Master in
Computer Vision
Barcelona

UAB UOC UPC upf

**Module: M2. Optimization and inference techniques for Computer Vision** **Final exam**
Date: December 3rd, 2015
Teachers: Juan Fco Garamendi, Coloma Ballester, Oriol Ramos, Joan Serra **Time: 2h30min**

- Books, lecture notes, calculators, phones, etc. are not allowed.
- All sheets of paper should have your name.
- Answer each problem in a separate sheet of paper.
- All results should be demonstrated or justified.

## Problem 1

*Juan F. Garamendi, 0.5 Points*

Let

$$\begin{aligned} J: \ & \mathcal{V} && \to \mathbb{R}, \\ & u && \mapsto J(u) = \int_{\Omega} \mathcal{F}\left(\mathbf{x}, u(\mathbf{x}), \nabla u(\mathbf{x})\right) d\mathbf{x} \end{aligned}$$

be a convex energy functional over functions $u$, where

- $\mathcal{V}$ is a suitable space of functions.
- $\Omega \in \mathbb{R}^d$ is a bounded open domain of the $d$ dimensional euclidean space $\mathbb{R}^d$.
- $u \in \mathcal{V}$, $u : \Omega \to \mathbb{R}$ is a scalar function defined on $\Omega$.
- $\mathbf{x} \in \Omega$ such that $\mathbf{x} = (x_1, .., x_d)$ is the spatial variable and $\nabla$ is the gradient operator such that $\nabla u(\mathbf{x}) = (u_{x_1}, .., u_{x_d})$

(a) (0.25 points) Say in few words which is the fundamental problem in calculus of variations.

The fundamental problem of the calculus of variations is to find the extremum (maximum or minimum) of the functional $J(u)$ with respect to $u$.

(b) (0.25 points) Say in a few words what are the following expresions:

(i)

$$-\text{div}_{\mathbf{x}}\left(\nabla_{\bar{g}} \mathcal{F}\right) + \frac{\partial \mathcal{F}}{\partial u} = 0$$

where

- The divergence $\text{div}_{\mathbf{x}}$ is computed with respect to variable $\mathbf{x}$
- The gradient $\nabla_{\bar{g}}$ is computed with respect to the components of $\bar{g} = \nabla u(\mathbf{x})$

(ii)

$$-\sum_{i=1}^{d} \frac{\partial}{\partial x_i} \frac{\partial \mathcal{F}}{\partial u_{x_i}} + \frac{\partial \mathcal{F}}{\partial u} = 0$$

Both are the canonical (general) expresion for the Euler-Lagrange equation and they are the necessary condition for the extremality of $J$.

**Problem 2** *Juan F. Garamendi, 1.5 Points*

Let $I_0 : \Omega \to \mathbb{R}$ and $I_1 : \Omega \to \mathbb{R}$ be two given (probably noisy) images, where $\Omega$ is a bounded open subset of $\mathbb{R}^2$ and $I_0, I_1 \in L^\infty(\Omega)$. Consider the following minimization problems

$$\arg\min_{\mathbf{u}} \left\{ \int_\Omega |\nabla u_1|^2 + |\nabla u_2|^2 d\mathbf{x} + \lambda \int_\Omega \left(I_1(\mathbf{x} + \mathbf{u}(\mathbf{x})) - I_0(\mathbf{x})\right)^2 d\mathbf{x} \right\}$$

Where

- $u_i \in W^{1,2}(\Omega)$.
- $W^{1,2}(\Omega) = \{u \in L^2(\Omega);\ \nabla u \in L^2(\Omega)^2\}$.
- $\mathbf{u} : \Omega \to \mathbb{R}^2$ such that $\mathbf{u} = (u_1(\mathbf{x}), u_2(\mathbf{x}))^T$ is a vector field.
- $|\cdot|$ is the usual Euclidean norm.
- $\lambda \in \mathbb{R}^+$ is a given parameter.

(a) (0.5 points) Describe in a few words what image problem solves the given minimization problem. The correspondence problem between the two images. Generally speaking, $\mathbf{u}$ registers the pixels of the image $I_0$ onto the pixels of the image $I_1$. Actually, this is the Horn-Schunk model for the optical flow estimation.

