

Note Xử lý ảnh

Ngày 14 tháng 11 năm 2022

1 Gaussian Weighted Histogram Intersection (GWHI)

$$GWHI = \sum_{\vec{c}_i \in C_M} \sum_{\vec{c}_j \in C_T} \min(h_M(\vec{c}_i), h_T(\vec{c}_j)) \cdot w(\|\vec{c}_i - \vec{c}_j\|),$$

trong đó:

$$w(\|\vec{c}_i - \vec{c}_j\|) = \begin{cases} f(\|\vec{c}_i - \vec{c}_j\|) & \|\vec{c}_i - \vec{c}_j\| \leq BW \\ 0 & \text{other} \end{cases}$$

Hàm Gaussian $f(x)$ là:

$$f(x) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(\frac{-x^2}{2\sigma^2}\right) \quad \text{với } -\infty \leq x \leq \infty$$

Ta có:

$$\int_{-3.3\sigma}^{3.3\sigma} f(x) dx = 99.9\% \int_{-\infty}^{\infty} f(x) dx$$

nên $BW = 3.3\sigma$. Với $\sigma = 0.8Th$ tức $BW = 2.64Th$, Th trong bài báo là 15 (hoặc lấy từ thuật toán MPHM)

$$\|\vec{c}_i - \vec{c}_j\| = \sqrt{(l_1 - l_2)^2 + (u_1 - u_2)^2 + (v_1 - v_2)^2} \quad (1)$$

Theo công thức của OpenCV:

- Công thức chuyển RGB thành XYZ:

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} 0.412453 & 0.357580 & 0.180423 \\ 0.212671 & 0.715160 & 0.072169 \\ 0.019334 & 0.119193 & 0.950227 \end{bmatrix} \cdot \begin{bmatrix} R \\ G \\ B \end{bmatrix}$$

Công thức chuyển đổi XYZ thành CIE Luv:

$$L^* = \begin{cases} \left(\frac{29}{3}\right)^3 Y & \frac{Y}{Y_n} \leq \left(\frac{6}{29}\right)^3 \\ 116Y^{1/3} - 16 & Y > \left(\frac{6}{29}\right)^3 \end{cases}$$

$$u^* = 13L^* \cdot (u' - u'_n)$$

$$v^* = 13L^* \cdot (v' - u'_n)$$

trong đó:

$$u' = \frac{4X}{X + 15Y + 3Z}$$

$$v' = \frac{9Y}{X + 15Y + 3Z}$$

$$u_n = 0.19793943$$

$$v_n = 0.46831096$$

Khi đó $0 \leq L \leq 100, -134 \leq u \leq 220, -140 \leq v \leq 122$.

2 Colour Ratio Gradients

Gradient của tỉ lệ màu không đổi được tính theo công thức:

$$\begin{cases} \nabla M_{RG} = \sqrt{\left(\frac{R_1 G_2 - R_2 G_1}{R_2 G_1 + R_1 G_2}\right)^2 + \left(\frac{R_3 G_4 - R_4 G_3}{R_4 G_3 + R_3 G_4}\right)^2} \\ \nabla M_{RB} = \sqrt{\left(\frac{R_1 B_2 - R_2 B_1}{R_2 B_1 + R_1 B_2}\right)^2 + \left(\frac{R_3 B_4 - R_4 B_3}{R_4 B_3 + R_3 B_4}\right)^2} \\ \nabla M_{GB} = \sqrt{\left(\frac{G_1 B_2 - G_2 B_1}{G_2 B_1 + G_1 B_2}\right)^2 + \left(\frac{G_3 B_4 - G_4 B_3}{G_4 B_3 + G_3 B_4}\right)^2} \end{cases}$$

trong đó $p = 1, 2$ là hai ô trái phải tức $(x-1, y)$ và $(x+1, y)$ của (x, y) , $p = 3, 4$ là hai ô trên dưới tức $(x, y-1)$ và $(x, y+1)$ của (x, y) .

Thực hiện đếm số lần một gradient của tỉ lệ màu xuất hiện trong ảnh, thu được histogram của ảnh truy vấn Q là \mathcal{H}^Q và ảnh cần xét I_t là \mathcal{H}^{I_t} . Ta có \mathcal{D}_a và \mathcal{D}_x là histogram intersection và histogram matching định nghĩa bởi chuẩn hóa tương quan chéo (normalized cross correlation) là:

$$\mathcal{D}_a(\mathcal{H}^Q, \mathcal{H}^{I_t}) = \frac{\sum_{\vec{k}=1}^N \min\left\{\mathcal{H}^{I_t}(\vec{k}), \mathcal{H}^Q(\vec{k})\right\}}{\sum_{\vec{k}=1}^N \mathcal{H}^Q(\vec{k})}$$

$$\mathcal{D}_x(\mathcal{H}^Q, \mathcal{H}^{I_t}) = \frac{\sum_{\vec{k}=1}^N \mathcal{H}^{I_t}(\vec{k}) \mathcal{H}^Q(\vec{k})}{\sum_{\vec{k}=1}^N \left(\mathcal{H}^{I_t}(\vec{k})\right)^2}$$

trong đó \vec{k} là bậc màu và N là số bậc màu.

