

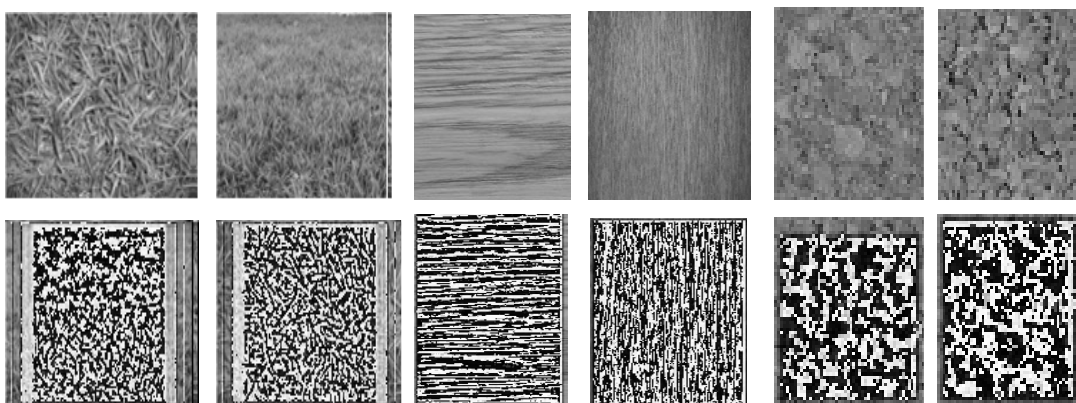
MỘT MÔ HÌNH NHIỄU VÀ ỨNG DỤNG TRONG VIỆC PHÁT HIỆN CHẤT LIỆU

ĐỖ NĂNG TOÀN, LÊ THỊ KIM NGA, NGUYỄN THỊ HỒNG MINH

1. ĐẶT VẤN ĐỀ

Trong cuộc sống hằng ngày, chúng ta luôn phải tiếp xúc các chất liệu hoặc các đối tượng nào đó. Việc phát hiện các loại chất liệu hay đối tượng trong bức ảnh một cách tự động là điều có thể và cần thiết, nó nhằm mục đích xây dựng các hệ thống hỗ trợ phục vụ con người như các hệ thống giám sát tự động, hệ thống phân tích hình ảnh trên vũ trụ,...Như vậy, vấn đề đặt ra là tìm xem trong một bức ảnh cho trước có các loại chất liệu nào đó không, hoặc tìm cho tôi các bức ảnh có nước, có người chẳng hạn..., những vấn đề đó đều quy về bài toán phát hiện chất liệu.

Một thực tế cho thấy rằng, ảnh sau quá trình thu nhận thường không tránh khỏi nhiễu do môi trường, bản thân đối tượng và các thiết bị quang học, điện tử [8, 9]. Về mặt vật lí, mỗi chất liệu sẽ có các thuộc tính khác nhau và do vậy lượng nhiễu dư gây bởi chất liệu cũng sẽ khác nhau (hình 1).



Hình 1. Ảnh chất liệu và nhiễu

(dòng trên: các ảnh chất liệu; dòng bên dưới: các ảnh nhiễu tương ứng)

Xuất phát từ thực tế đó, trong bài báo này chúng tôi đề xuất một mô hình nhiễu cho chất liệu dựa vào lượng nhiễu dư trong quá trình thu nhận ảnh. Trên cơ sở mô hình nhiễu chất liệu đã xây dựng, chúng tôi đưa ra một thuật toán phát hiện chất liệu dựa vào nhiễu. Kết quả này là sự kế thừa kết quả của một nghiên cứu trước [4] của chính nhóm tác giả. Đó là, dựa vào đặc điểm, mỗi hãng chế tạo thiết bị thu nhận thường có các chiến lược thu nhận ảnh khác nhau nên lượng nhiễu dư do các camera khác nhau để lại trên ảnh là khác nhau. Nhờ vào việc xác định lượng nhiễu dư này, nhóm tác giả đề xuất kĩ thuật nhận dạng ra loại camera đã dùng để chụp bức ảnh

và do đó, có thể phát hiện được ảnh giả được ghép bởi các bức ảnh được chụp bởi các camera khác nhau.

Phần còn lại của bài báo được cấu trúc như sau: Phần 2 sẽ trình bày về nhiễu và mô hình nhiễu cho chất liệu. Phần 3, chúng tôi đưa ra thuật toán phát hiện chất liệu dựa vào nhiễu. Phần bốn sẽ là kết quả thử nghiệm thuật toán. Cuối cùng là kết luận về kỹ thuật đề xuất.

