

## Chương 2

# Phương pháp từ điển Bag of Word

---

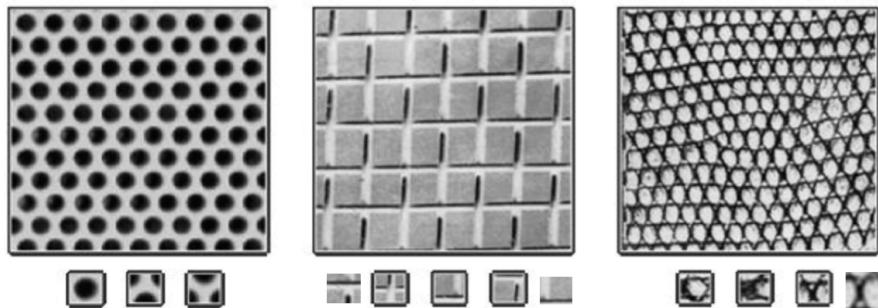
### 2.1 Giới thiệu

Phương pháp từ điển Bag of Word (BOW) trong gán nhãn và phân loại ảnh dựa trên ý tưởng thuật toán mã hoá, quản lý và tìm kiếm văn bản chữ. Theo đó thì, để dễ dàng quản lý và tìm kiếm các văn bản, một từ điển các từ khoá "word" được xây dựng dựa trên nội dung của toàn bộ văn bản. Mỗi văn bản văn bản được duyệt và được mã hoá lại thành dạng vector, có chiều dài bằng số "word" có trong từ điển; chỉ số tương ứng với vị trí của các từ khoá trong từ điển, đồng thời giá trị các thành phần của vector là số lần xuất hiện của từ khoá. Nếu một từ khoá nào đó không xuất hiện, giá trị vector tại chỉ số đó bằng 0.

Để hiểu rõ hơn, ta có thể xem ví dụ sau. Giả sử ta có 2 văn bản "*John likes football. Mary likes football too*" và "*John also likes cooking*". Từ điển được xây dựng từ 2 văn bản trên sẽ gồm tất cả các "word" xuất hiện trong hai văn bản: { "John", "likes", "football", "Mary", "too", "also", "cooking" }. Tương ứng với từ điển trên, văn bản đầu tiên được mã hoá thành dạng vector là {1, 2, 2, 1, 1, 0, 0}. Điều này bởi vì: trong văn bản đầu tiên, từ khoá "John" xuất hiện 1 lần, "likes" xuất hiện 2 lần, "football" xuất hiện 2 lần, "Mary", "too" mỗi từ xuất hiện một lần; nhưng "also", "cooking" không xuất hiện. Cũng theo cách đó, văn bản còn lại sẽ được mã hoá thành {1, 1, 0, 0, 0, 1, 1}. Bằng cách mã hoá này, chỉ cần lưu từ điển và các vector mã hoá, ta có được thông tin sơ lược về các văn bản. Hơn nữa với cách lưu trữ này còn cho phép tìm kiếm theo từ khoá nhanh chóng. Ví dụ ta cần tìm văn bản có từ khoá "football", ta tìm các tất cả các văn bản mà vector mã hoá có giá trị ở vị trí thứ 3 khác không. Giá trị vector tại vị trí thứ 3 càng lớn, chứng tỏ văn bản đó chứa càng nhiều từ khoá *football*, hay có nội dung càng gần với văn bản mong muốn. Ta cũng có thể sử dụng cách này để so sánh các văn bản, tìm kiếm các văn bản tương tự nhau hay phân loại các văn bản theo nội dung của chúng. Công việc này được thực hiện bằng cách so sánh các vector mã hoá. Về mặt toán học,

nếu hai văn bản có vector mã hoá là  $v_1$  và  $v_2$ . Sự tương tự giữa hai văn bản được định nghĩa là sự tương tự của hai vector hay khoảng cách giữa chúng. Có rất nhiều cách để đánh giá. Ví dụ, sử dụng hàm khoảng cách Euclid:  $d(v_1, v_2) = (v_1 - v_2)^2 \dots$

Ta hoàn toàn có thể áp dụng lí thuyết về xử lý văn bản cho ảnh bằng cách xây dựng các thuật toán mô tả ảnh bằng các từ khoá đặc trưng *visual word*. Ta xem xét một ví dụ cực kỳ đơn giản sau. Giả sử ta có ba ảnh như hình 2.1, mỗi ảnh đều có thể mô tả dưới dạng tổ hợp của một vài unit-block phía bên dưới. Bằng cách chọn tập từ điển là



