

Computer Vision

MSc in Informatics, option Graphics, Vision and Robotics
James L. Crowley

Fall Semester

13 november 2008

Lesson 5

Bayesian Image Analysis and Recognition

Lesson Outline:

1	Invariant Image Description	2
1.1	Reminder : Discrete Gaussian Functions	2
1.2	Laplacian as Difference of Gaussian	3
1.3	Discrete Scale Space	3
1.4	Color Opponent Scale Space	5
1.5	Scale Invariant Interest Points	7
1.6	Scale Invariant Feature Transform (SIFT)	7
2	Bayesian Recognition	8
2.1	Baysian Classification	10
2.2	The probability of an event	10
2.3	The probability of the value of a random Variable.	12
2.4	Bayes Rule	12
3	Classification by Ratio of Histograms of pixel values	15
3.1	Histograms	15
3.2	Example: Object detection by pigment color	15
3.3	Histograms of Receptive Field Values	18

1 Invariant Image Description

1.1 Reminder : Discrete Gaussian Functions

Let $G(i, j, \sigma) = e^{-\frac{(j^2 + i^2)}{2\sigma^2}} = e^{-\frac{i^2}{2\sigma^2}} * e^{-\frac{j^2}{2\sigma^2}}$

$$\vec{\nabla}G(i, j, \sigma) = \begin{pmatrix} G_x(i, j, \sigma) \\ G_y(i, j, \sigma) \end{pmatrix}$$

For an image $P(i, j)$, the derivatives can be approximated by convolution with Derivatives of Gaussians

$$\begin{aligned} P_x(i, j) &\approx P * G_x(i, j, \sigma) \\ P_y(i, j) &\approx P * G_y(i, j, \sigma) \\ P_{xx}(i, j) &\approx P * G_{xx}(i, j, \sigma) \\ P_{xy}(i, j) &\approx P * G_{xy}(i, j, \sigma) \\ P_{yy}(i, j) &\approx P * G_{yy}(i, j, \sigma) \end{aligned}$$

Note: it is NECESSARY to specify σ . Small σ is not necessarily best.

The Gradient $\vec{\nabla}P(i, j)$ is calculated by $P(i, j) * \vec{\nabla}G(i, j, \sigma)$

where $\vec{\nabla}G(i, j, \sigma) = \begin{pmatrix} G_x(i, j, \sigma) \\ G_y(i, j, \sigma) \end{pmatrix}$

Gradient: $\vec{\nabla}P(i, j) = \begin{pmatrix} P_x(i, j) \\ P_y(i, j) \end{pmatrix} \approx \vec{\nabla}(P * G(i, j, \sigma)) = P * \vec{\nabla}G(i, j, \sigma) = \begin{pmatrix} P * G_x(i, j, \sigma) \\ P * G_y(i, j, \sigma) \end{pmatrix}$

Laplacien: $\nabla^2 P(i, j) = P * \nabla^2 G(i, j, \sigma) = P_{xx}(i, j) + P_{yy}(i, j) \approx P * G_{xx}(i, j, \sigma) + P * G_{yy}(i, j, \sigma)$

Gaussian Derivatives are Steerable:

$$G_1^\theta(x, y, \sigma) = \cos(\theta) \cdot G_x(x, y, \sigma) + \sin(\theta) \cdot G_y(x, y, \sigma)$$

Thus:

1st order $P_1^\theta(i, j) = \cos(\theta)P_x(i, j) + \sin(\theta)P_y(i, j)$

2nd order $P_2^\theta(i, j) = \cos(\theta)^2 P_{xx}(i, j) + \sin(\theta)^2 P_{yy}(i, j) + 2\cos(\theta)\sin(\theta)P_{xy}(i, j)$

3rd order

$$P_3^\theta(i, j) = \cos(\theta)^3 P_{xxx}(i, j) + \cos(\theta)^2 \sin(\theta) P_{xxy}(i, j) + \cos(\theta) \sin(\theta)^2 P_{xyy}(i, j) + \sin(\theta)^3 P_{yyy}(i, j)$$

By steering the derivatives to the local orientation, we obtain an "invariant" measure of local contrast. We can also "steer" in scale to obtain invariance to size.