2. NHIỄU VÀ MÔ HÌNH NHIỄU CHẤT LIỆU

2.1. Các nghiên cứu liên quan

Ảnh sau quá trình thu nhận thường không tránh khỏi nhiễu, nhiễu gây ra bởi ánh sáng tác động lên chính bản thân chất liệu và trong đó một phần nhỏ do thiết bị thu nhận ảnh đã tạo nên các đặc trưng riêng cho mỗi chất liệu. Có thể nói, nhiễu được xem như thành phần không mong muốn có trong ảnh, nhiễu là một hiện tượng ngẫu nhiên luôn luôn có mặt trên mọi hệ thống xử lý tín hiệu thực. Nhiễu xuất hiện trong ảnh bởi nhiều nguyên nhân như do sự thay đổi độ nhạy của đầu dò, do sự biến đổi của môi trường, do chính bản thân chất liệu sinh ra, do sai số lượng tử hóa hay sai số truyền,... Tất cả các nguyên nhân gây ra nhiễu ở trên đã sinh ra nhiễu được phân thành các loại chính như sau:

- **Nhiễu độc lập với dữ liệu ảnh (Independent Noise):** Là một loại nhiễu cộng (additive noise): ảnh thu được $f(i,j)$ là tổng của ảnh đúng (true image) và nhiễu $n(i,j)$: $f(i,j) = s(i,j) + n(i,j)$.

- **Nhiễu phụ thuộc vào dữ liệu (Data dependent noise):** Nhiễu xuất hiện khi có sự bức xạ đơn sắc nằm rải rác trên bề mặt ảnh, độ lớn chớm trên bề mặt tùy thuộc vào bước sóng của điểm ảnh. Do có sự giao thoa giữa các sóng ảnh nên làm xuất hiện những vết lốm đốm trên ảnh.

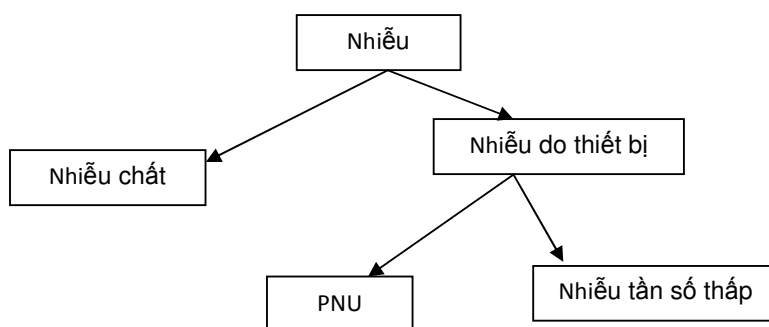
- **Nhiễu Gauss:** Nhiễu này có được do bản chất rời rạc của bức xạ (hệ thống ghi ảnh bằng cách đếm các photon (lượng tử ánh sáng)). Là nhiễu cộng và độc lập (independent, additive noise) – nhiễu $n(i,j)$ có phân phối Gauss (trung bình = zero) được mô tả bởi độ lệch chuẩn (standard deviation), hay phương sai. Mỗi pixel trong ảnh nhiễu là tổng giá trị pixel đúng (true pixel) và pixel ngẫu nhiên

- **Nhiễu muối – tiêu (Salt & Pepper noise):** Nhiễu này sinh ra do xảy ra sai số trong quá trình truyền dữ liệu. Những pixel đơn được đặt luân phiên mang giá trị zero hay giá trị maximum tạo ra hình chấm dạn muối tiêu trên ảnh.

Thông thường trong các bài toán xử lý với ảnh số, người ta thường phải khử bỏ nhiễu trước khi sử dụng bức ảnh trong những ứng dụng nào đó, đặc biệt là trong các ứng dụng chuyên dụng của các lĩnh vực phân tích và xử lý hình ảnh. Gần đây, đã có một số ứng dụng của việc phân tích lượng nhiễu dư trong các bức ảnh để giải quyết bài toán phát hiện ảnh giả mạo dạng cắt dán, như nhóm Fridrich [7], nhóm đã xây dựng được nhiễu bất biến cho camera, với mỗi camera sẽ có một lượng nhiễu đặc trưng cho nó. Với một bức ảnh giả mạo dạng cắt dán cho trước, nhóm nghiên cứu này đã tính được ảnh nhiễu trong bức ảnh đó. Ảnh không phải là giả mạo nếu bức ảnh nhiễu của nó không tương thích tại tất cả các vùng. Sau đó nhóm chúng tôi, cũng dựa trên cơ sở phân tích nhiễu và đưa ra một mô hình cho nhiễu cảm biến, và được ứng dụng khá thành công cho đề tài phát hiện ảnh giả mạo của Viện Công Nghệ Thông Tin [4]. Dựa trên tư tưởng đó chúng tôi đã nghiên cứu và đề xuất một mô hình cho nhiễu chất liệu, từ mô hình này sẽ xây dựng các đặc trưng nhiễu bất biến cho chất liệu. Chúng tôi cũng sử dụng các đánh giá thông kê để tính toán đặc trưng nhiễu bất biến chất liệu, các đánh giá thông kê là phù hợp nhất cho việc mô tả và phát hiện chất liệu, đặc biệt là chất liệu tự nhiên [1, 2, 8, 9].