(b) (1 point) This energy functional can be locally linearized using a first order Taylor approximation of the data fidelity term:

$$\arg\min_{\mathbf{u}} \left\{ \int_\Omega |\nabla u_1|^2 + |\nabla u_2|^2 d\mathbf{x} + \lambda \int_\Omega \left(\langle \nabla I_0, \mathbf{u}\rangle + I_1 - I_0\right)^2 d\mathbf{x} \right\} \tag{1}$$

where the product $\langle \cdot, \cdot \rangle$ is the Euclidean scalar product.

Let $\Delta = \frac{\partial^2}{\partial x_1^2} + \frac{\partial^2}{\partial x_2^2}$ be the Laplace operator. Prove that

$$\lambda \frac{\partial I_0}{\partial x_1}\left(u_1 \frac{\partial I_0}{\partial x_1} + u_2 \frac{\partial I_0}{\partial x_2} + I_1 - I_0\right) - \Delta u_1 \quad = \quad 0$$

is a necessary condition for the minimization problem (1) with respect to $u_1$.

The minimization problem can be written as

$$\arg\min_{\mathbf{u}} \left\{ \int_\Omega |\nabla u_1|^2 + |\nabla u_2|^2 d\mathbf{x} + \lambda \int_\Omega \left(u_1 \frac{\partial I_0}{\partial x_1} + u_2 \frac{\partial I_0}{\partial x_2} + I_1 - I_0\right)^2 d\mathbf{x} \right\}$$

Following the canonical expression for the Euler-Lagrange equation given in the previous problem

$$-\sum_{i=1}^{d} \frac{\partial}{\partial x_i} \frac{\partial \mathcal{F}}{\partial u_{1_{x_i}}} + \frac{\partial \mathcal{F}}{\partial u_1} = 0$$

and expanding the sumatory

$$-\left(\frac{\partial}{\partial x_1} \frac{\partial \mathcal{F}}{\partial u_{1_{x_1}}} + \frac{\partial}{\partial x_2} \frac{\partial \mathcal{F}}{\partial u_{1_{x_2}}}\right) + \frac{\partial \mathcal{F}}{\partial u_1} = 0$$

and taking into account that $\mathcal{F} = |\nabla u_1|^2 + |\nabla u_2|^2 + \lambda\left[\left(u_1\frac{\partial I_0}{\partial x_1} + u_2\frac{\partial I_0}{\partial x_2} + I_1 - I_0\right)^2\right]$ we have that

$$\frac{\partial \mathcal{F}}{\partial u_1} = 2\lambda\left(u_1\frac{\partial I_0}{\partial x_1} + u_2\frac{\partial I_0}{\partial x_2} + I_1 - I_0\right)\frac{\partial I_0}{\partial x_1}$$

$$\frac{\partial \mathcal{F}}{\partial u_{1_{x_1}}} = 2u_{1_{x_1}}$$

$$\frac{\partial \mathcal{F}}{\partial u_{1_{x_2}}} = 2u_{1_{x_2}}$$

$$\frac{\partial}{\partial x_1}2u_{1_{x_1}} + \frac{\partial}{\partial x_1}2u_{1_{x_2}} = 2\left[\frac{\partial^2}{\partial x_1^2}u_1 + \frac{\partial^2}{\partial x_1^2}u_1\right] = 2\Delta u_1$$

So,

$$-2\Delta u_1 + 2\lambda\frac{\partial I_0}{\partial x_1}\left(u_1\frac{\partial I_0}{\partial x_1} + u_2\frac{\partial I_0}{\partial x_2} + I_1 - I_0\right) = 0$$

$$\lambda\frac{\partial I_0}{\partial x_1}\left(u_1\frac{\partial I_0}{\partial x_1} + u_2\frac{\partial I_0}{\partial x_2} + I_1 - I_0\right) - \Delta u_1 = 0$$

## Problem 3 — *Juan F. Garamendi, 0.5 Points*

Consider the following iterative scheme

- $\bar{x}^0 = \bar{0}$
- $\bar{x}^{k+1} \leftarrow S(\bar{x}^k, \bar{x}^{k+1}, \bar{b})$

applied to the algebraic problem $\mathbf{A}\bar{x} = \bar{b}$, where $\mathbf{A}$ is a known matrix, $\bar{b}$ is a known vector, $\bar{x}$ is an unknown vector , $S$ some function and super-index represents iteration number.

