

“73”

Marie Moin / Olivier Rodot / Thierry Géraud

Projet de “droit / maths / info” 2003

Intentions

Lorsque l’on teste la vision des couleurs d’une personne, on utilise des images telles que celle de la figure 1.

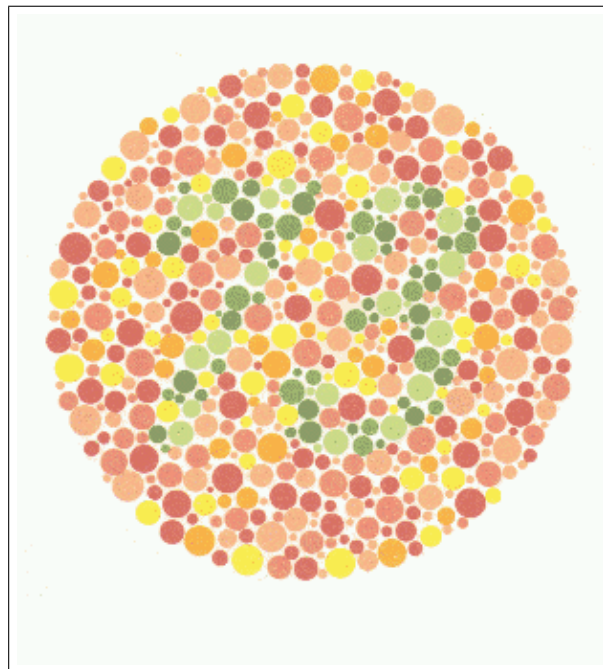


Figure 1: Une image de test de vision (y).

Pour des informations sur les tests de vision, allez butiner sur les sites
http://www.geocities.com/alfredo_71/vidaurri_5.html et
http://www.csun.edu/~vceed002/biology/senses/vision_labs/vision.htm

Le but de ce projet est que l’ordinateur passe le test à notre place et reconnaisse dans l’image le “73”.

1 Détails pratiques

1.1 Notation et groupe de projet

Ce projet donne lieu à deux notes séparées : une en “droit” (Cf. partie 3) et une en “maths/info”.

Pour ce qui est de la partie maths/info, vous serez *autant* évalué sur le contenu mathématique de vos réponses (la compréhension du support mathématique de ce projet) que sur la partie informatique (la mise en œuvre du projet). Le barème sera le suivant :

- réponses aux questions de maths (Cf. section 4) = 10 points ;
- propreté / lisibilité du code = 2 points ;
- modularité du projet (pertinence du découpage en fichiers + `Makefile` + etc.) = 3 points ;
- gestion de projet (pertinence de la répartition des tâches et de l’avancement du projet) = 3 points ;
- adéquation “cahier des charges / résultat” (le minimum attendu est que le projet compile et que l’exécution aboutisse à un résultat qui va dans le sens de la solution attendu) = 2 points.

Comme vous l’aurez compris, si vous n’obtenez pas de résultat probant, ce n’est pas important pour ce projet (la partie “adéquation” n’est notée que sur 2 points). Sachez que la moulinette qui va corriger votre projet ne vous autorise que *cinq* minutes de calcul sur une machine moderne ; au bout de ces 5 minutes, si votre programme tourne encore (ou a planté avant de sauvegarder un résultat !), vous êtes notés sur 18 (la note “adéquation” est nulle).

Ce projet est à réaliser en trinôme. Une note collective sera attribuée qui sera éventuellement réajustée par individu. Chaque personne du groupe doit avoir compris le projet, c’est-à-dire la partie mathématique *et* la partie mise en œuvre. S’il s’avère qu’un membre du groupe est à la ramasse, sa note sera individuelle.

Enfin, le délit de pompage sera sanctionné d’une note négative pour tous les membres du groupe fautif (mise en garde : la pompe des réponses aux questions de maths ou du code est très facilement repérable pour ce projet...)