2.2. Mô hình nhiễu chất liệu

Có một vài nguồn không hoàn thiện và nhiễu đã có tác động đến bức ảnh trong quá trình xử lí. Khi camera thu nhận một bức ảnh tuyệt đối từ môi trường, kết quả của ảnh vẫn có sự thay đổi nhỏ trong cường độ của các điểm ảnh riêng. Một phần nguyên nhân do các thành phần nhiễu ngẫu nhiên gây ra, cũng như “*readout noise*” hoặc “*shot noise*” và một phần bởi do chính nhiễu chất liệu hay nhiễu đối tượng, chúng là thành phần xác định tồn tại dưới dạng xấp xỉ được sinh ra trong quá trình thu nhận thông qua bộ cảm biến. Có hai thành phần chính của nhiễu là nhiễu do bản thân đối tượng chất liệu sinh ra và nhiễu do thiết bị điện tử. Với nhiễu do chất liệu sinh ra thực chất bị ảnh hưởng bởi ánh sáng tác động vào nó, ánh sáng này tác động lên mỗi chất liệu khác nhau sẽ sinh ra một lượng nhiễu khác nhau. Mặt khác, nhiễu do thiết bị bao gồm: nhiễu cố định (mẫu nhiễu có cường độ biến đổi trong miền biên độ ảnh) và nhiễu hỗn tạp không đều, điểm ảnh bất định (PNU: *pixel non-uniformity*) dựa trên sự khác nhau giữa pixel-pixel khi màn cảm biến không đặt vào nguồn sáng. Với nhiễu do thiết bị chúng ta có thể xấp xỉ nó thành một đặc trưng bất định. Do đó sự khác nhau giữa chất liệu này với chất liệu khác có thể được xem xét thông qua nhiễu chất liệu. Xét tín hiệu thô $x = (x_{ij})$ với $i = 1, \dots, m, j = 1, \dots, n$ với n, m là các kích thước của ảnh.



Hình 2. Mô hình nhiễu

Xét một tín hiệu nhiễu ngẫu nhiên: $\eta = (\eta_{ij})$, nhiễu được thêm vào do tác động bên ngoài là $\varepsilon = (\varepsilon_{ij})$, đốm đen hiện thời là $c = (c_{ij})$. Tín hiệu đầu ra được tính bởi $y = (y_{ij})$ theo công thức sau:

$$y_{ij} = f_{ij}(x_{ij} + n_{ij} + \eta_{ij}) + c_{ij} + \varepsilon_{ij} \quad (1)$$

Đối với chất liệu thì nhiễu chất liệu n_{ij} là thành phần chủ yếu có trong nhiễu và nó thể hiện được sự phân biệt giữa các chất liệu khác nhau. Các thành phần còn lại là không đáng kể và có thể xem như một thành phần bất định.

Tín hiệu y thu được phải qua một chuỗi các xử lí phức tạp trước khi file ảnh cuối cùng được lưu trữ. Quá trình xử lí bao gồm các thao tác trên một láng giềng cục bộ của các điểm ảnh, cũng như demosaicking, hiệu chỉnh màu, hoặc bộ lọc kernel. Một vài toán tử không tuyến tính, cũng như hiệu chỉnh Gamma, tính toán số dư trắng, hoặc áp dụng nội suy màu. Điểm ảnh cuối cùng có giá trị là P_{ij} , với giả sử rằng $0 \leq P_{ij} \leq 255$ cho mỗi kênh màu là

$$P_{ij} = T(y_{ij}, N(y_{ij}), i, j) \quad (2)$$

ở đây T là một hàm không tuyến tính của y_{ij} , vị trí điểm ảnh (i, j), và giá trị y từ một láng giềng cục bộ $N(y_{ij})$.

3. PHÁT HIỆN CHẤT LIỆU DỰA VÀO NHIỄU

Giải pháp chung nhất cho bài toán phát hiện chất liệu là trước hết chúng ta phải xây dựng mô tả cho chất liệu hay nói khác hơn là tìm ra những thuộc tính đặc tả cho mỗi chất liệu, sau đó dò tìm trong bức ảnh cho trước có vùng đặc trưng của chất liệu đó hay không trên cơ sở đối sánh đặc trưng [3, 5, 6]. Chúng ta cũng chú ý rằng, các đặc trưng mô tả các đặc tính chất liệu đều được trích rút và xây dựng từ ảnh.