(a) (0.5 points) Which is the most fundamental property that $S$ must have for being a smoother component in the Multigrid context. In a few words explain why. As the name says, $S$ must have a remarkable smoothing effect on the error $e = \bar{x} - \bar{x}^k$ of an approximation $\bar{x}^k$. This is because the residual equation $\mathbf{A}\bar{e} = \bar{r}$, being $\bar{r} = \bar{b} - \mathbf{A}\bar{x}^k$ the residual, will be solved in a coarser level than the original algebraic problem $\mathbf{A}\bar{x} = \bar{b}$.

## Problem 4 — Coloma Ballester *1. Points*

Consider the function $f : R^n \to R$ defined by

$$f(x) = \frac{1}{2}\langle Ax, x\rangle - \langle x, b\rangle + c,$$

for $x \in R^n$, where $A$ is a $n \times n$ symmetric matrix, $n \in N$, $b \in R^n$ and $c \in R$. (The notation $\langle\cdot,\cdot\rangle$ stands for the scalar product.)

(a) What codition on the matrix $A$ implies that $f$ is a convex function?

The answer is $A$ positive definite (all eigenvalues of $A$ are positive). Indeed, we know that, for a differentiable function, $f$ is convex if and only if all eigenvalues of its Hessian $D^2f(x)$ are positive for all $x$. In our case, $D^2f(x_0) = A$.

(b) Let $v \in R^n, v \neq 0$. Compute $D_v f(x)$, the directional derivative of $f$ at the point $x$ in the direction $v$.

Given $v \in R^n, v \neq 0$, and $x \in R^n$:

$$D_v f(x) = \lim_{\epsilon \to 0+} \frac{f(x+\epsilon v) - f(x)}{\epsilon} = \lim_{\epsilon \to 0+} \frac{\frac{1}{2}\langle A(x+\epsilon v), x+\epsilon v\rangle - \langle x+\epsilon v, b\rangle + c - \left(\frac{1}{2}\langle Ax, x\rangle - \langle x, b\rangle + c\right)}{\epsilon} =$$

$$= \lim_{\epsilon \to 0+} \frac{\frac{\epsilon}{2}\langle Ax, v\rangle + \frac{\epsilon}{2}\langle Av, x\rangle + \frac{\epsilon^2}{2}\langle Av, v\rangle - \epsilon\langle v, b\rangle}{\epsilon} = \frac{1}{2}\langle Ax, v\rangle + \frac{1}{2}\langle A^t x, v\rangle - \epsilon\langle v, b\rangle = \langle Ax - b, v\rangle.$$

(c) Use the result in (a) to compute $\nabla f(x)$. Which is the equation satisfied by a minimum of $f(x)$? (it is called the Euler-Lagrange equation).

We know that

$$D_v f(x_0) = \langle \nabla f(x), v\rangle.$$

for all $v \in R^n$. We can extract from this the vector $\nabla f(x)$:

$$\nabla f(x) = Ax - b.$$

The equations satisfied by the minimum are

$$\nabla f(x) = 0.$$

In our case, $Ax = b$.

(d) Give a codition on the matrix $A$ implying that a solution $x_0$ of the Euler-Lagrange equation is a unique minimum of $f(x)$.

The answer is $A$ strictly positive definite. Two possible explanations:
(1) As $D^2 f(x_0) = A$, we know that if $A$ strictly positive definite implies that $f$ is strictly convex, and we know that strictly convex functions on (closed) convex sets ($R^n$ in our case) have a unique minimum.
(2) (using the second order sufficient conditions): For a given function $f(x) : \mathbb{R}^n \to \mathbb{R}$, if $\nabla f(x_0) = 0$ and $D^2 f(x_0)$ is positive definite (respectively strictly positive definite), then $x_0$ is a local minimum (respectively, a unique minimum).
In our case, $D^2 f(x_0) = A$, therefore the condition is $A$ strictly positive definite.

## Problem 5 — Coloma Ballester *0.75 Points*

Let $A$ be a $m \times n$ matrix, and $b \in R^m$. Consider the following constrained optimization problem (P) defined as

$$\begin{array}{l} \min \frac{1}{2}\langle x, x\rangle \\ \text{subject to } Ax = b. \end{array}$$

(a) Write problem (P) as a min-max problem and define the duality gap.