Hình 2.1: Ví dụ mô tả ảnh dưới dạng tổ hợp của các *visual word*

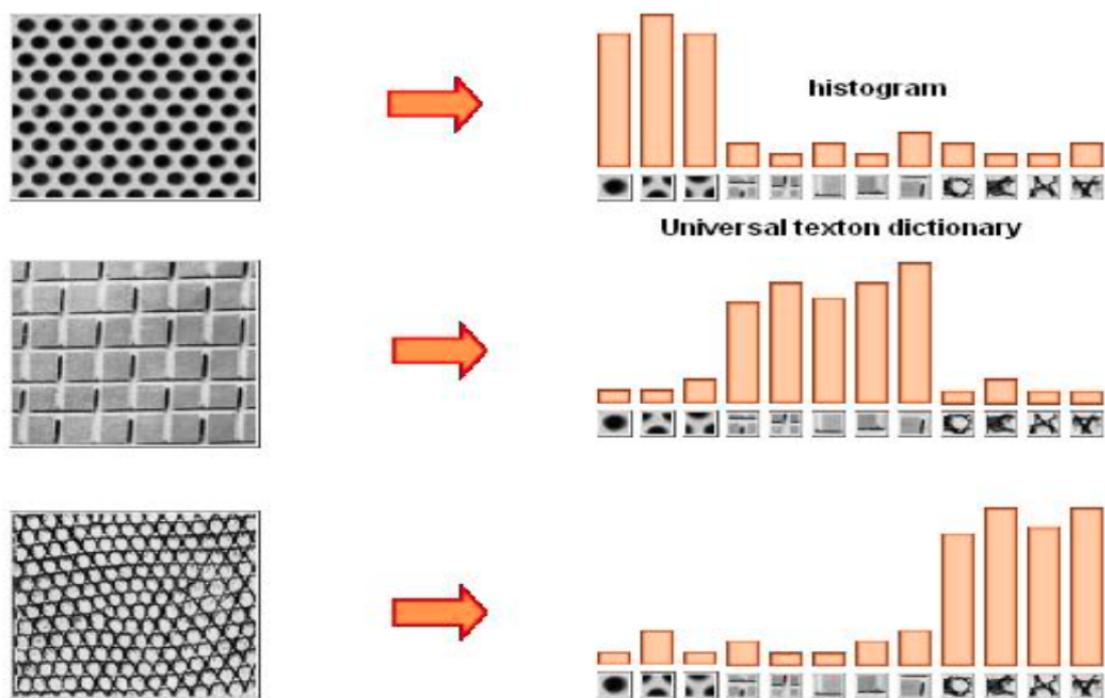
tập hợp tất cả các unit-block, ta có thể biểu diễn lại ba ảnh trên dưới dạng vector tần suất xuất hiện của các “*visual word*” như hình 2.2. So sánh sự tương tự về nội dung của các ảnh bây giờ trở nên đơn giản hơn nhiều, chỉ bằng cách so sánh các vector biểu diễn với nhau.

Như vậy, việc áp dụng thuật toán BOW cho ảnh là hoàn toàn khả thi. Vấn đề lớn nhất là làm thế nào để biểu diễn ảnh bằng một số hữu hạn các từ khoá? Khi tập ảnh trở nên lớn hơn, đa dạng và phức tạp hơn, việc chọn tập các từ khoá dùng để mô tả ảnh sẽ như thế nào (số lượng từ, gồm các từ khoá nào...) “*Visual word*” không giống với các “*word*” trong văn bản, có ý nghĩa rõ ràng, có thể phân biệt nhau. “*Visual word*” được xây dựng từ các các block ảnh chứa các đặc trưng thuộc tính ảnh. Do nó được trích xuất từ ảnh, nên nó có vô hạn trạng thái và khó phân biệt với nhau.

Phần bên dưới sẽ lần lượt trình bày cụ thể các bước thực hiện thuật toán BOW và nguyên lý/nội dung của từng bước cho ảnh.

## 2.2 Nguyên lý

Thực hiện mô tả ảnh sử dụng thuật toán từ điển BOW có thể chia làm 3 bước quan trọng:



Hình 2.2: Ví dụ mã hoá ảnh thành dạng vector của các visual word

- Trích xuất các đặc trưng của ảnh.
- Xây dựng từ điển đặc trưng: từ tập hợp các đặc trưng cơ bản lấy từ toàn bộ dữ liệu, chúng ta cố gắng xây dựng một từ điển hữu hạn nhưng có thể bao quát được nội dung của tập ảnh. Trong bước này có 2 tiêu chí quan trọng là: xác định loại thuộc tính của ảnh muốn hoặc cần sử dụng (màu sắc, đặc tính, hình dạng...) và kích thước của từ điển (số lượng từ khoá "visual word" cần sử dụng).
- Biểu diễn lại ảnh sử dụng từ điển các từ khoá đặc trưng "visual word".

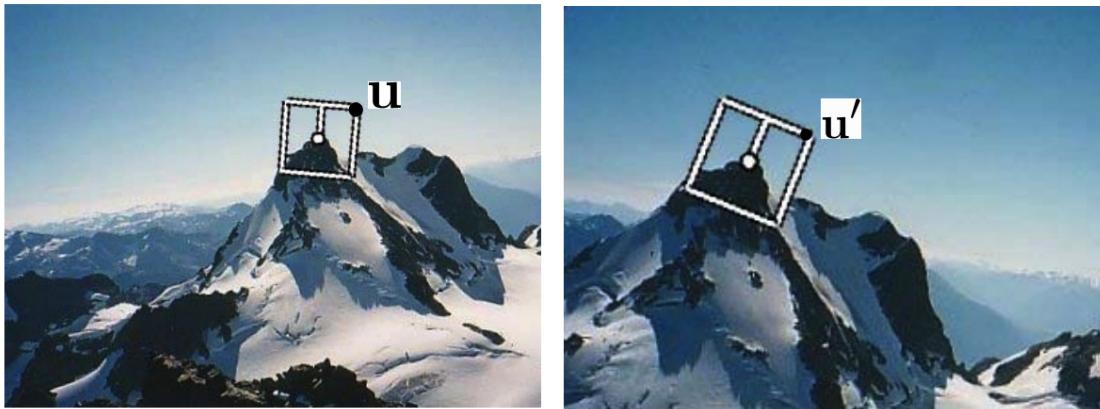
### 2.3 Trích xuất các đặc trưng của ảnh