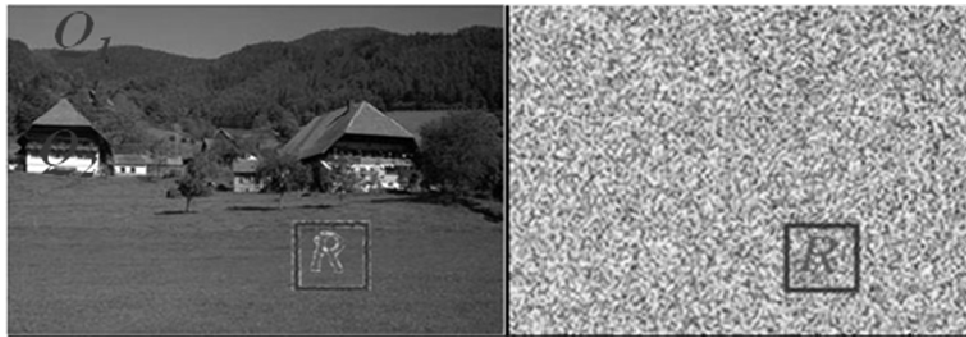
3.1. Dò tìm đặc trưng nhiễu

Các vùng chất liệu khác nhau sẽ có lượng nhiễu khác nhau. Phát hiện chất liệu bằng cách tìm kiếm vùng nhiễu dư tương quan với nhiễu chất liệu đã được xây dựng trước. Do đó phát hiện chất liệu phải bắt đầu bằng việc xác định nhiễu chất liệu tham chiếu. Quá trình xây dựng nhiễu chất liệu như sau:

Về nguyên tắc, với một chất liệu, ta tìm nhiễu chất liệu P_M bằng cách lấy trung bình của nhiều phần tử ảnh (pixels) $p^{(k)}$, $k = 1, \dots, N_p$. Để tốc độ tính toán nhanh hơn cần thực hiện các phép khử nhiễu từ bức ảnh trước khi lấy trung bình và thu được nhờ sử dụng bộ lọc nhiễu F và lấy trung bình lượng nhiễu $n^{(k)}$:

$$n^{(k)} = p^{(k)} - F(p^{(k)}) \quad (3)$$

Chúng tôi tiến hành thực nghiệm với một số bộ lọc khử nhiễu F và quyết định chọn phương pháp lọc nhiễu wavelet, có kết quả lọc tốt nhất.



a. Ảnh được thu nhận

b. Ảnh Mẫu nhiễu

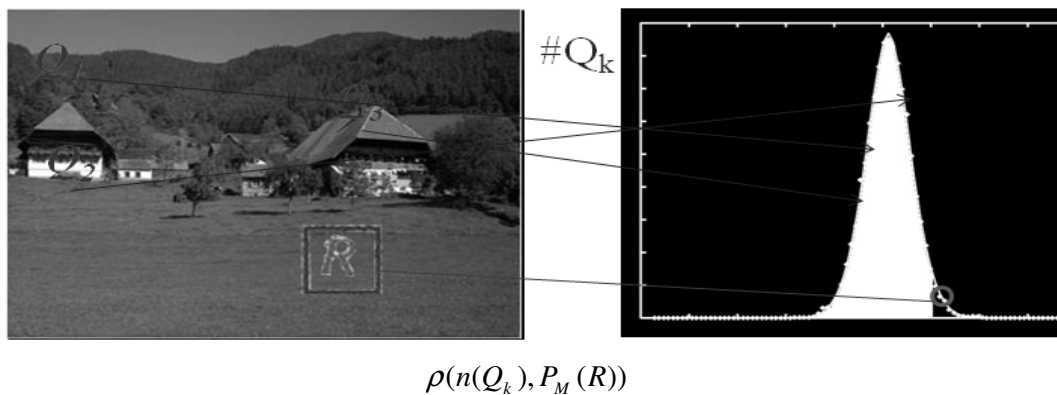
Hình 3. Minh họa vùng R được chọn và mẫu nhiễu tham chiếu tương ứng

Sau đó, với mỗi vùng của một bức ảnh cho trước chúng tôi tìm xem nhiễu chất liệu của vùng này có tương tự nhiễu chất liệu tham chiếu hay không, nếu tương tự nhau thì ta sẽ kết luận trong bức ảnh đó có chất liệu M. Giả sử vùng cần xét là một vùng R trong bức ảnh p nào đó có phải là chất liệu tương thích với mẫu nhiễu tham chiếu đã có hay không, đầu tiên chúng ta tính toán độ tương quan giữa số dư nhiễu $n(R) = p(R) - F(p(R))$ với mẫu tham chiếu P_M :

$$\rho(n(R), P_M(R)) = \frac{(n(R) - E[n](R))(P_M(R) - E[P_M](R))}{\|n(R) - E[n](R)\| \|P_M(R) - E[P_M](R)\|} \quad (4)$$

với $n(\mathfrak{R})$ và $P_M(\mathfrak{R})$ biểu thị n và P_M trong vùng R và được viết dưới dạng vector. Dấu gạch ngang trên đầu kí tự biểu thị giá trị trung bình. “.” sử dụng để biểu thị dấu nhân, và $\|\cdot\|$ kí hiệu chuẩn L_2 .