$Ax = b$ gives $m$ equality constraints on $x$: $(Ax)_i = b_i$, $i = 1, \cdots, m$. Therefore, we introduce $m$ Lagrange multipliers (or dual variables), $\nu_1, \ldots, \nu_m$, and we construct the Lagrangian function $\mathcal{L}(x,\nu) = f(x) - \sum_{i=1}^{m} \nu_i((Ax)_i - b_i) = \frac{1}{2}\langle x, x\rangle - \langle \nu, Ax - b\rangle = \frac{1}{2}\langle x, x\rangle - \langle A^t\nu, x\rangle + \langle \nu, b\rangle$ , where $\nu = (\nu_1, \ldots, \nu_m)^t \in R^m$. Therefore

$$\min_{\text{subject to } Ax=b} \frac{1}{2}\langle x, x\rangle = \min_{x \in R^n} \max_{\nu \in R^m} \mathcal{L}(x, \nu)$$

The duality gap is the difference

$$DG = \min_{x \in R^n} \max_{\nu \in R^m} \mathcal{L}(x,\nu) - \max_{\nu \in R^m} \min_{x \in R^n} \mathcal{L}(x,\nu).$$

(b) Define and compute the dual function of problem (P).

In our case, $DG = 0$ because $\mathcal{L}(x,\nu)$ is convex on $x$ ($f$ is convex and $d_i(x) = (Ax)_i - b_i$ are linear constraints) and $\mathcal{L}(x,\nu)$ is concave on $\nu$. Therefore we can change min-max by max-min:

$$\min_{\text{subject to } Ax=b} \frac{1}{2}\langle x,x\rangle = \min_{x\in R^n}\max_{\nu\in R^m} \mathcal{L}(x,\nu) = \max_{\nu\in R^m}\min_{x\in R^n} \mathcal{L}(x,\nu) = \max_{\nu\in R^m} g_D(\nu),$$

where $g_D(\nu) = \min_{x\in R^n} \mathcal{L}(x,\nu)$ is the dual function. We compute the dual function from $\nabla_x \mathcal{L}(x,\nu) = 0$ (indeed, $\mathcal{L}(x,\nu)$ is a quadratic function of $x$, and for each $\nu$ there is a unique minimizer $x_0(\nu)$. The minimizer is the solution of $\nabla_x\mathcal{L}(x,\nu) = 0$). In our case, $x - A^t\nu = 0$, which gives $x_0(\nu) = A^t\nu$. Then,

$$g_D(\nu) = \mathcal{L}(x_0(\nu),\nu) = -\frac{1}{2}\langle A^t\nu, A^t\nu\rangle + \langle \nu, b\rangle$$

(c) Write down the dual problem.

$\max_{\nu\in R^m} (-\frac{1}{2}\langle A^t\nu, A^t\nu\rangle + \langle\nu,b\rangle)$.

## Problem 6

Coloma Ballester *0.75 Points*

Working on a data fitting (or regression) problem where we were interested in fitting a function to a given set of data, we have transformed our data fitting problem to the following least squares problem:

$$\min_x \|A\mathbf{x} - \mathbf{b}\|^2 \tag{2}$$

where $A$ and $\mathbf{b}$ are a fixed matrix and vector, respectively, obtained from the given data, and $\mathbf{x}$ is a vector of unknowns.

(a) Write down the normal equations associated to this problem.

The normal equations associated to the least squares problem (2) are $A^tA\mathbf{x} = A^t\mathbf{b}$.

(b) How could you determine the solution $\mathbf{x}$ using the SVD or the pseudoinverse of the matrix associated to your problem? (Recall that SVD stands for Singular Value Decomposition of a matrix).

The solution of the normal equations with minimum norm is $\bar{\mathbf{x}} = A^+\mathbf{b}$ where $A^+$ is the pseudoinverse of $A$. To compute the pseudoinverse, we use the SVD decomposition of $A$, which in turn is:

$$A = U\Sigma V^t$$

where $U$ is a $N\times N$ orthogonal matrix, $V$ is a $M\times M$ (where $M$ is the number of unknowns in $\mathbf{x}\in\mathbb{R}^M$) orthogonal matrix, and $\Sigma$ is a rectangular $N\times M$ diagonal matrix. The diagonal values of $\Sigma$, $\sigma_i > 0$, for $i = 1,\ldots,r$ are the sigular values. The number of singular values $r$ is the rank of $A$ (thus, $r \leqslant \min\{N,M\}$).