Note, we can NOT steer the mixed derivatives, i.e $P_{xy}(i,j)$

1.2 Laplacian as Difference of Gaussian

Diffusion Equation: $\nabla^2 G_x(i,j,\sigma) = G_{xx}(i,j,\sigma) + G_{yy}(i,j,\sigma) = \frac{\partial G(i,j,\sigma)}{\partial \sigma}$

As a consequence: $\nabla^2 G(i,j,\sigma) \approx G(i,j,\sigma_1) - G(i,j,\sigma_2)$

and thus $\nabla^2 P(i,j) \approx P * G(i,j,\sigma_1) - P * G(i,j,\sigma_2)$

where $\sigma_1/\sigma_2 > \sqrt{2}$

1.3 Discrete Scale Space

Let $P(i,j)$ be an image of size $M \times M$ samples. The image $P(i,j,k)$ is sampled at a distance of $\Delta s_k = 2^{(k-1)/2}$

Let $G(i,j, 2^{k/2})$ be a Gaussian filter of $\sigma_k^2 = 2^k$

k	$\sigma_k^2 = 2^k$	$\sigma_k = 2^{k/2}$	$s_k = 2^{(k-1)/2}$
1	2	$\sqrt{2}$	1
2	4	2	$\sqrt{2}$
3	8	$2\sqrt{2}$	2
4	16	4	$2\sqrt{2}$
5	32	$4\sqrt{2}$	4
6	64	8	$4\sqrt{2}$
7	128	$8\sqrt{2}$	8
8	256	16	$8\sqrt{2}$
	512	$16\sqrt{2}$	16

Size of $G(i,j, 2^{k/2})$?

$R_k = 4\sigma_k \quad N_k = 2R_k + 1 = 8\sigma_k + 1 = 8(2^{k/2}) + 1 = 2^{k+3} + 1$

size $N_k \times N_k$ were $N_k = 2^{k+3} + 1$

Then a Discrete Scale Space is

$P(i,j,k) = P(i,j) * G(i,j, 2^{k/2})$

There exists a fast $O(N)$ pyramid algorithm for computing $P(i,j,k)$.

The pyramid automatically resamples each image at $\Delta s_k = 2^{(k-1)/2}$

The Gradient $\bar{\nabla}P(i, j)$ is calculated by $P(i, j) * \bar{\nabla}G(i, j, \sigma)$

With Discrete Scale space, for any point i, j at level k :

$$\begin{aligned} P_x(i, j, k) &= P(i+s_k, j, k) - P(i-s_k, j, k) \\ P_y(i, j, k) &= P(i, j+s_k, k) - P(i, j-s_k, k) \\ P_{xx}(i, j, k) &= P(i+s_k, j, k) - 2P(i, j, k) + P(i-s_k, j, k) \\ P_{yy}(i, j, k) &= P(i, j+s_k, k) - 2P(i, j, k) + P(i, j-s_k, k) \\ P_{xy}(i, j, k) &= P(i+s_k, j+s_k, k) - P(i-s_k, j+s_k, k) - P(i+s_k, j-s_k, k) + P(i-s_k, j-s_k, k) \end{aligned}$$

Where $s_k = \sigma_k/2 = 2^{(k-1)/2}$

Gradient: $\bar{\nabla}P(i, j, k) \approx \bar{\nabla}(P * G(i, j, 2^{k/2})) = P * \bar{\nabla}G(i, j, 2^{k/2}) = \begin{pmatrix} P_x(i, j, k) \\ P_y(i, j, k) \end{pmatrix}$

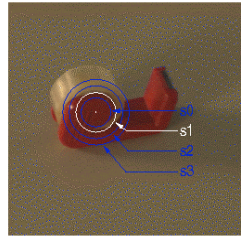
Laplacien: $\nabla^2 P(i, j, k) = P * \nabla^2 G(i, j, \sigma) = P_{xx}(i, j, k) + P_{yy}(i, j, k) \approx P(i, j, k) - P(i, j, k-1)$

$$\nabla^2 P(i, j, k) = P_{xx}(i, j, k) + P_{yy}(i, j, k) = P * \nabla^2 G(i, j, \sigma_k) \approx P * G(i, j, \sqrt{2}\sigma_k) - P * G(i, j, \sigma_k) = P(i, j, k+1) - P(i, j, k)$$