Nhiều sinh ra do thiết bị thu nhận cùng với nhiễu chất liệu tạo nên một đặc trưng nhiễu bất biến cho mỗi chất liệu khác nhau. Nếu mẫu nhiễu của hai chất liệu có độ tương quan lớn thì cũng có nghĩa rằng hai chất liệu đó tương tự nhau. Hay nói khác hơn nhiễu của chất liệu cần dò tìm phải nằm trong lân cận nhiễu tham chiếu chất liệu cho trước nếu chất liệu đó tương tự với chất liệu tham chiếu. Chúng ta có thể thử đánh giá độ tương quan $\rho(n(Q), P_M(Q))$ trong vùng Q giống như vùng \mathfrak{R} của ảnh. Sau đó để thu được kết quả về tính chính xác của \mathfrak{R} ta dựa trên phép thống kê các giá trị $\rho(n(\mathfrak{R}), P_M(\mathfrak{R}))$.



Hình 4. Hình ảnh minh họa chọn các vùng Q_k và mẫu tham chiếu \mathfrak{R}

Nhưng vấn đề quan trọng ở đây là làm sao xác định độ chính xác thông qua lân cận. Để giải quyết khó khăn này chúng tôi sử dụng cách đánh giá bằng phân phối Gaussian. Sự thuận lợi ở đây là chúng ta có thể luôn luôn thu được số các danh sách mẫu thử lớn cần thiết để đánh giá các tham số $\rho(n(\mathfrak{R}), P_M(\mathfrak{R}))$. Tiếp sau đây chúng tôi sẽ miêu tả cách thức xử lí với bài toán này.

Đầu tiên tính toán độ tương quan $\rho(n(Q_k), P_M(\mathfrak{R}))$ cho các vùng $Q_k, k = 1, \dots, N_{\mathfrak{R}}$ với kích thước như nhau. Chúng ta tính toán các $\rho(n(Q_k), P_M(\mathfrak{R})), k = 1, \dots, N_{\mathfrak{R}}$, quá trình ra quyết định được xác định thông qua phân phối Gaussian (generalized Gaussian distribution) với hàm tích lũy $G(x)$. Thông qua mô hình phân phối Gaussian, chúng ta sẽ xác định được p hợp lí để lựa chọn kết quả phát hiện chính xác. Áp dụng mô hình này có thể tổng quát hoá một phân phối Gaussian biến đổi ngẫu nhiên với ước lượng phân loại sẽ thu được giá trị $\rho(n(\mathfrak{R}), P_M(\mathfrak{R}))$ hoặc khác hơn là:

$$p = 1 - G(\rho(n(\mathfrak{R}), P_M(\mathfrak{R}))) \quad (5)$$

thực nghiệm cho chúng tôi kết quả nếu $p > \alpha = 0,01$ thì vùng R chính là vùng chất liệu cần tìm.

3.2. Thuật toán phát hiện chất liệu dựa vào nhiễu

Dựa vào đặc trưng nhiễu theo mô hình được đề xuất ở trên, mục này chúng tôi đề xuất thuật toán phát hiện chất liệu dựa vào nhiễu theo hai giai đoạn sau:

- Giai đoạn 1: Tạo các mẫu nhiễu tham chiếu cho chất liệu.
- Giai đoạn 2: Phát hiện chất liệu bằng cách tìm độ tương quan của từng mẫu nhiễu chất liệu trong ảnh đầu vào với mẫu nhiễu tham chiếu.

Tạo mẫu nhiễu tham chiếu cho chất liệu

Với mỗi chất liệu M cho trước ta thu nhận được một cơ sở dữ liệu ảnh bao gồm N_p bức ảnh $I_k, k=1, \dots, N_p$ ($N_p > 50$), sau đó ta thực hiện lần lượt các bước sau:

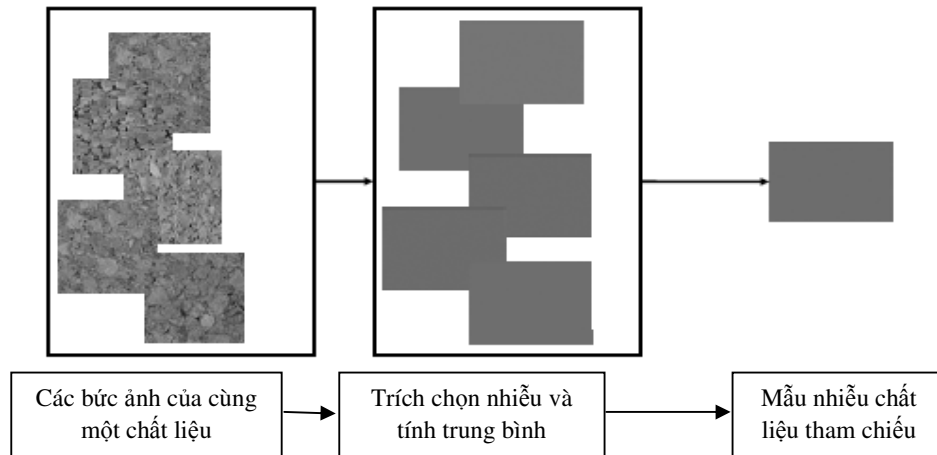
- Khử nhiễu tất cả các bức ảnh của cùng một chất liệu đó qua hàm khử nhiễu F (sử dụng bộ lọc khử nhiễu wavelet) ta được các ảnh sau khử nhiễu $F(I_k)$.