Ta chia thuật toán thành hai bước tách ảnh thành các vùng và thu nhận ảnh nhằm mục đích giảm số lượng ảnh khớp sai.

2.1 Chia ảnh thành các vùng đặc trưng (Segmentation into Texture Regions)

- Khởi tạo: Ảnh đầu vào là ảnh truy vấn Q và ảnh cần xét I_k . Từ gradient tỉ lệ màu, ta thu được histogram tương ứng $\mathcal{H}^Q, \mathcal{H}^{I_k}$.
- Phân tách: Ta thực hiện tách một góc phần tư thành 4 góc phần tư nhỏ khi nó không đủ độ tương thích được tính theo \mathcal{D}_a . Nếu \mathcal{D}_a dưới ngưỡng cho trước, ta coi góc phần tư có kết cấu tương ứng ảnh truy vấn, ngược lại, ta tiếp tục chia nhỏ. Việc chia nhỏ tiếp tục đến khi có kết cấu phù hợp hoặc đạt cỡ pixel nhỏ nhất, ở đây ta đặt là 8×8 .
- Gộp lại: Thực hiện nối các vùng lân cận lại. Ta cho phép các vùng lân cận được gộp vào khi \mathcal{D}_x của chúng với ảnh truy vấn đạt dưới ngưỡng nhất định. Ta bắt đầu từ khối có độ tương đồng cao nhất với ảnh truy vấn và xem xét các khối liền kề, việc xem xét dừng lại khi mọi khối lân cận gộp vào đều cho \mathcal{D}_x vượt ngưỡng.
- Mở rộng vùng: Do có rất nhiều vùng nhỏ ở biên vùng liên kết, cần có phương pháp cho phép các vùng đủ lớn có thể thực hiện mở rộng. Thuật toán chọn những vùng có cỡ đủ lớn, sau đó cho vùng gộp với vùng lân cận phù hợp nhất, đồng thời kết quả gộp có \mathcal{D}_x thỏa mãn độ đo kết cấu cho trước. Quá trình gộp dừng lại khi ko còn vùng lớn nào cần gộp.
- Output: Một ảnh gán nhãn xây bằng cách gán nhãn từng pixel dựa vào vùng của pixel. Mỗi vùng có một nhãn j , $1 \leq j \leq$ số vùng cuối.

2.2 Thu thập ảnh (Color Constant Texture Retrieval)

Để tiết kiệm thời gian, ta thực hiện theo các bước:

- Bước 1: Lọc sơ các ảnh bằng histogram intersection, chọn ra những ảnh có điểm cao nhất.
- Bước 2: áp dụng thuật toán phân đoạn ảnh (segmentation) ở trên cho các ảnh lọc ra từ bước 1.
- Bước 3: Với mỗi ảnh cần xét, ta chọn vùng có độ tương quan \mathcal{D}_a cao nhất với ảnh gốc để so sánh, sau đó sắp xếp chúng theo \mathcal{D}_x .

3 Colour Edge Cooccurrence Histograms (CECH)

3.1 Cách tính CECH histogram

Cho ảnh F , ta thực hiện tính CECH histogram của ảnh theo các bước sau:

- Bước 1: Detecting edges

- Để thu được tập cạnh E của ảnh $F = f(x, y)$, ta dùng các điểm cạnh giả có cường độ gradient mạnh. Gradient của ảnh $f(x, y)$ tại điểm (x, y) được định nghĩa bằng:

$$\nabla f = \begin{bmatrix} G_x \\ G_y \end{bmatrix} = \begin{bmatrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \end{bmatrix}$$

độ lớn và hướng của vecto này được, kí hiệu bởi $\nabla f(x, y)$ và $\theta(x, y)$ là:

$$\nabla f(x, y) = \sqrt{G_x^2 + G_y^2}$$

và

$$\theta(x, y) = \tan^{-1} \left(\frac{G_y}{G_x} \right)$$

- Hướng của cạnh tại vị trí (x, y) vuông góc với hướng của vecto gradient. Tính gradient của ảnh dựa vào đạo hàm riêng $\frac{\partial f}{\partial x}$ và $\frac{\partial f}{\partial y}$ tại mọi pixel. Trong bài này, toán tử Sobel được sử dụng để lấy đạo hàm cấp 1 tại mỗi điểm.
- Các pixel có cường độ mạnh hơn ngưỡng $TH_G = 30$ cho trước được coi là cạnh và được xem xét. Do đó các tập cạnh có thể mô tả bằng:

$$E(x, y) = \begin{cases} 1 & \nabla f(x, y) \geq TH_G \\ 0 & \text{còn lại} \end{cases}$$