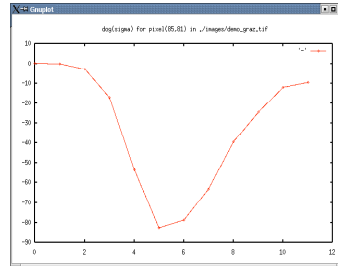
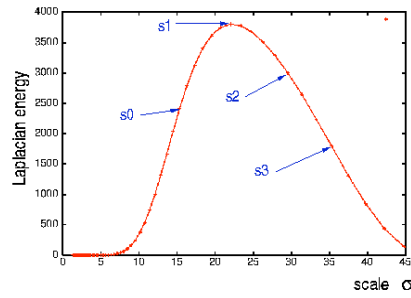
With this we can obtain a feature set for each pixel, i, j, k , for example.

$$\bar{X}(i, j, k) = \begin{pmatrix} P^L \\ P^{C_1} \\ P^{C_2} \\ P_x^L \\ P_y^L \\ P_x^{C_1} \\ P_y^{C_1} \\ P_x^{C_2} \\ P_y^{C_2} \\ P_{xx}^L \\ P_{xy}^L \\ P_{yy}^L \end{pmatrix}$$

At every point in the image, there will be some value of σ for which $\nabla^2 P(i, j, k)$ will be a maximum in σ .



zero crossing of Laplacian at s_i



The scale σ_i is an "invariant" for the appearance at $P(i,j)$.

$$\sigma_i = \arg\max_{\sigma} \{P * \nabla^2 G(i, j, \sigma)\}$$

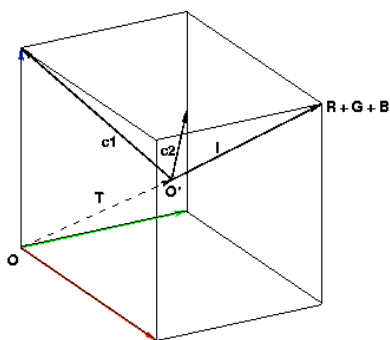
$$\sigma_i = \arg\max_{\sigma} \{\nabla^2_{\sigma=2^{k/2}} P(i, j)\}$$

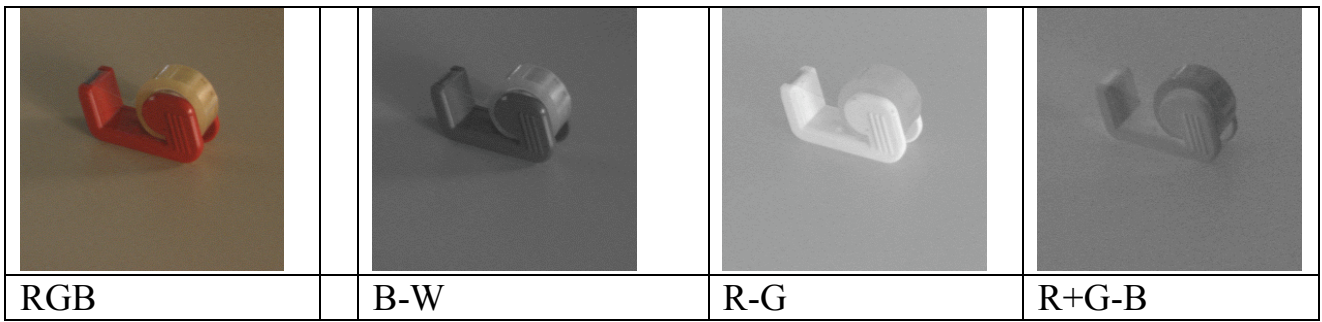
$$\sigma_i = 2^{k_i/2} \text{ where } k_i = \arg\max_k \{P(i, j, k) - P(i, j, k-1)\}$$

1.4 Color Opponent Scale Space

$$(R, G, B) \Rightarrow (L, C_1, C_2) \quad \begin{pmatrix} L \\ C_1 \\ C_2 \end{pmatrix} = \begin{pmatrix} 0.33 & 0.33 & 0.33 \\ -0.5 & -0.5 & 1 \\ 0.5 & -0.5 & 0 \end{pmatrix} \begin{pmatrix} R \\ G \\ B \end{pmatrix}$$