From the SVD of $A$, we compute its pseudoinverse:

$$A^+ = V(\Sigma^t)^+U^t$$

where the pseudo-inverse $(\Sigma^t)^+$ is a $M\times N$ diagonal matrix with entries

$$\sigma_i^+ = \begin{cases} \frac{1}{\sigma_i} & \text{if } \sigma_i \neq 0 \\ 0 & \text{if } \sigma_i = 0, \end{cases}$$

for $i = 1,\ldots,M$.

## Problem 7 J.Serrat *0.5 Points*

The starting point for the probabilistic learning of the parameters of a conditional random field

$$\begin{aligned} p(y|x,w) &= \frac{1}{Z(x,w)} \exp[-E(x,y,w)] \\ E(x,y,w) &= \langle\, w, \psi(x,y) \,\rangle \\ Z(x,y) &= \sum_{y \in \mathcal{Y}} \exp[-E(x,y,w)] \\ y^\star &= \arg\max_{y \in \mathcal{Y}} \, \langle\, w, \psi(x,y) \,\rangle \end{aligned}$$

was to maximize the conditional likelihood, from which we arrived to

$$w^\star = \arg\min_w \sum_{i=1}^N \langle\, w, \psi(x^i, y^i) \,\rangle + \sum_{i=1}^N \log Z(x^i, w)$$

Interpret this formula, that is, tell what is (need to correctly answer all of them):

(a) the number of samples in the training set

(b) the training set itself

(c) the partition function

(d) the model parameters

(a) $N$

(b) $(x^i, y^i), i = 1 \dots N$

(c) $Z(x, w)$

(d) $w$

## Problem 8 J.Serrat *0.5 Points*

Which was the solution to the three main problems found when trying to optimize the fomer expression by gradient descent? Answer writing the pairs of problem-solution labels, like 1-a, 2-b, 3-c.
Problems:

(1) $Z(x^i, w)$ or $\mathbb{E}_{y \sim p(y|x^i,w)} \psi(x^i, y)$ impossible to calculate in practice

(2) $N$ large and therefore we have to run belief propagation $N$ times

(3) $N$ small compared to number of parameters, causing overfitting

Solutions:

(a) regularization, assuming $w$ follows a Gaussian distribution

(b) since $\psi(x, y)$ decomposes in factors, we can apply some inference method like belief propagation to compute it

(c) perform stochastic gradient descent

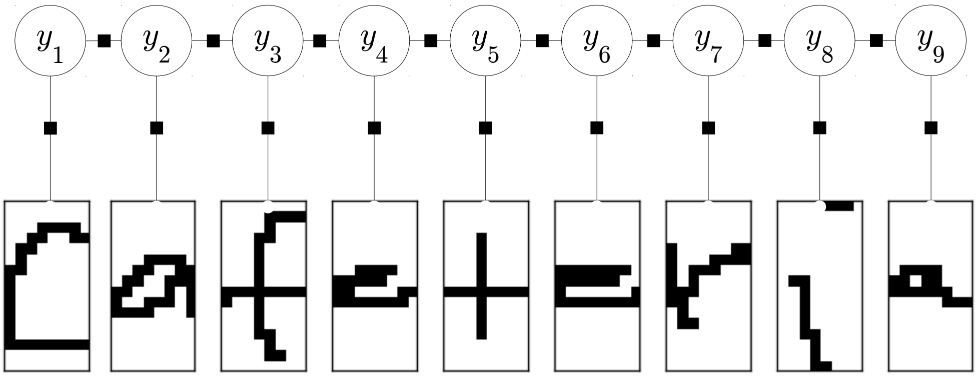


Figura 1

1-b, 2-c, 3-a

**Problem 9** J.Serrat *0.5 Points*

Consider the graphical model of figure 1 where observations are binary images $16 \times 8 = 128$ pixels, that is, $x_i \in \{0,1\}^{128}$, $y_i \in Y = \{\mathrm{a}, \mathrm{b} \ldots \mathrm{z}\}$ (26 lowercase letters), $x = (x_1 \ldots x_9), y = (y_1 \ldots y_9)$.