Có thể bạn sẽ thắc mắc tại sao chúng ta phải thực hiện trích xuất các đặc trưng của ảnh và các đặc trưng của ảnh phải đảm bảo yêu cầu gì? Như đã giới thiệu ở bên trên, thuật toán từ điển BOW cố gắng mô tả tập ảnh bằng một số hữu hạn các "visual word". Đây là một công việc cực kỳ khó khăn bởi vì:

- Số lượng ảnh rất lớn, số lượng tổ hợp các blocks ảnh là vô hạn, điều đó có nghĩa là ta không thể dùng trực tiếp block ảnh làm "visual word" bởi vì kích thước từ điển là vô hạn. Như vậy, bằng mọi cách ta phải trích xuất ra các đặc trưng cơ bản nhất của ảnh, có thể đại diện cho ảnh, để xây dựng từ điển "visual words"
- Ảnh có thể chứa cùng một vật, nhưng ở nhiều scale khác nhau. Làm thế nào để thể hiện một block ảnh ở các scale khác nhau (ví dụ như 8x8 giống với block ảnh 16x16) khi có nội dung như nhau. Hay nói cách khác, làm thế nào để có thể trích xuất ra cùng một đặc trưng của ảnh tại các scale khác nhau.
- Tương tự, ảnh chụp cùng một đối tượng nhưng tại nhiều góc độ khác nhau (do không gian là 3 chiều), ví dụ chỉ cần xoay nhẹ hoặc lắc tay nhẹ là có thể có 2 ảnh trông có vẻ giống nhau nhưng thực tế 2 ma trận pixel biểu diễn ảnh khác xa nhau; làm sao có thể so sánh trích xuất đặc trưng của ảnh có cùng nội dung nhưng bị xoay các góc khác nhau?
- Ảnh chụp cùng một ảnh tại các thời điểm khác nhau, có cường độ sáng khác nhau, độ tương phản rõ nét khác nhau, dẫn đến các block ảnh khác nhau về mặt giá trị các pixel nhưng mắt thường ta thấy rõ chúng giống nhau. Làm thế nào để miêu tả nội dung của các ảnh này bằng các đặc trưng giống nhau?

Ví dụ trên hình 2.3 thể hiện điều này. Hai bức ảnh về cùng một ngọn núi, nhưng được chụp tại các thời điểm khác nhau, với góc chụp và với độ phóng khác nhau. Gọi  $u$  và  $u'$

là các block ảnh được lấy ra từ hai ảnh. Rõ ràng  $u, u'$  có cùng nội dung nhưng có kích thước khác nhau, có mức tương phản sáng tối khác nhau, bị quay các góc khác nhau. Làm thế nào để biểu diễn chúng bằng cùng một từ khoá “visual word”. Các visual word cần phải bất biến với scale, góc quay và sự thay đổi cường độ sáng.

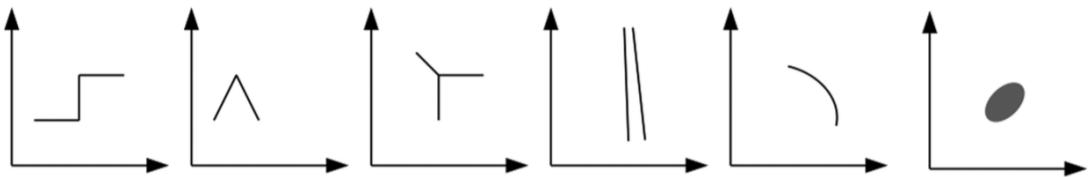


**Hình 2.3:** Ví dụ về khó khăn khi biểu diễn ảnh bằng visual word: các visual word cần phải bất biến với scale, góc quay và sự thay đổi cường độ sáng. Hai block ảnh  $u$  và  $u'$  nên được mã hoá bởi cùng một từ khoá dù chúng có kích thước khác nhau, chụp với góc chụp khác nhau, với cường độ sáng khác nhau.

Các nghiên cứu về trích xuất các đặc trưng của ảnh được tiến hành từ những năm 80 của thế kỷ trước. Một bản tổng kết chi tiết các nghiên cứu này có thể tìm thấy trong [1]. Các thuật toán trích xuất đặc trưng ảnh có thể chia làm 2 loại chính: trích xuất dựa trên các điểm nổi bật (interesting point) và trích xuất đều (dense sampling). Tuỳ vào ứng dụng (phân loại ảnh, nhận diện đối tượng, tìm kiếm đối tượng trong ảnh...), loại dữ liệu (ảnh phong cảnh, ảnh đối tượng) ta lựa chọn thực hiện các thuật toán trích xuất khác nhau với các loại đặc trưng khác nhau. Ví dụ như nếu sử dụng cho mục đích phân loại ảnh, thì cả phần nền của ảnh cũng có vai trò của nó. Vì vậy tất cả thông tin ảnh đều phải được sử dụng. Ta thực hiện trích xuất đều bằng cách lấy mẫu đều đặn trên ảnh (dense sampling). Ngược lại, khi thực hiện nhận diện đối tượng, chỉ có đối tượng là quan trọng, ta thực hiện thuật toán trích xuất dựa trên điểm nổi bật (interesting point).