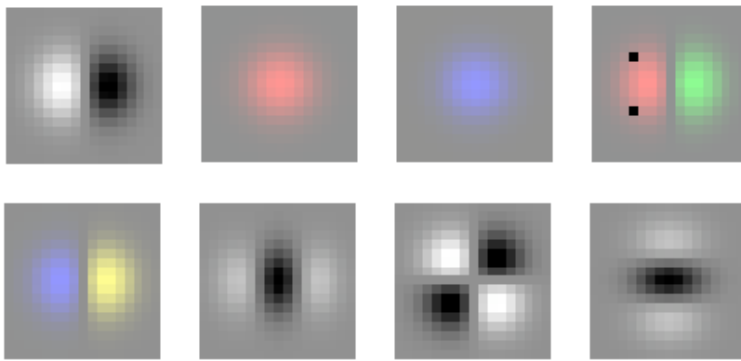
This representation separates luminance and chrominance.





This makes it possible to "steer" the chrominance to an illumination color

$$\begin{pmatrix} L \\ C_1 \\ C_2 \end{pmatrix} = \begin{pmatrix} 0.33 & 0.33 & 0.33 \\ -0.5 & -0.5 & 1 \\ 0.5 & -0.5 & 0 \end{pmatrix} \begin{pmatrix} \alpha_1 R \\ \alpha_2 G \\ \alpha_3 B \end{pmatrix}$$



1.5 Scale Invariant Interest Points

Let us define a "Local Maximum" operator as:

$$\{x_m\}_{\Delta x} = \underset{\Delta x}{\text{Arg-LocalMax}}\{P(x)\} \equiv \forall \varepsilon \leq \Delta x : P(x) \geq P(x + \varepsilon)$$

Local Max returns the set of values x_m for which $P(x)$ is larger than $P(x+\varepsilon)$ within a neighborhood $\pm s_k$.

For the derivatives in a discrete pyramid, $s_k = \sigma_k/2 = 2^{(k-1)/2}$

The Scale invariant "keypoints" are the local max of P in i, j, k :

$$\{\bar{X}_m\}_{i,j,k} = \underset{i,j,k}{\text{Arg-LocalMax}}\{\nabla^2 P(i, j, k)\}$$

Maximally stable invariant points are found as :

$$X(i, j, k) = \underset{i,j,k}{\text{arg-max}}\{P(i, j, k) - P(i, j, k-1)\}$$

1.6 Scale Invariant Feature Transform (SIFT)

Notes not yet available

2 Bayesian Recognition

Recognition is a fundamental ability for intelligence, and indeed for all life. To survive, any creature must be able to recognize food, enemies and friends.

Recognition: The fact to recognize, to identify an object as itself.

Identify: To recognize an entity as an individual

Classify: The recognize an individual as a member of a class.

A class is defined by a membership test.

Classification is a process of associated an entity (or an event) as a member of a class. The entity is described by a vector of features, provided by an observation.

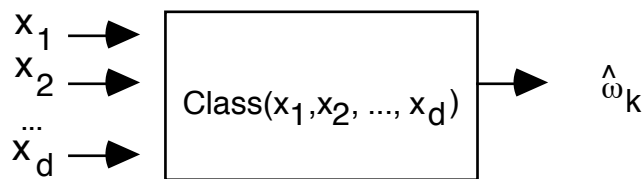
The assignment of an entity to a class provided by a test made on the feature vector.

Features: observable properties that permit discrimination between classes.

A set of D features, x_d , are assembled into a feature vector \vec{X}

$$\vec{X} = \begin{pmatrix} x_1 \\ x_2 \\ \dots \\ x_D \end{pmatrix}$$

For a feature vector, \vec{X} , a classifier is a process that proposes the identity of the class. This arrives in the form of a proposition $\hat{\omega}_k = E \in \text{Class } C_k$



The techniques from Pattern recognition and statistics provide a variety of methods to construct membership tests for classification of observations. The most appropriate technique depends on the number and nature of the classes and the features.

There are two families of technique: Generative and discriminative.

These correspond to the two methods to define a set:

Extension: Provide a list of members.

Intension: Provide a conjunction of predicates.

Generative methods compare the pattern to a set of prototype examples.

Discriminative methods apply a set of tests.

In either case, our objective is to minimize the probability of error.

$$\hat{\omega}_k = \underset{k}{\operatorname{arg-max}} \left\{ \Pr(E \in T_k \mid \vec{X}) \right\}$$

The operator "|" is called to as "given" or "provided that". It is the Bayesian conditional operator.

For a Generative method, we enumerate the M examples of the K classes. The estimate is the most similar, as provided by some similarity function.