- Tính nhiễu dư của từng ảnh $n(I_k)$ như sau:

$$n(I_k) = I_k - F(I_k) \quad (6)$$

- Tính trung bình tất cả các mẫu nhiễu dư ta thu được mẫu tham chiếu của chất liệu M như sau:

$$P_M = \frac{\sum_{k=1}^{N_p} n(I_k)}{N_p} \quad (7)$$



Hình 5. Mô hình tạo nhiễu tham chiếu chất liệu

Do quá trình thu thập dữ liệu của một chất liệu phải được thực hiện trước khi tạo mẫu và tạo một mẫu nhiễu chất liệu với lượng dữ liệu lớn tại một thời điểm thường tốn thời gian rất lớn. Để đảm bảo cho quá trình tạo mẫu tham chiếu được thực hiện cùng với quá trình thu thập dữ liệu tham chiếu theo thời gian, chúng tôi đã đề xuất giải pháp tạo mẫu nhiễu theo phương pháp tích lũy dần, ban đầu việc tạo nhiễu chỉ thực hiện với một số ít mẫu dữ liệu các bức ảnh chất liệu đó, sau đó với mỗi mẫu dữ liệu thu nhận được chúng ta tiếp tục thực hiện quá trình tạo mẫu tham

chiếu với mẫu dữ liệu đưa vào và mẫu tham chiếu đã có sẵn, với các mẫu mới thêm vào thì quá trình tính toán được thực hiện tương tự cho đến khi số bức ảnh N_p đủ lớn ($N_p > 50$). Giả sử có

$$N_p \text{ bức ảnh ban đầu. Ta có mẫu nhiễu tính được: } P_M = \frac{\sum_{k=1}^{N_p} n(I_k)}{N_p}$$

Khi đó nếu có một ảnh T bất kỳ và yêu cầu T là mẫu dữ liệu để tạo mẫu nhiễu. Khi đó ta có thể làm như sau: Gọi P_{MT} là mẫu nhiễu sẽ được tạo sau khi thêm T. Ta có công thức được biến đổi:

$$\begin{aligned} P_{MT} &= \frac{1}{N_p + 1} \left(\sum_{i=1}^{N_p} n(I_i) + n(T) \right) \\ \Leftrightarrow (N_p + 1)P_{MT} &= \sum_{i=1}^{N_p} n(I_i) + n(T) \\ \Leftrightarrow (N_p + 1)P_{MT} &= N_p P_M + n(T) \Leftrightarrow P_{MT} = \frac{1}{(N_p + 1)} (N_p P_M + n(T)) \quad (8) \end{aligned}$$

Đây chính là phép lấy trung bình các ảnh nhiễu theo phương pháp tích lũy mà chúng tôi đã sử dụng trong quá trình xây dựng mẫu nhiễu.

Thuật toán phát hiện chất liệu dựa vào nhiễu – IMBN_Detection

Thuật toán IMBN_Detection nhằm phát hiện chất liệu dựa vào lượng nhiễu dư đặc trưng của chất bao gồm các bước sau:

Bước 1: Tính toán mẫu nhiễu chất liệu tham chiếu theo công thức (8) và xác định ảnh nhiễu của ảnh đầu vào.

Bước 2 : Với mỗi vùng \mathfrak{R} trong ảnh.

Bước 3: Tính độ tương quan giữa vùng \mathfrak{R} và mẫu tham chiếu $P(\mathfrak{R})$ theo (4) ta được $\rho_{\mathfrak{R}}$.

Bước 4: Áp dụng hàm phân loại tích lũy Gauss với giá trị $\rho_{\mathfrak{R}}$ ta được $G(\rho_{\mathfrak{R}})$, sau đó tính giá trị $p_{\mathfrak{R}} = 1 - G(\rho_{\mathfrak{R}})$

Bước 5: Kiểm tra nếu $P_{\mathfrak{R}} < \alpha = 0.01$ thì kết luận \mathfrak{R} chính là vùng chất liệu cần phát hiện. Ngược lại thì không phải.