1.2 À suivre

Les informations et questions relatives à ce projet seront traitées dans les forums de discussion suivants :

- `epita.comp.lang.c++`
- `epita.projet.maths`
- `epita.cours.modelisation`

Je ne saurais que trop vous conseiller de lire impérativement ces forums. En particulier, les modalités de rendu y seront exposées en temps et en heure.

2 Préambule

2.1 Image et variables aléatoires

Un point de l'image est tout simplement repéré par un indice que l'on notera i par la suite (c'est une simplification pour ne pas se traîner deux indices ligne/colonne).

On va modéliser notre problème sous une forme *probabiliste*. D'autres approches peuvent être envisagées mais elles seraient hors sujet (en clair, inutile de faire preuve d'imagination dans le hors-piste, cela ne vous rapporterait aucun point).

Soit une image ayant n points. On va considérer qu'un point est une *variable aléatoire*, c'est-à-dire une variable (de notre problème) dont la valeur suit une loi de probabilité. On notera Z_i cette variable aléatoire (ce point) et z_i une valeur possible de cette variable (une valeur possible de ce point). Une image z est alors considérée comme étant une réalisation (c'est le terme savant pour dire un "tirage") d'une loi Z définie par l'ensemble des variables aléatoires. On notera $z = \{z_1, \dots, z_n\}$ pour dire qu'une image est un ensemble de points où chaque point a une valeur (le point d'indice i a la valeur z_i). On notera $Z = \{Z_1, \dots, Z_n\}$ pour dire que la loi de probabilité qui décrit notre image est l'ensemble des variables aléatoires. Dire que Z a pour réalisation z se note $Z = z$. Dit autrement, notre image z est une réalisation de Z (z est un tirage suivant Z).

Ce qui va nous intéresser, ce sont les lois de probabilité $P(Z_i = z_i)$ (en français : les lois de probabilité P telle que Z_i ait pour valeur z_i).

Prenons un premier exemple, très simple. Supposons que Z soit une image binaire c'est-à-dire une image dont les points ont pour valeur "noir" ou "blanc" ($z_i \in \{\text{noir}, \text{blanc}\}$). Supposons de plus que Z soit constituée de deux points. On a alors $Z = \{Z_1, Z_2\}$ (cette image est un ensemble de deux points décrits respectivement par deux variables aléatoires). Supposons que la valeur du premier point suive la loi de proba :

$$P(Z_1 = \text{noir}) = 0 \quad P(Z_1 = \text{blanc}) = 1$$

ce qui signifie que le premier point ne peut qu'être blanc. Supposons maintenant que le second point suive la loi de proba :

$$P(Z_2 = \text{noir}) = 0,5 \quad P(Z_2 = \text{blanc}) = 0,5$$

ce qui signifie que le second point est noir ou blanc avec la même probabilité. Un tirage suivant Z nous donne de façon équiprobable soit $z = \{\text{blanc}, \text{noir}\}$ soit $z = \{\text{blanc}, \text{blanc}\}$.

Une remarque en passant. Nous venons de définir deux lois de probabilité distinctes, une loi pour le premier point de l'image et une pour le second point. Nous avons appelé du même nom, P , ces deux lois qui sont pourtant distinctes et, cependant, il n'y a eu aucune ambiguïté. Cela est dû à la notation $P(Z_i = z_i)$ qui remplace avantageusement une notation à laquelle vous êtes plus habitués, $P_i(v)$, qui exprime qu'une loi de probabilité P_i est dédiée au $i^{\text{ème}}$ point de l'image et que cette loi nous donne une valeur de probabilité pour toute valeur v possible de ce point. Par la suite, nous préférons donc la notation $P(Z_1 = \text{noir})$ à la notation $P_1(\text{noir})$ et nous parlerons de la loi de probabilité $P(Z_1)$ pour désigner P_1 .