Thus for an observed unknown even X, the "estimated" class, $\hat{\omega}_k$ is given by :

$$\hat{\omega}_k = \underset{k}{\forall_k} \underset{m}{\forall_m} : \operatorname{arg-max} \left\{ \operatorname{Sim}(\vec{X}, \vec{X}_m^k) \right\}$$

Simple Euclidean distance is often used to measure similarity.

$$\hat{\omega}_k = \underset{k}{\forall_k} \underset{m}{\forall_m} : \operatorname{arg-max} \left\{ \|\vec{X}, \vec{X}_m^k\| \right\}$$

A more intelligent method is to normalize distance by a Metric Λ_k

$$\hat{\omega}_k = \underset{k}{\forall_k} \underset{m}{\forall_m} : \operatorname{arg-max} \left\{ (\vec{X} - \vec{X}_m^k)^T \Lambda_k (\vec{X} - \vec{X}_m^k) \right\}$$

We can avoid having to scan all samples by replacing samples of the same class with the average of the samples.

$$\bar{\mu}_k = E\{\vec{X}_m^k\} = \frac{1}{M} \sum_{m=1}^M \vec{X}_m^k$$

Then:

$$\hat{\omega}_k = \underset{k}{\forall_k} : \operatorname{arg-max} \left\{ (\vec{X} - \bar{\mu}_k)^T \Lambda_k (\vec{X} - \bar{\mu}_k) \right\}$$

where the metric Λ_k is provide by the inverse of the class covariance : $\Lambda_k = C_k^{-1}$

$$C_k = E\{(\vec{X}_m^k - \bar{\mu}_k)^2\}$$

Discriminative tests avoid iterating through the M examples of each class by compiling a series of simple tests. These can be combined in a variety of ways.

A classical (and effective) means is by vote over as large set of simple linear classification functions. We will see more of this later.

2.1 Bayesian Classification

With a Bayesian approach, the tests are designed to minimize the number of errors.

False positives and false negatives count equally as errors.

An alternative would be to include the cost of error, which may not be the same for a false positive and a false negative. This is an easy extension.

Let ω_k be the proposition that the event belongs to class k: $\omega_k = E \in T_k$

ω_k	Proposition that event $E \in$ the class k
$p(\omega_k) = p(E \in T_k)$	Probability that E is a member of class k

Given an observation \vec{X} , the decision criteria is

$$p(\omega_k | \vec{X}) = \Pr(E \in T_k \text{ given } \vec{X})$$

$$\hat{\omega}_k = \arg\text{-max}_{\omega_k} \{ p(\omega_k | \vec{X}) \}$$

The meaning of "given" is provided by Bayes Rule:

$$p(\omega_k | \vec{X}) = \frac{p(\vec{X} | \omega_k)p(\omega_k)}{p(\vec{X})}$$

2.2 The probability of an event

There are two ways to define "probability": 1) Statistics : Frequency of Occurrence: The fraction of times that something is true, or 2) Probability: Using a systems of axioms.

A Frequency based, or statistical approach is more intuitive, but not always possible to apply. In some cases, the axioms of probability theory can provide a solution that is not possible from frequency of occurrence.

In either case, probability is a function that returns a number between 0 and 1.

$\Pr() \in [0, 1]$.

2.2.1 Probability as Frequency of Occurrence.

A frequency based definition of probability is sufficient for many practical vision problems.

Given M observations of random events, of which M_k belong to the class k . The probability of observing an event E of class k is

$$p(E \in A_k) \equiv \lim_{M \rightarrow \infty} \left\{ \frac{M_k}{M} \right\}$$

For the practical case where M is finite, $p(E \in \text{class } k) \approx \frac{M_k}{M}$

The precision of this approximation depends on the number of sample, M .

2.2.2 Axiomatic Definition

An axiomatic definition makes it possible to apply analytical techniques to the design of classification systems. Only three postulates (or axioms) are necessary: In the following, let E be an event, let S be the set of all events, and let A_k be set of events that belong to class k .

Postulate 1 : $\forall A_k \in S : p(E \in A_k) \geq 0$

Postulate 2 : $p(E \in S) = 1$

Postulate 3 :

$\forall A_i, A_j \in S$ such that $A_i \cap A_j = \emptyset : p(E \in A_i \cup A_j) = p(E \in A_i) + p(E \in A_j)$

2.3 The probability of the value of a random Variable.

Pour x entier, tel que $x \in [x_{\min}, x_{\max}]$, on peut traiter chacun des valeurs possibles comme une classe d'événement.