We want to learn $w$ to later infer a word from a series of binary images of letters as

$$\begin{aligned} y^\star &= \arg\max_{y \in \mathcal{Y}} \langle w, \psi(x,y) \rangle \\ &= \arg\max_{y \in Y^9} \sum_{i=1}^{9} \sum_{p=\mathrm{a}}^{\mathrm{z}} \sum_{j=1}^{16} \sum_{k=1}^{8} w_{pjk}\, x_{ijk} + \sum_{i=1}^{8} \sum_{p=\mathrm{a}}^{\mathrm{z}} \sum_{q=\mathrm{a}}^{\mathrm{z}} w_{pq}\, \mathbf{1}_{y_i=\mathrm{p},\, y_{i+1}=\mathrm{q}} \end{aligned}$$

where $\mathbf{1}_{y_i=\mathrm{p},\, y_{i+1}=\mathrm{q}}$ evaluates to 1 if $y_i = \mathrm{p}$ and $y_{i+1} = \mathrm{q}$. In this context,

(a) what's the total number of parameters to learn ? (no need to write the final number, just an expression like $12\times34 + 56^7$ is ok)

(b) what does $w_{p=\mathrm{a},q=\mathrm{b}}$ mean or represent ?

(a) $16\times8\times26 + 26^2$, (b) compatibility of 'b' right after 'a', high if this is a frequent pair in our training set

**Problem 10** J.Serrat *0.5 Points*

With regard to the problem of question

(a) what do you think the image of figure 2 is, I mean, which specific $w_{pq}$ or $w_p$ ?

(b) what's the total number of unary coefficients to learn if we apply the two-stage training technique? (again, an arithmetic expression is ok)

(a) $w_{p=s}$ unary coefficients for letter 's', (b) just 26 instead of former $16\times8\times26$

**Problem 11** J.Serrat *0.5 Points*

In the exercise of labeling segments of a jacket contour we proposed the model of figure 3 (again a chain). What's the advantage of labeling by inference on this model, that is, to do structured prediction, over the simpler approach of classifying each segment independently with, say, a multiclass SVM ?

We can impose constraints, or weight, each successive pair of labels. A classifier like SVM would just tell us which is the label (letter) for each observed binary image, but can not prevent wrong pairs

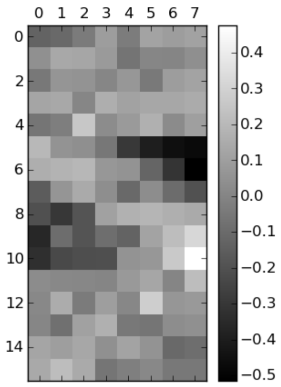


Figura 2

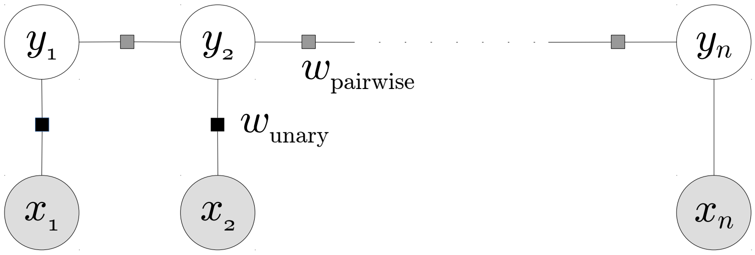


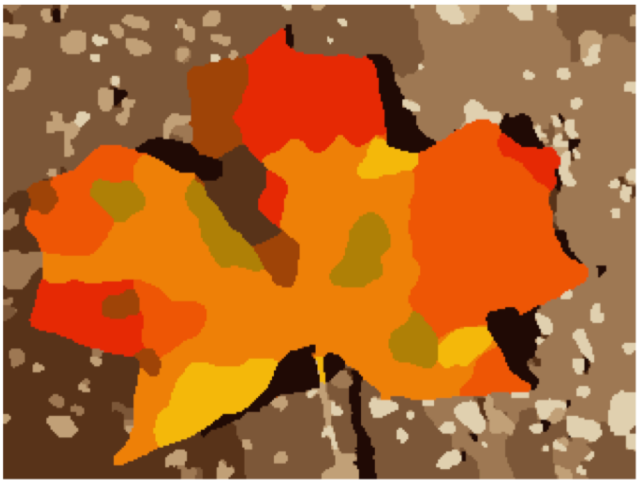
Figura 3

of successive labels. Instead, in structured prediction we learn that certain pairs are more frequent than others and this influences the final labeling, not only the observations count. The pairwise coefficients act as a kind of prior on the solution.