### 2.3.1 Trích xuất dựa trên điểm nổi bật

Các thuật toán trích xuất điểm nổi bật tập trung vào tìm kiếm các vùng hấp dẫn cao (high interesting locations) trong ảnh. Đó có thể là các điểm, góc, cạnh (points, corners, blobs and high texture region). Ví dụ hình 2.4. Các thuật toán đầu tiên



Hình 2.4: Ví dụ về một số kiểu điểm nổi bật trong ảnh. Từ trái qua phải: Step, roof, corner, line/edge, ridge/contour, region.

về trích xuất điểm nổi bật là *Harris detector* [1] và *Hessian detector* [2]. Trong khi thuật toán Harris-detector tập trung tìm kiếm các cấu trúc góc cạnh trong ảnh, thuật toán Hessian-detector lại cố gắng tìm kiếm các vùng có sự thay đổi mạnh trong ảnh (strong texture variation). Hai thuật toán Harris-detector và Hessian-detector có thể dễ dàng tìm kiếm và phát hiện được các vùng ảnh bị quay, cường độ sáng thay đổi [3], tuy nhiên lại không thể phát hiện và tìm kiếm các vùng ảnh bị thay đổi scale. Năm 1998, [2] giới thiệu thuật toán *Laplacian of Gaussian (LoG)* detector tìm kiếm các vùng có đặc trưng cao, hơn nữa nhờ việc áp dụng kỹ thuật xử lý block ảnh dạng kim tự tháp, thuật toán còn cho phép tìm kiếm các vùng đặc trưng ảnh tại nhiều scale khác nhau. Tương tự, [4] giới thiệu *Difference of Gaussian (DoG)* trong đó thay thế toán tử Laplace bằng đạo hàm bậc nhất Gaussian, cho phép tính toán đơn giản hơn nhưng có hiệu quả tương đương với thuật toán LoG, cho phép trích xuất các block ảnh tại nhiều scale. Gần đây nhất, [5] đề xuất kết hợp toán tử LoG với thuật toán Harris-detector và Hessian-detector tạo ra *Harris-Laplace detector* and *Hessian-Laplace detector* cho phép trích xuất các vùng ảnh bất biến với sự thay đổi scale của ảnh. Trong nghiên cứu sau đó một năm [6] họ lại mở rộng hai thuật toán trên và giới thiệu *Harris-Affine*, *Hessian-Affine* cho phép trích xuất các vùng ảnh bất biến với các phép thay đổi không gian Affine transformation (phép dịch, phép quay trong ảnh). Hai thuật toán này trở thành phương pháp trích xuất điểm nổi bật được dùng nhiều nhất hiện nay. Chi tiết về tất cả các phương pháp trích xuất điểm đặc trưng được tổng kết và trình bày trong bài tổng kết [6, 7].

### 2.3.2 Trích xuất đều

Ngược lại với trích xuất điểm nổi bật chỉ quan tâm đến những vị trí đặc biệt trong ảnh, phương pháp trích xuất đều lấy thông tin một cách đều đặn trên ảnh. Thông thường, ta tiến hành lấy mẫu dựa theo một lưới đều. Phương pháp này, có 2 giá trị phải quan tâm: kích thước block ảnh lấy mẫu  $n$ , độ dịch giữa các cửa sổ lấy mẫu hay độ chồng lấn giữa các cửa sổ lấy mẫu  $x$ . Ta thường chọn block ảnh lấy mẫu 8x8 pixel hoặc

16x16, độ dịch 8 pixel. Phương pháp trích xuất đều không thể trích xuất các đặc trưng của ảnh tại nhiều scale khác nhau. Để đảm bảo các block ảnh có thể chứa thông tin scale của ảnh, tại một vị trí lấy mẫu, đồng thời nhiều block ở các scale khác nhau cũng được sử dụng, như block 8x8, 12x12, 16x16, 20x20, 25x25.... Việc thay đổi kích thước block lấy mẫu cho phép trích xuất cùng một loại đặc trưng tại nhiều độ phân giải của ảnh. Phương pháp trích xuất đều thích hợp cho các ứng dụng phân loại và gán nhãn ảnh, khi thông tin nền cũng hữu ích trong phân loại ảnh. Hạn chế của phương pháp lấy mẫu đều là số lượng block ảnh được lấy ra rất lớn, dẫn đến các bước xử lý phía sau khó khăn. [8] đề nghị lấy ngẫu nhiên một tập nhỏ từ các mẫu làm đại diện cho cả tập block được lấy mẫu đều của ảnh và qua thực nghiệm cho kết quả gần như tương đương. Gần đây, [9] giới thiệu *Dense Interest point*, kết hợp tinh hoa của 2 phương pháp lấy mẫu đều và lấy mẫu điểm nổi bật, cho phép cải thiện tốc độ tính toán cũng như chất lượng của quá trình trích xuất mẫu đáng kể.