Si les valeurs de x sont entières, tel que $x \in [x_{\min}, x_{\max}]$ on peut estimer la probabilité a partir de M observations de la valeur, $\{X_m\}$.

Pour estimer la probabilité d'une valeur on peut compter le nombre d'observation de chaque valeur, x , dans une table, $h(x)$.

L'existence des ordinateurs avec des centaines de megabytes rendre des tables de fréquence très pratique pour la mise en œuvre en temps réel des algorithmes de reconnaissance. Dans certains domaines, comme l'analyse d'images, par abus de langage, un tel table s'appelle une histogramme. Proprement dit, l'histogramme est une représentation graphique de $h(x)$

Ainsi la probabilité d'une valeur de $X \in [X_{\min}, X_{\max}]$ est la fréquence d'occurrence de la valeur. Avec M observations de la valeur, X , on peut faire une table, $h(x)$, de fréquence pour chacun des valeurs possibles. On observe M exemples de X , $\{X_m\}$.

Pour chaque observation on ajoute "1" à son entré dans la table.

$$\forall m=1, M : h(X_m) := h(X_m) + 1; M := M+1;$$

$h(x)$ est une table de fréquence pour chaque $x \in [x_{\min}, x_{\max}]$.

Ainsi, on peut définir la probabilité d'une valeur x par sa fréquence :

$$p(X_m=x) \equiv \lim_{M \rightarrow \infty} \left\{ \frac{1}{M} h(x) \right\}$$

Quand M est fini, on peut faire appel à l'approximation.

$$P(X=x) \approx \frac{1}{M} h(x)$$

La validité de l'approximation depend du nombre de valeurs possible et de M . En règle générale, on dit qu'il faut 10 exemples par cellule de l'histogramme.

2.4 Bayes Rule

Soit un evennement "E". Soit deux tribus d'evennements A et B tel que certains evennements sont common à A et à B.

E peut appartenir à $A \cap B$ ou à $\neg A \cap B$ ou à $A \cap \neg B$ ou à $\neg A \cap \neg B$

Soit deux propositions p et q.

$$p \equiv E \in A \text{ et } q \equiv E \in B$$

donc $P(p) \equiv \Pr\{E \in A\}$ et $P(q) \equiv \Pr\{E \in B\}$.

Par axiome 2 de la définition des systèmes de probabilités :

$$P(q) + P(\neg q) = 1.$$

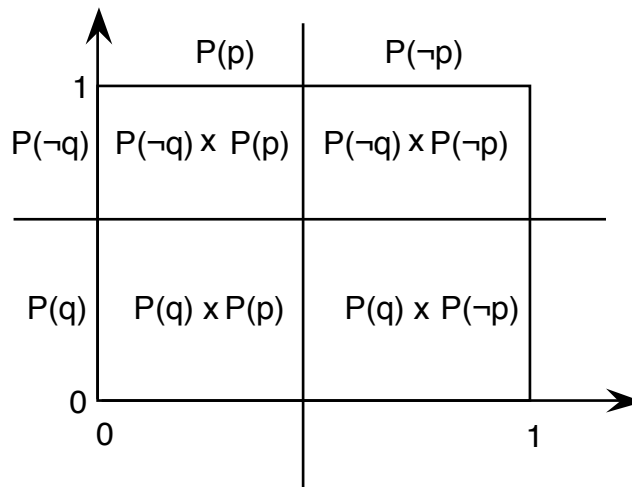
$P(p \wedge q)$ est la probabilité "conjointe" de p et q.