**Problem 12** *Oriol Ramos Terrades, 2.5 Points*

Consider the segmentation problem seen in the lectures and illustrated with the following images:

(a) Original Image



(b) Segmented image by color

The goal is, given color image such us for instance the one shown in (a), to segment it into regions of similar colors. Assume that a palette of $K$ colors has already learned and $\mu_k$ represents the $k$-th color in the RGB space.

a) This problem can be modeled as a conditional random field (CRF). Which are the hidden (or latent) variables? and the observed variables? Write the domain of both kind of variables (0.5 point).

**Solution:** We have to kinds of random variables:

- latent variables: We denote them by $y_i$. We have one for each pixel $i$. Its domain is $\{0, \ldots, K-1\}$

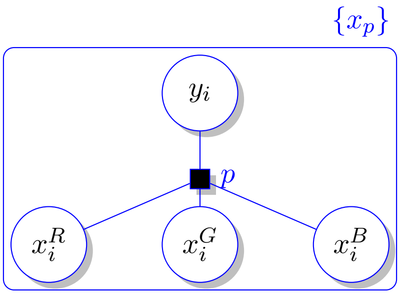
- Observed variables: For each pixel, we have one for each color channel. We denote them by a random vector: $x_i = (x_i^R, x_i^G, x_i^B)$. The domain is $\{0, \dots, 255\}$ for all of them

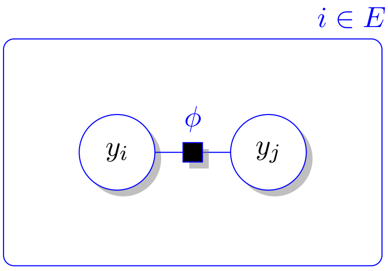
b) Draw the factor graph that models this problem and write the associated joint distribution in terms of factor functions (0.5 point).

**Solution:** The joint distribution factorizes:

$$p(y|x) = \frac{1}{Z} \prod_i p(y_i | x_i^R, x_i^G, x_i^B) \prod_{(i,j) \in E} \phi(y_i, y_j) \tag{3}$$

where $E$ is the set of adjacent pixels given a 4-connectivity scheme. The factor graph is:





c) We define the factor function, $\phi(y_i, y_j)$, that models the interactions between hidden variables, by a Potts model of parameter $\theta$. Write the matrix that represents this factor function (0.5 point).

**solution:** We can represent $\phi$ as a square symmetric matrix of dimension $K$. if we denote by $M = e^\theta$, it has the following shape:

$$\phi(y_i, y_j) = \begin{pmatrix} M & \cdots & 1 \\ \vdots & \ddots & \vdots \\ 1 & \cdots & M \end{pmatrix} \tag{4}$$

d) If we replace the Potts model by the following feature function:

$$f(y_i, y_j) = \sum_{s,t=1}^{K} \theta_{s,t} 1_{\{y_i = s\}}(y_i) 1_{\{y_j = t\}}(y_j),$$

which is the matrix that represents this new factor function (0.5 point)?

**solution:** $\phi$ is also as a square symmetric matrix of dimension $K$. The entry of this matrix at row $s$ and column $t$ is $= e^{\theta_{s,t}}$.

e) Given the kind of problem that we want to solve, which are the most suitable inference algorithms that we can apply to solve it (0.5 point)?

**solution:** We have to find the **most probable state** of hidden variables $y_1, \dots,$ given the observations $x_i$. This implies to solve the problem:

$$\hat{y} = \operatorname*{argmax}_y p(y|x) \tag{5}$$

To solve this kind of problems we can apply the max-sum algorithm, Graph Cuts or Linear Programing Relaxation based algorithms.