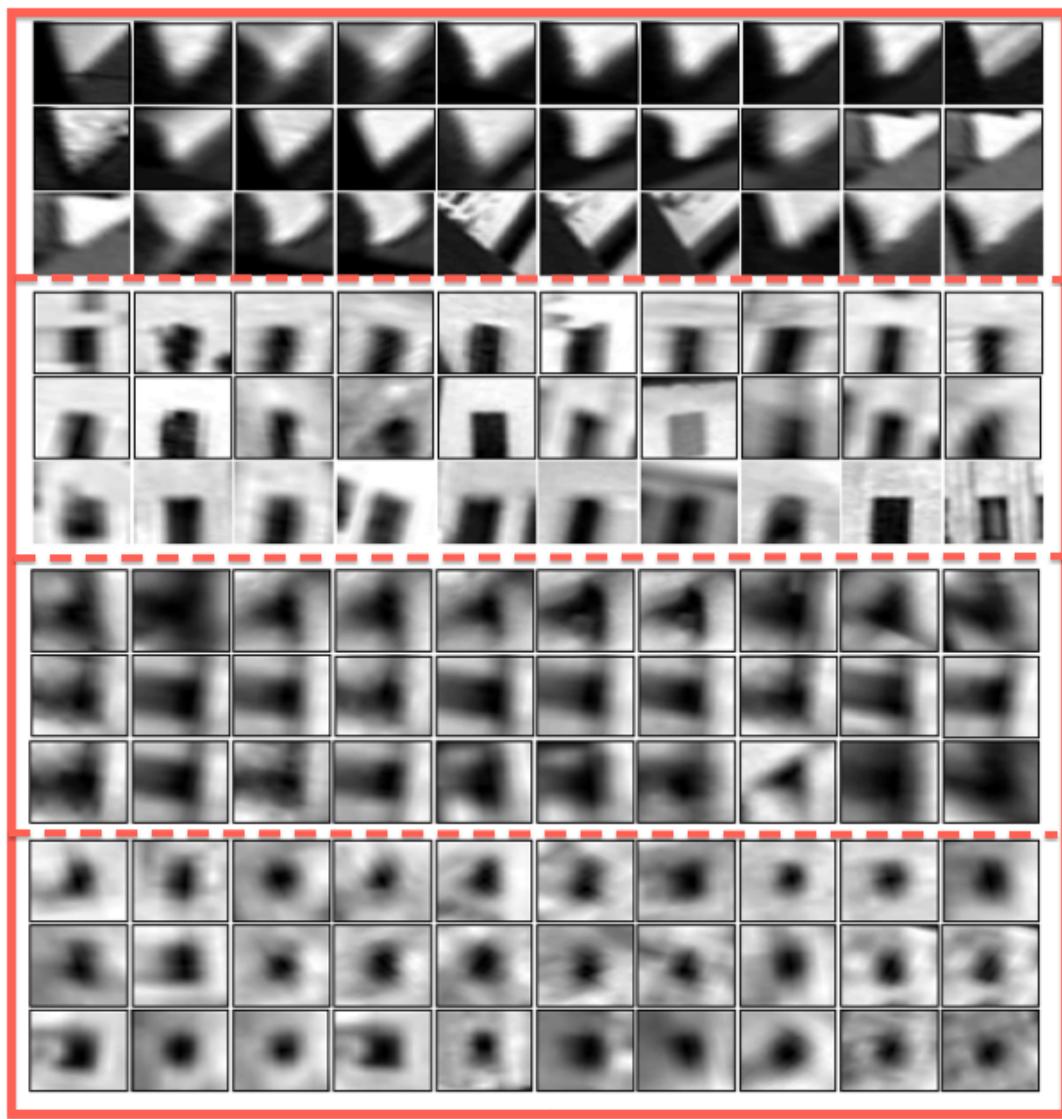
### 2.3.3 Mã hóa đặc trưng của ảnh

Như vậy bằng việc áp dụng các kỹ thuật trích xuất đặc trưng, ta thu được các block ảnh chứa các thông tin đặc trưng của ảnh. Câu hỏi đặt ra là chúng ta có thể sử dụng trực tiếp các đặc trưng này trong việc xây dựng từ điển đặc trưng không?

Câu trả lời là không. Lý do rất đơn giản: Các block ảnh có thể có kích thước khác nhau hoặc độ sáng tối khác nhau hoặc bị quay, hoặc ngay cả trông giống hệt nhau cũng không đảm bảo là 2 ma trận pixel của block ảnh có giá trị giống hệt (ví dụ như hình 2.5). Nói cách khác, mặc dù, bằng mắt thường, có thể thấy nội dung các block đó như nhau, thì làm sao có thể xác nhận một cách tự động (bằng máy tính) chúng giống nhau. Nếu không thể xác nhận chúng giống nhau, làm sao có thể gán chúng cho cùng một "word".

Do đó, cần có một bộ mã hóa đảm bảo: các block ảnh có nội dung như nhau, cho dù bị quay, bị scale, tại các cường độ sáng khác nhau khi qua bộ mã hóa này sẽ cho cùng một giá trị (hoặc một vector giá trị). Ta gọi các bộ này là các bộ mô tả đặc trưng *feature descriptor*.