Si p et q sont indépendentes

$$P(p \wedge q) = P(p) \cdot P(q),$$

$$P(p \vee q) = P(p) + P(q).$$

On peut voir ça d'une manière graphique :



$$P(p \wedge q) + P(p \wedge \neg q) + P(\neg p \wedge q) + P(\neg p \wedge \neg q) = 1$$

Dans ce cas, les probabilités marginales sont

$$P(p) = P(p \wedge q) + P(p \wedge \neg q)$$

$$P(q) = P(p \wedge q) + P(\neg p \wedge q)$$

La probabilité conditionnelle de q étant donnée p s'écrit $P(q | p)$

$$P(q | p) = \frac{P(p \wedge q)}{P(p)} = \frac{P(p \wedge q)}{P(p \wedge q) + P(p \wedge \neg q)}$$

de la même manière :

$$P(p | q) = \frac{P(p \wedge q)}{P(q)} = \frac{P(p \wedge q)}{P(p \wedge q) + P(\neg p \wedge q)}$$

Par algèbre on déduit :

$$P(q | p) P(p) = P(p \wedge q) = P(p | q) P(q)$$

d'où

$$P(q | p) P(p) = P(p | q) P(q)$$

Ceci est une forme de règle de Bayes. On peut écrire :

$$P(q | p) = \frac{P(p | q) P(q)}{P(p)}$$

$P(q | p)$ est la probabilité "conditionnelle" ou "postérieur"

3 Classification by Ratio of Histograms of pixel values

3.1 Histograms

La probabilité d'une valeur x est sa fréquence d'occurrence.

$$p(X_m=x) \equiv \lim_{M \rightarrow \infty} \left\{ \frac{1}{M} h(x) \right\}$$

Quand M est fini, on peut faire appel à l'approximation.

$$P(X=x) \approx \frac{1}{M} h(x)$$

La validité de l'approximation dépend du ratio entre le nombre de Cellules, $Q = N^d$, de $h(x)$ et le nombre d'échantillons, M .

L'erreur moyenne entre $\frac{1}{M} h(\vec{C})$ et $P(\vec{C})$ est $E_{ms} \sim O\left(\frac{Q}{M}\right)$

Pour que l'estimation soit "raisonnable", il faut assuré que $M \gg Q = N^d$
En règle générale, on dit qu'il faut 10 exemples par cellule de l'histogramme.

3.2 Example: Object detection by pigment color

On peut utiliser les histogrammes avec la règle de Bayes pour détecter les objets.

Par exemple, construisons une histogramme pour le vecteur de chrominance (r,v) .

La chrominance $C = \begin{pmatrix} c_1 \\ c_2 \end{pmatrix}$ est une signature pour l'objet.

La chrominance peut être définie par plusieurs codages. Par exemple, pour la détection du peau, il est fréquent de voir

$$c_1 = r = \frac{R}{R+V+B} \quad c_2 = v = \frac{V}{R+V+B}$$

Supposons qu'on code c_1 et c_2 avec les entiers entre 0 et $N-1$

$$c_1 = \text{Round} \left((N-1) \cdot \frac{R}{R+G+B} \right) \quad c_2 = \text{Round} \left((N-1) \cdot \frac{G}{R+G+B} \right)$$

On alloue un tableau 2D, $h(c_1, c_2)$, de taille $N \times N$ cellules.
(exemple $Q = 32 \times 32 = 1024$ cellules)

Pour chaque pixel $\vec{C} = C(i, j)$ dans l'image, on incrémente la cellule de l'histogramme qui correspond à \vec{C} : $h(\vec{C}) := h(\vec{C}) + 1$
c'a-dire $h(c_1, c_2) := h(c_1, c_2) + 1$

Soit M Pixels dans l'image. Un histogramme des chrominance, $h(\vec{C})$, des M pixels dans une l'image donne leurs fréquences d'occurrence.

$$P(\vec{C}) \approx \frac{1}{M} h(\vec{C})$$

Considère une région W de M_0 pixels du même image correspondance à l'objet O .

$$\forall (i,j) \in W : h_o(\vec{C}(i,j)) := h_o(\vec{C}(i,j)) + 1$$

Ensuite: pour tout pixel $\vec{C}(i, j) = \begin{pmatrix} r \\ v \end{pmatrix}(i, j) : p(\vec{C} | \text{objet}) \approx \frac{1}{M_0} h_o(\vec{C})$

Parce que W est dans l'image, la probabilité de rencontrer un pixel de W ,

$$P(W) = \frac{M_0}{M}$$

L 'histogramme permet d'utiliser la règle de Bayes afin de calculer la probabilité qu'un pixel corresponde à un objet.

Pour chaque pixel $\vec{C}(i, j) : p(\text{objet} | C) = p(C | \text{objet}) \frac{p(\text{objet})}{p(C)}$

Soit M images de $I \times J$ pixels. Ceci fait $N = I \times J \times M$ Pixels.