- Bước 2: Định vị cặp pixel

- Với điểm cạnh p tại (x, y) , ta lấy hai điểm p_1, p_2 thuộc hai phía của hướng gradient của p và cách p một khoảng d nhất định. Khi đó tọa độ (x_1, y_1) và (x_2, y_2) được tính bằng

$$\begin{cases} x_1 = x - d \times \cos \theta \\ y_1 = y + d \times \sin \theta \end{cases}$$

và

$$\begin{cases} x_2 = x + d \times \cos \theta \\ y_2 = y - d \times \sin \theta \end{cases}$$

- Bước 3: Cộng lại

- Kí hiệu giá trị màu ở điểm p_1, p_2 là \vec{c}_1, \vec{c}_2 . Tập màu tương ứng của CECH được tính bằng:

$$CECH(\vec{c}_1, \vec{c}_2, d) = \text{size}(\{(p_1, p_2) | p_1, p_2 \in F \text{ và } \|p_1 - p\| = \|p_2 - p\| = d\})$$

trong đó $\vec{c}_1, \vec{c}_2 \in C$ là tập màu của ảnh.

- Sau khi quét mọi điểm cạnh trên ảnh, ta tổng hợp được các cặp màu giống nhau trong ảnh và tổng hợp chúng vào các bậc màu của biểu đồ CECH.

- Bước 4: Chuẩn hóa CECH histogram với tổng số điểm cạnh của ảnh.

3.2 Giảm số màu.

Giả sử số màu trong ảnh là N_C . Khi đó số cặp màu pixel có thể xảy ra là $N_C \times N_C$, đây cũng là chiều của biểu đồ CECH khi d cố định. Không mất tính tổng quát, giả sử hai ảnh cần xét có cùng số màu. Nếu tập màu hai ảnh giống hệt nhau, ta chỉ cần thực hiện so sánh từng bậc màu tương ứng. Do đó, số phép toán tối thiểu khi so sánh hai CECH histogram là N_C^2 . Tuy nhiên, nếu tập màu của hai ảnh khác nhau, với mỗi bậc màu trong CECH này, cần so sánh với mọi bậc màu trong CECH còn lại. Do đó, số phép toán là N_C^4 để so sánh tìm sự giống nhau giữa hai biểu đồ CECH.

Với một ảnh 24 bit màu, ta có $N_C = 2^{24}$, rất khó để có thể thực hiện so sánh trong thời gian thực, do vậy ta cần một phương pháp để giảm số màu.

Theo phương pháp của Xiaolin Wu, với một tập màu $\Omega(c_i, c_j]$ ($c_i = (c_{ir}, c_{ig}, c_{ib})$), ta cần chọn c sao cho biểu thức sau nhỏ nhất:

$$E(c_i, c] + E(c, c_j)$$

trong đó

$$c \in \{c_{jr} \times c_{jg} \times (c_{ib}, c_{jb}]\} \cup \{c_{jr} \times (c_{ig}, c_{jg}] \times c_{jb}\} \cup \{(c_{ir}, c_{jr}] \times c_{jg} \times c_{jb}\} \quad (2)$$

Các công thức, ký hiệu được định nghĩa như sau:

- Một màu c thuộc tập $\Omega(c_i, c_j]$ khi ba giá trị c_r, c_g, c_b lần lượt thuộc tương ứng $(c_{ir}, c_{jr}]$, $(c_{ig}, c_{jg}]$, $(c_{ib}, c_{jb}]$.
- $\omega(c_i, c_j] = \sum_{c \in \Omega(c_i, c_j]} P(c)$ với $P(c)$ là tỉ lệ màu c trong ảnh cần xét.
- $E(c_i, c_j] = \sum_{c \in \Omega(c_i, c_j]} c^2 P(c) - \frac{[\sum_{c \in \Omega(c_i, c_j]} c P(c)]^2}{\omega(c_i, c_j]}$