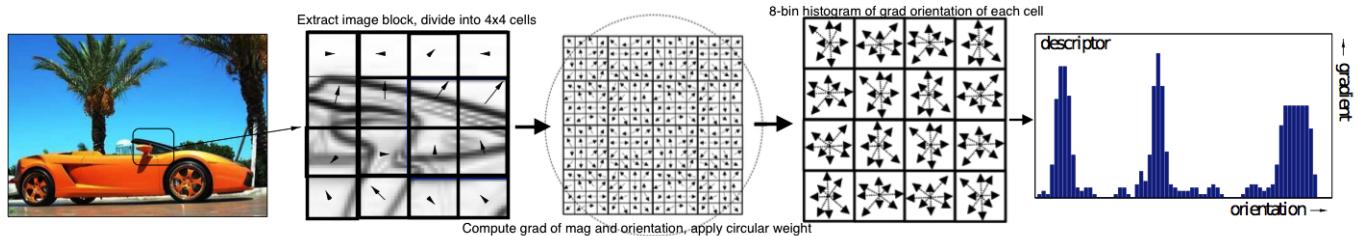
Có nhiều kỹ thuật mô tả đặc trưng. Tuỳ loại ảnh (ảnh phong cảnh, ảnh đối tượng, ảnh mặt người...), loại đặc trưng (colors, textures, shapes...), mục đích sử dụng (phân loại ảnh, nhận diện đối tượng...) có thể chọn sử dụng các kỹ thuật mã hóa đặc trưng khác nhau. Trong bảng 2.1 chúng tôi tóm tắt sơ lược một số loại descriptor thông dụng. Cụ thể chi tiết về phương pháp tính toán, ưu nhược điểm của chúng có thể xem trong tài liệu tham khảo hoặc bài báo tổng hợp thực hiện so sánh các phương pháp descriptor [].



Hình 2.5: Ví dụ về các block ảnh được biểu diễn thành bởi cùng một “visual word”. Bằng mắt thường, ta có thể thấy chúng tương đối giống nhau, tuy nhiên chúng rất khác nhau về mặt scale, độ sáng tối, bị quay, bị dịch, giá trị pixel...

Methods	Encode	Features	Robustness
SIFT	Textures	16x16 block at DoG point	Brightness, contrast, rot, scales, affine, noise
SUFT	Textures	Hessian point	scale, rot, illumination, noise
HOG	Textures	64x128 dense sampling	Illumination, viewpoint, scale, noise
BRISK	Textures	31x31 at FAST interesting points	Brightness, contrast, rota, scale
BRIEF	Textures	31x31 at interesting point	Brightness, contrast.
Daisy	Textures	Dense sampling	Illumination, occlusion, noise
Local Binary Pattern [ ]	Textures	3x3 block dense sampling	Brightness, contrast , rotate
PHOG	Shape	Dense sampling	illumination, viewpoint, noise
Shape context	Shape		
MSER	Shape	Interesting point	Scales, Affine, noise
Color SIFT	Color		
Color histogram	Color		
MPEG7	Color		

Bảng 2.1: *Feature descriptors and their properties*



Hình 2.6: Cách tính SIFT descriptor cho 1 block ảnh.

Trong phần này chúng tôi trình bày cụ thể về phương pháp dùng nhiều nhất, đơn giản nhất và có liên quan trực tiếp đến phương pháp từ điển. Đó là SIFT (sử dụng mã hoá đặc trưng texture)

### SIFT descriptor

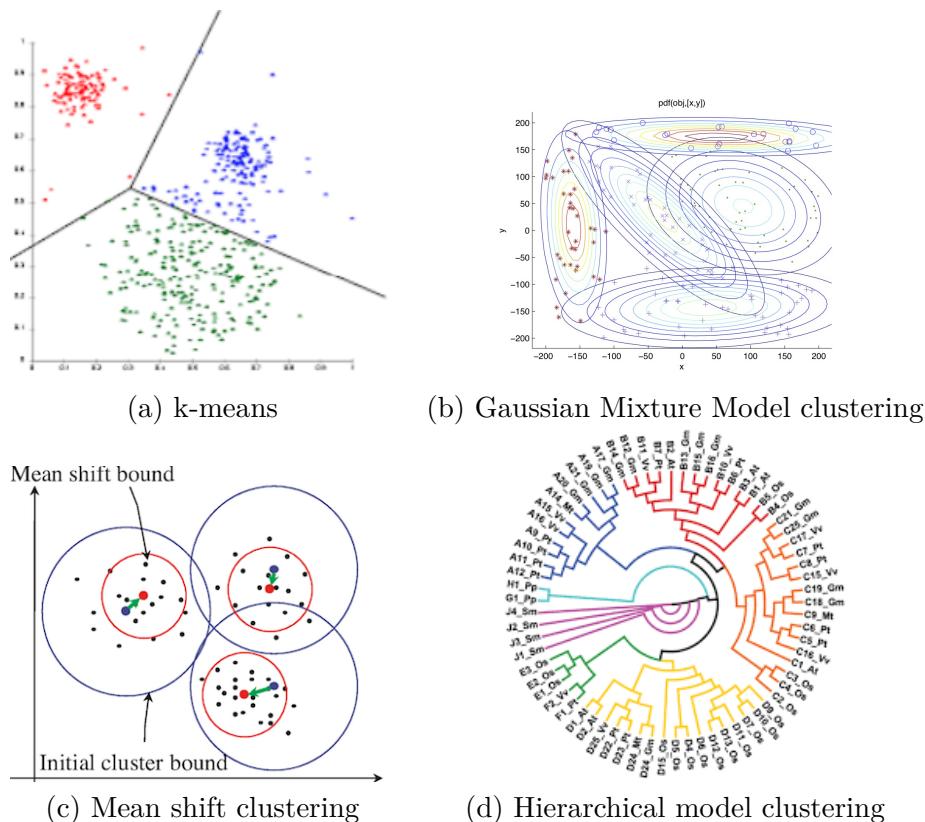
SIFT viết tắt của Scale Invariant Features Transform, giới thiệu bởi David Lowe năm 1994 [10], được coi là chìa khoá cho việc thực hiện thuật toán từ điển. SIFT bao gồm 2 phần: (i) trích xuất điểm nổi bật dựa trên toán tử đạo hàm Gaussian DoG và (ii) mã hoá các đặc trưng này thành dạng vector 128. Thực tế thì, nghiên cứu của [6] đã chứng minh rằng thuật toán SIFT descriptor có thể thực hiện cho mọi phương pháp trích xuất đặc trưng ảnh, cả trích xuất điểm nổi bật lẫn trích xuất đều.