Soit $h(r, v)$, l'histogramme de tous les N pixels.

Soit $h_o(r, v)$, l'histogramme des N_0 pixels de l'objet "o".

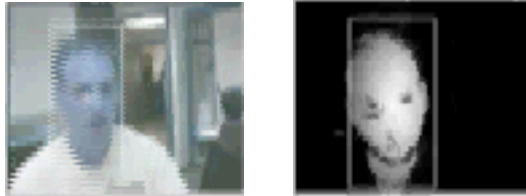
$$p(\text{objet}) = \frac{M_0}{M}$$

$$p(\vec{C}) = \frac{1}{M} h(\vec{C})$$

$$p(\vec{C} | \text{objet}) = \frac{1}{M_0} h_o(\vec{C})$$

$$p(\text{objet} | \vec{C}) = p(\vec{C} | \text{objet}) \frac{p(\text{objet})}{p(\vec{C})} = \frac{1}{M_0} h_0(\vec{C}) \frac{\frac{M_0}{M}}{\frac{1}{M} h(\vec{C})} = \frac{h_0(\vec{C})}{h(\vec{C})}$$

Voici une image de la probabilité de peau fait par ratio d'histogramme de r, v



3.3 Histograms of Receptive Field Values

Cette méthode peut être généralisé en remplaçant la chrominance par un vecteur de champs réceptifs. Mais il faut bien gerer la ration Q/M !

Soit une image $p(i,j)$ et un vecteur de "d" champs réceptifs \vec{G}

$\vec{V}(i,j) = \langle \vec{G}, p(i,j) \rangle$ est un vecteur de caractéristiques de d dimensions.

$h(\vec{V})$ aura $Q = N^d$

N \ d	1	2	3	4	5	6
2	2^1	2^2	2^3	2^4	2^5	2^6
4	2^2	2^4	2^6	2^8	$2^{10} = 1 \text{ Kilo}$	$2^{12} = 2 \text{ K}$
8	2^3	2^6	2^9	2^{12}	2^{15}	2^{18}
16	2^4	2^8	2^{12}	2^{16}	$2^{20} = 1 \text{ Meg}$	$2^{24} = 4 \text{ M}$
32	2^5	$2^{10} = 1 \text{ Kilo}$	2^{15}	$2^{20} = 1 \text{ Meg}$	2^{25}	$2^{30} = 1 \text{ G}$
64	2^6	2^{12}	2^{18}	2^{24}	$2^{30} = 1 \text{ Gig}$	2^{36}
128	2^7	2^{14}	$2^{21} = 2 \text{ Meg}$	2^{28}	2^{35}	$2^{42} = 2 \text{ T}$
256	2^8	2^{16}	2^{24}	$2^{32} = 2 \text{ Gig}$	$2^{40} = 1 \text{ Tera}$	2^{48}

Soit les champs receptifs achromatique

$$\vec{G}_\sigma = (G_x, G_y, G_{xx}, G_{xy}, G_{yy})$$

$$d = 5$$

ou chromatique avec normalisation de l'orientation et echelle :

$$\vec{G}_{\sigma,\theta} = (G_x^L, G_x^{C1}, G_x^{C2}, G_{xx}^L, G_{xy}^L, G_{xx}^{C1}, G_{xx}^{C2})$$

$$d = 7.$$

On peut faire

$$p(\text{objet}(i,j) | \vec{V}(i,j)) = \frac{p(\vec{V}(i,j) | \text{objet}(i,j)) p(\text{objet}(i,j))}{p(\vec{V}(i,j))} \approx \frac{h_o(\vec{V}(i,j))}{h_{\text{tot}}(\vec{V}(i,j))}$$

sur condition de gerer M et Q.

Rappel :

$$P(X=x) \approx \frac{1}{M} h(x)$$

La validité de l'approximation dépend du nombre de valeurs possible et de M .
En règle générale, on dit qu'il faut 10 exemples par cellule de l'histogramme.

On peut démontrer que l'écart type moyenne de l'erreur est en proportion avec la
ration de le nombre de cellule de $h(x)$, N , sur le nombre d'échantillons, M .

$$\sigma_{\text{MSE}} \approx \frac{N}{M}$$

Que faire si la masse d'exemple est insuffisante : $M \ll N$?

Que faire si x n'est pas entier ? Il faut une fonction paramétrique pour $p(X)$.