Để rút gọn số phép tính cho công thức, ta áp dụng:

$$\sum_{c \in \Omega(c_i, c_j]} f(c) P(c) = \left[\sum_{c \in \Omega(0, v_7]} - \sum_{c \in \Omega(0, v_6]} - \sum_{c \in \Omega(0, v_5]} + \sum_{c \in \Omega(0, v_4]} \right. \\ \left. - \sum_{c \in \Omega(0, v_3]} + \sum_{c \in \Omega(0, v_2]} + \sum_{c \in \Omega(0, v_1]} - \sum_{c \in \Omega(0, v_0]} \right] f(c) P(c) \quad (3)$$

trong đó v_i là tám đỉnh của $\Omega(c_i, c_j]$

$$\begin{aligned} v_0 &= (c_{ir}, c_{ig}, c_{ib}), v_4 = (c_{jr}, c_{ig}, c_{ib}) \\ v_1 &= (c_{ir}, c_{ig}, c_{jb}), v_5 = (c_{jr}, c_{ig}, c_{jb}) \\ v_2 &= (c_{ir}, c_{jg}, c_{ib}), v_6 = (c_{jr}, c_{jg}, c_{ib}) \\ v_3 &= (c_{ir}, c_{jg}, c_{jb}), v_7 = (c_{jr}, c_{jg}, c_{jb}) \end{aligned}$$

, vecto 0 là điểm thỏa mãn $\sum_{c \in \Omega(-\infty, 0]} P(c) = 0$.

Thay $f(c)$ bằng c^d , với $c^0 = 1, c^2 = cc^T$, ta thu được:

$$M_d(c_t) = \sum_{c \in \Omega(0, c_t]} c^d P(c)$$

Ta thấy 4 trên 8 $M_d(c)$ thuộc công thức (3) có giá trị không đổi với mọi c thuộc tập 2, từ đó giảm số lần cần tính lại các giá trị biểu thức dạng $\sum_{c \in \Omega(c_i, c_j]} c^d P(c)$.

Như vậy, với mỗi $\Omega(c_i, c_j]$, cần tìm giá trị nhỏ nhất của biểu thức sau:

$$\begin{aligned} E(c_i, c] + E(c, c_j] &= \sum_{c \in \Omega(c_i, c_j]} c^2 P(c) \\ &\quad - \frac{\left[\sum_{c \in \Omega(c_i, c]} c P(c) \right]^2}{\omega(c_i, c]} - \frac{\left[\sum_{c \in \Omega(c, c_j]} c P(c) \right]^2}{\omega(c, c_j]}, \end{aligned}$$

tương ứng với việc tìm giá trị lớn nhất của:

$$\begin{aligned} &\frac{\left[\sum_{c \in \Omega(c_i, c]} c P(c) \right]^2}{\omega(c_i, c]} + \frac{\left[\sum_{c \in \Omega(c, c_j]} c P(c) \right]^2}{\omega(c, c_j]} \\ &= \frac{\left[\sum_{c \in \Omega(c_i, c]} c P(c) \right]^2}{\omega(c_i, c]} + \frac{\left[\sum_{c \in \Omega(c_i, c_j]} c P(c) - \sum_{c \in \Omega(c_i, c]} c P(c) \right]^2}{\omega(c_i, c_j] - \omega(c_i, c]}. \end{aligned}$$

3.3 So sánh biểu đồ CECH bằng GWHI mở rộng.

Để so sánh hai biểu đồ CECH có tập màu tương tự nhau, khởi tạo η bằng 0. Với mỗi $\text{CECH}_M(\vec{c}_i, \vec{c}_j, d)_{|\vec{c}_i, \vec{c}_j \in C_M}$ trong biểu đồ CECH của ảnh cần xét, quét từng $\text{CECH}_T(\vec{c}_k, \vec{c}_l, d)_{|\vec{c}_k, \vec{c}_l \in c_t}$ trong biểu đồ CECH của ảnh gốc khi thực hiện các bước (1) đến (4) đến khi tất cả các cặp màu trong CECH của ảnh cần xét đã được quét:

1. Tính khoảng cách $d_1 = \|\vec{c}_i - \vec{c}_k\|, d_2 = \|\vec{c}_j - \vec{c}_l\|$ theo công thức (1), chọn $d_C = \max(d_1, d_2)$.
2. Nếu $d_C \leq BW = 52.8??$, phần giao nhau giữa hai biểu đồ CECH được định nghĩa:

$$\text{its} = \min(\text{CECH}_M(\vec{c}_i, \vec{c}_j, d), \text{CECH}_T(\vec{c}_k, \vec{c}_l, d))$$

3. Áp dụng hàm Gaussian của khoảng cách vào its ta có trọng số:

$$\text{its}' = \text{its} \times \exp\left(-\frac{d_C^2}{2\sigma^2}\right) \quad \text{với} \quad -BW \leq d_C \leq BW,$$

trong đó σ và BW được nêu lên trong phần trước.

4. Tổng các trọng số ta được: $\eta \leftarrow \eta + \text{its}'$.

η cuối cùng là tỉ lệ đối sánh giữa hai biểu đồ CECH thuộc khoảng $[0, 1]$. η càng lớn, hai ảnh càng giống nhau.