Để tính toán SIFT descriptor, đầu tiên ta chia các block ảnh (được lấy xung quanh điểm nổi bật, hoặc điểm lấy mẫu đều) thành  $4 \times 4$  cell . Gradient of magnitude và gradient of orientation được áp dụng tại mỗi cell, sau đó ta tính histogram of 8-bin gradient orientation tại các cell đó. Kết quả ta thu được vector  $4 \times 4 \times 8$  or 128 bin. Tóm tắt cách tính SIFT được thể hiện tại hình 2.6.

Cách tính SIFT descriptor như trên cho phép SIFT bất biến với phép quay và thay đổi cường độ sáng trong ảnh. Hơn thế nữa, block ảnh có kích thước khác nhau, qua phép mã hoá SIFT đều có định dạng vector 128 chiều.

## 2.4 Xây dựng từ điển đặc trưng

Không giống như trong văn bản, các “word” xác định một cách rõ ràng và số lượng hữu hạn. Việc xây dựng từ điển các “word” trong văn bản đơn giản chỉ bằng cách tập hợp tất cả các word lại, có thể loại bỏ các word không quan trọng đi. Ngược lại với ảnh, các vector mô tả descriptor vẫn có vô số trạng thái. Để tạo một từ điển, ta phải thực hiện một cách xấp xỉ. Tất cả các mô tả descriptor lấy ra từ tập ảnh được sử dụng để tạo từ điển. Tuy nhiên nếu số lượng mô tả quá lớn, ta có thể chọn ngẫu nhiên 1/10 hoặc 1/20 số mô tả trên. Tập mô tả con này được phân cụm ra nhằm nhóm các mô tả tương tự nhau lại. Có rất nhiều thuật toán phân cụm như K-means [11, 12], Gaussian Mixture Models (GMM) [13], on-line clustering with mean-shift [14], and hierarchical clustering [15]. Ví dụ về một số phương pháp phân cụm được thể hiện ở hình 2.7. Khi phân cụm thành công, tâm điểm của cụm được gọi là “visual word”. Nói cách khác, số lượng cụm chính là số lượng “visual word” sử dụng trong phương pháp từ điển. Ở đây lưu ý là, trong các phương pháp phân cụm này, số lượng cụm hay số lượng visual word phải được người dùng xác định trước. Giá trị này, nếu quá nhỏ, các mô tả hoàn toàn khác nhau có thể được phân vào cùng một cụm, nói cách khác là được biểu diễn chung bằng một visual word. Điều này dẫn đến sai lầm trong phân loại ảnh dựa vào nội dung. Ngược lại, nếu số lượng visual word quá lớn, các block ảnh giống nhau có thể lại bị phân ra thành 2 cụm khác nhau. Để xác định được số cụm tối ưu cho mỗi tập ảnh, có thể ta phải tiến hành phân cụm nhiều lần, chạy trên tập



Hình 2.7: Several clustering metshods

huấn luyện để tìm ra giá trị thích hợp.