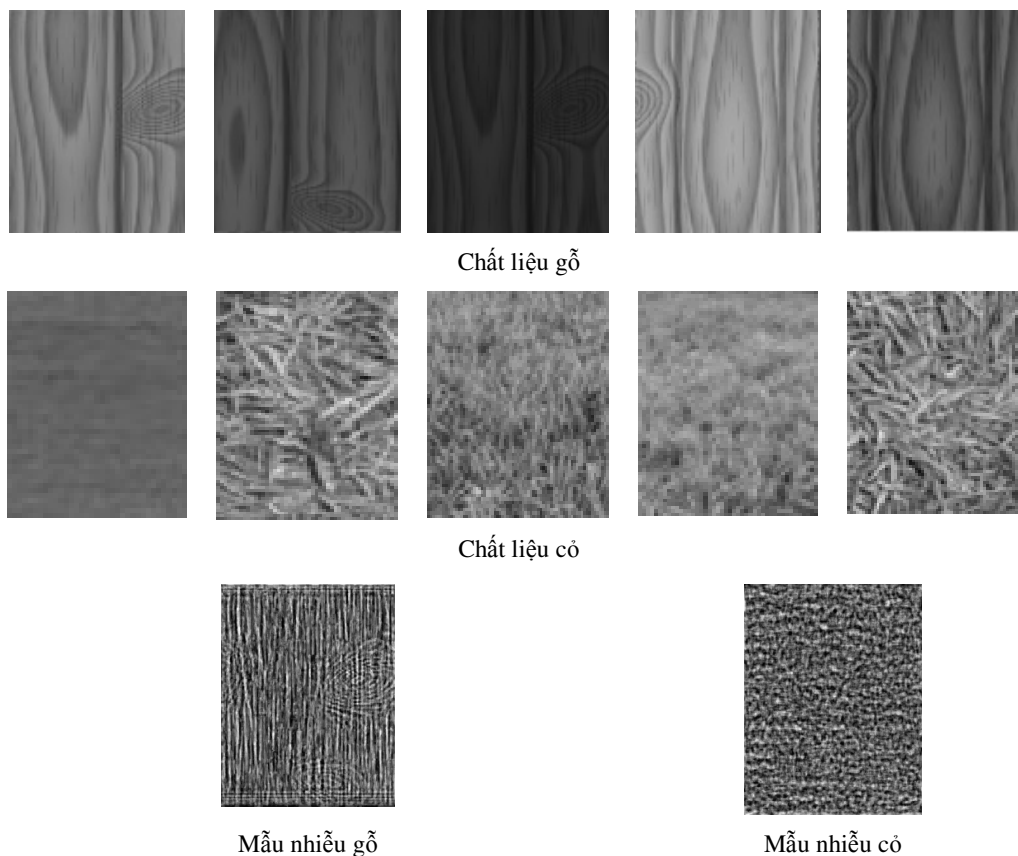
Đánh giá độ phức tạp thuật toán IMBN_Detection

- Tạo mẫu nhiễu: Chúng tôi dùng phương pháp tính trung bình nhiễu dựa trên kỹ thuật tích lũy. Cơ sở chủ yếu là phép cộng trừ ảnh.

- Phát hiện chất liệu dựa trên việc dò tìm các mẫu nhiễu chất liệu không tương đồng với mẫu chất liệu tham chiếu, thực chất vẫn là phép trừ các vùng ảnh với nhau. Do chúng tôi duyệt toàn bộ bức ảnh với độ trượt một pixel, nên dễ dàng tính toán được độ phức tạp tính toán trong trường hợp này là $O(M \times N)$. Nếu chúng ta vùng lấy $R(K, I)$, $K \ll M$, $I \ll N$. Thì kết quả độ phức tạp sẽ là $O(M \times N \times K \times I)$ xấp xỉ tối đa là $O(M^2 \times N^2)$.

4. THỰC NGHIỆM

Chúng tôi đã cài đặt thử nghiệm cho thuật toán IMBN_Detection được đề xuất ở trên, bởi ngôn Visual C++ 9.0, với cấu hình máy intel pentium dual core ≥ 2.0 .2GB RAM. Việc xây dựng mẫu nhiễu cho từng loại chất liệu, mỗi loại chất liệu chúng tôi tính toán trung bình nhiều trên 50 bức ảnh với các điều kiện môi trường khác nhau. Thời gian tính toán khoảng 2s cho một bức ảnh kích thước 250×180. Hình 6 chúng tôi minh họa mẫu nhiễu của một số ảnh của cùng một chất liệu, qua đó chúng ta thấy cách tiếp cận này không những phân biệt các loại chất liệu khác nhau mà còn có thể phân biệt được các kiểu của một chất liệu.



