

$\text{MaxStroke} := \max(D);$

**for**  $\text{Stroke} = \text{MaxStroke to } 1$  **do**

$\text{StrokeIndex} := \text{find}(D == \text{Stroke});$

$\text{NeighborIndex} = \text{Lookup}(\text{StrokeIndex});$

**while**  $\text{NeighborIndex}$  not empty **do**

$D(\text{NeighborIndex}) := \text{Stroke};$

$\text{NeighborIndex} := \text{Lookup}(\text{NeighborIndex});$

**end while**

**end for**

**return**  $\text{SW} := D;$

---

Hình 4.3

### Text Line Formation and Word Separation

- ✓ Xác định định dạng dòng chữ và tách từ. Dòng văn bản là dấu hiệu quan trọng để xác định sự tồn tại của văn bản. Hầu hết các văn bản đều xuất hiện dạng đường thẳng hoặc đường cong nhẹ. Để xác định các dòng văn bản ta gom nhóm các đối tượng chữ cái thành từng cặp. Các chữ cái thuộc một dòng thường có độ dày và có cùng chiều gần tương đương nhau. Hai đối tượng chữ được ghép đôi khi và chỉ khi tỉ lệ về độ dày trung bình giữa hai chữ không vượt quá 1.5 và tỉ lệ về chiều cao giữa chúng không quá 2. Đồng thời nếu hai chữ cái cách quá xa thì ta cũng không gom nhóm chúng thành một đôi

## 5. Các chức năng của chương trình hiện tại

- Xác định được vùng text trong ảnh, tuy nhiên không phải hầu hết tất cả các ảnh. Vì khoảng cách giữa các chữ, độ rộng, cũng như khoảng cách giữa các bit ảnh duyệt theo giới hạn miền mà chương trình cung cấp cho MSER khác nhau, các thành phần liên quan cũng vậy. Hiện đang dần khắc phục
- Sử dụng OCR để đọc vùng text đã định vị được, độ chính xác tùy thuộc vào bước tạo ảnh tối ưu các nét cạnh.
  - ✓ OCR sử dụng: Tesseract của Google
  - ✓ Link: <https://code.google.com/p/tesseract-ocr/>

## 6. Danh sách tài liệu tham khảo

- [1] Lukas Neumann and Jiri Matas, A method for text localization and recognition in real-world images
- [2] Epshtein, B., Ofek, E., Wexler, Y.: Detecting text in natural scenes with stroke width transform. In: CVPR '10: Proc. of the 2010 Conference on Computer Vision and Pattern Recognition. (2010)
- [3] Chen, X., Yuille, A.L.: Detecting and reading text in natural scenes. Computer Vision and Pattern Recognition, IEEE Computer Society Conference on **2** (2004)
- [4] Chen, Huizhong, et al. "Robust Text Detection in Natural Images with Edge-Enhanced Maximally Stable Extremal Regions." Image Processing (ICIP), 2011 18th IEEE International Conference on. IEEE, 2011.
- [5] Automatically Detect and Recognize Text in Natural Images