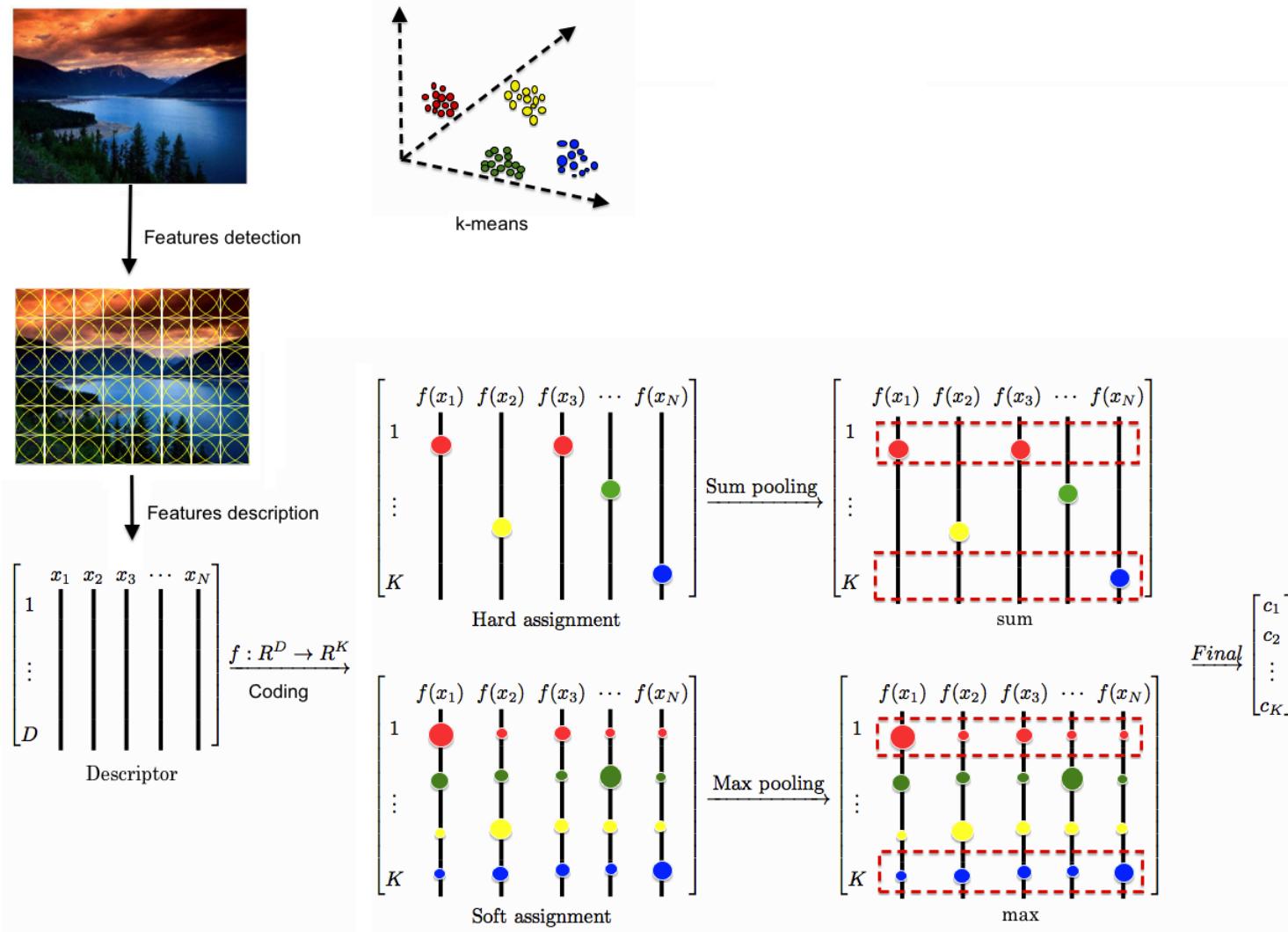
## 2.5 Biểu diễn ảnh dựa trên từ điển đặc trưng

Bây giờ ta đã có từ điển các visual words trong đó mỗi word có thể coi như đại diện cho một block ảnh. Làm thế nào để biểu diễn lại ảnh dựa trên các visual words này? Quay trở lại với trường hợp của văn bản text trong ví dụ ở mở đầu của chương này, ta thấy: để mã hoá một văn bản khi đã có từ điển, ta đếm số lần xuất hiện của mỗi word xuất hiện trong đó. Văn bản được biểu diễn dưới dạng vector có số chiều bằng số word trong từ điển, tại mỗi vị trí tương ứng với chỉ số của word đó trong từ điển có giá trị là tần suất xuất hiện của word đó trong văn bản. Như vậy ta cũng có thể làm tương tự với ảnh. Trước hết, tất cả các mô tả được gán với "visual word" mà nó được phân cụm. Mỗi ảnh được đại diện bởi một tập các đặc trưng, qua bước "mã hoá đặc trưng" trở thành một tập các mô tả, lúc này được biểu diễn thành một tập các "visual word". Bằng cách đếm tần suất xuất hiện các visual word xuất hiện, mỗi ảnh được biểu diễn thành một vector histogram của các word. Phương pháp này được gọi là *Hard Assignment Coding*. Bên cạnh đó, do các đặc trưng của ảnh (đại diện bởi các block ảnh) không có giống nhau cũng như khác nhau tuyệt đối, việc phân cụm các đặc trưng cũng chỉ là xấp xỉ nên việc gán trực tiếp một mô tả cho visual word của cụm nó được phân có vẻ không phải tối ưu vì nó bỏ qua mối liên hệ/ khả năng nó có thể được gán cho các word khác. [16] đề xuất phương pháp *Soft Assignment Coding* cho phép gán mỗi mô tả cho tất cả các word kèm theo một trọng số, được định nghĩa bằng hàm Gaussian của khoảng cách từ mô tả đó đến tâm điểm của mỗi phân cụm. Cách này tuy cải thiện được độ chính xác trong việc biểu diễn ảnh, nhưng phức tạp hơn rất nhiều. Gần đây, [17] giới thiệu *Semi-Soft Assignment Coding* cho phép gán mỗi mô tả chỉ bởi top-k phân cụm gần nhất. Phương pháp này cho thấy, nó đạt được độ chính xác như *Soft Assignment Coding*,

nhưng độ phức tạp lại rất thấp.

## 2.6 Kết luận

Chương này đã trình bày một cách tổng quan về phương pháp biểu diễn ảnh dựa theo từ điển các “visual word”. Sơ đồ tổng quát của phương pháp được tóm tắt lại trong hình 2.8. Sau toàn bộ quá trình, mỗi ảnh được biểu diễn thành dạng vector 1 chiều histogram của các visual word, cho phép dễ dàng áp dụng các kỹ thuật phân loại như K-NN, SVM, ANN nhằm mục đích phân loại, nhận diện ảnh.



Hình 2.8: Sơ đồ tóm tắt phương pháp biểu diễn ảnh dạng từ điển