Hình 6. Minh họa các mẫu nhiễu của một số chất liệu

Hình 7a là ảnh thu nhận với chất liệu cỏ điển hình. Hình 7b là kết quả phát hiện chất liệu cỏ, vùng được cho là cỏ được khoanh bởi các khoanh màu trong ảnh. Hình 8a là một ảnh thu nhận khác với nhiều loại chất liệu khác nhau. Hình 8b là kết quả phát hiện chất liệu, với các loại chất liệu khác nhau được khoanh bởi các vùng có viền màu khác nhau theo cấu trúc số hiệu của chất liệu.



a) Ảnh thu nhận



b) Ảnh kết quả sau khi thực hiện phát hiện chất liệu

Hình 7. Phát hiện chất liệu cỏ



a) Ảnh thu nhận



b) Ảnh kết quả sau khi thực hiện phát hiện chất liệu gỗ và cỏ

Hình 8. Minh họa kết quả thuật toán

5. KẾT LUẬN

Ảnh thu được sau quá trình thu nhận ảnh thường không tránh khỏi nhiễu gây ra bởi bản thân đối tượng tạo nên và nhiễu do môi trường. Mỗi chất liệu khác nhau sẽ có một lượng nhiễu khác nhau đặc trưng cho chất liệu đó, lượng nhiễu đặc trưng cho mỗi chất liệu được gọi là nhiễu chất liệu. Dựa vào đặc điểm này bài báo đề xuất một mô hình nhiễu chất liệu, trên cơ sở đó đưa ra một thuật toán phát hiện chất liệu dựa vào nhiễu.

Kĩ thuật đề xuất đã được cài đặt thử nghiệm phát hiện chất liệu dựa trên tính toán độ tương quan giữa ảnh và mẫu nhiễu tham chiếu của chất liệu cho trước. Thuật toán tỏ ra có hiệu quả đối với các chất liệu có sự tương đồng về độ sáng nhưng thô như cỏ, gỗ,...

Thuật toán đã phát hiện được nhiều chất liệu khá chính xác, song độ phức tạp tính toán vẫn còn phải được nghiên cứu. Trong thời gian tiếp theo chúng tôi sẽ tiếp tục nghiên cứu các chiến lược tối ưu về thời gian tính toán cũng như các mô hình xác định nhiễu để có thể áp dụng cho một số chất liệu có cấu trúc mịn, bóng như gương.

TÀI LIỆU THAM KHẢO

1. J. Y. Tou, Y. H. Tay, and P. Y. Lau - A Comparative Study for Texture Classification Techniques on Wood Species Recognition Problem, ICNC **5** (2009) 8-12.
2. G. Liu, Z. Lin, and Y. Yu - Randon Representation-Based Feature Descriptor for Texture Classification, IEEE TIP **18** (5) (2009) 921-928.
3. Marko Heikkila, Matti Peaitikanen, Cordilia Schmid - Description of Interest Regions with Local Binary Patterns, Pattern Recognition **42** (2009) 425-436.
4. Đỗ Năng Toàn, Hà Xuân Trường - Tiếp cận phát hiện ảnh giả mạo dựa trên mẫu nhiễu cảm biến, Kỷ yếu Hội Thảo Quốc gia về “Các vấn đề chọn lọc của CNTT”, Đại Lải 14-15/09/2007, 2008, tr. 566-577.
5. M. Takagi and H. Fujiyoshi - Road Sign Recognition using SIFT feature, Symposium on Sensing via Image Information, 2007, LD2-06.
6. M. Varma and D. Ray - Learning the discriminative power-invariance trade-off, In ICCV, 2007.
7. Jan Lukáš, Jessica Fridrich, and Miroslav Goljan - Detecting Digital Image Forgeries Using Sensor Pattern Noise. Department of Electrical and Computer Engineering SUNY Binghamton, Binghamton, NY 13902-6000, 2005.
8. D. G. Lowe - Distinctive Image Features from Scale Invariant Keypoints, Journal of Computer Vision **60** (2) (2004) 91-110.
9. [http:// research.microsoft.com/maniks](http://research.microsoft.com/maniks)

SUMMARY

A NOISED MODEL AND ITS APPLICATION FOR DETECTING IMAGE MATERIALS

Images always have noises due to image acquisition processes. Noises are created by self of objects and environments. Each of materials has specific number of noises, which characterizes that material. Noise features of the material are called material noises. This paper deals with a technique for detecting image materials which base on noise analysis of images. By experiments, the technique exposed fairly accurate results for some natural materials at first, especially homogeneous materials about illuminates and extracted invariant features is sparse.

Keywords. Noises, materials, invariant feature, reduce noise, wavelet, texture feature.

Địa chỉ:

Nhận bài ngày 10 tháng 5 năm 2009

Đỗ Năng Toàn,
Viện Công nghệ thông tin.
Lê Thị Kim Nga, Trường Đại học Quy Nhơn.
Nguyễn Thị Hồng Minh,
Trường Đại học Quốc gia Hà Nội.