2.2 Un exercice et sa solution

Nous sommes maintenant prêts pour un exercice. Conservons la loi de probabilité $P(Z_1)$ définie dans la section précédente et modifions la loi de probabilité $P(Z_2)$ (la loi du second

point de l'image) suivant :

$$P(Z_2 = \textit{noir}) = 0,6 \quad P(Z_2 = \textit{blanc}) = 0,4.$$

Nous pouvons nous demander quelle est l'image zs la *plus probable*. Cet exercice est facile ; notre logique nous dicte le bon résultat, il s'agit de $zs = \{\textit{blanc}, \textit{noir}\}$.

À retenir : définir des lois de probabilités et chercher la réalisation qui a la plus forte probabilité est une façon de poser un problème et de chercher sa solution. Ici, nous définissons une loi pour chaque point, $P(Z_i)$, et nous cherchons une image, zs , qui maximise la loi de probabilité "globale" $P(Z) = P(\{Z_i\})$.

Nous noterons :

$$zs = \arg \max_z P(Z = z)$$

pour désigner que zs est tel que $P(Z = zs) = \max_z P(Z = z)$.

2.3 Probabilités et énergies

Soit $P(var)$ une loi de probabilité ; nous désignons ici de façon générique par var ce sur quoi P s'applique. var peut être une variable aléatoire, un ensemble de variables aléatoires ou même une expression conditionnelle du type " $A|B$ " (A sachant B).

Nous savons que $P(var) \in [0, 1]$ et que $\sum_{var} P(var) = 1$. Cela nous donne donc l'idée d'exprimer la loi de probabilité sous la forme d'une énergie U suivant :

$$P(var) = \frac{1}{N} e^{-U(var)}$$

où N est un coefficient de normalisation valant :

$$N = \sum_{var} e^{-U(var)}.$$

Dans la suite, les énergies seront *toujours* positives ; soit :

$$U(var) \geq 0.$$

Comme nous l'avons remarqué en fin de section 2.1, il ne faudra pas confondre deux lois de probabilité. Ainsi, nous aurons deux énergies distinctes (avec deux coefficients normatifs *a priori* différents) pour deux lois distinctes : $P(var_1) = \frac{1}{N_1} e^{-U_1(var_1)}$ et $P(var_2) = \frac{1}{N_2} e^{-U_2(var_2)}$.

Prenons maintenant $T \in \mathbb{R}^+$, nous pouvons définir une nouvelle loi de probabilité P_T à partir de n'importe quelle loi de probabilité P . Cette modification a la forme suivante :

$$P_T(var) = \frac{1}{N_T} e^{-\frac{U(var)}{T}}.$$

Comme vous pouvez le constater, P_T s'appuie sur l'énergie U correspondant à P et le coefficient normatif de P_T dépend de T . Une autre écriture est :

$$P_T(var) = \frac{1}{N_T} e^{-U_T(var)} \quad \text{où} \quad U_T = \frac{U}{T}.$$

En section 2.2, l'exercice définissait des lois de probabilités et avait pour but de chercher la réalisation qui a la plus forte probabilité.

À comprendre : Supposons maintenant que nous définissions un problème sous la forme d'une fonction-énergie à minimiser. Avec cette énergie, nous pouvons en déduire une loi de probabilité grâce à $P(\text{var}) = e^{-U(\text{var})}/N$. Si nous connaissons une méthode qui permet de déterminer la réalisation qui maximise P , alors nous avons peut-être la solution de notre problème...

2.4 Modélisation du problème

L'image de test de vision est composée de trois éléments :

- le fond de l'image, sur les bords, en blanc ;
- un disque, au centre, de couleur verdâtre ;
- deux chiffres, 7 et 3, inscrits dans le disque et de couleur rougeâtre.

Si vous avez du mal à voir ces chiffres, vous présentez un défaut de vision. Demandez aux autres membres du groupe ce qu'ils voient. Si aucun membre du groupe n'arrive à discerner les chiffres, je vous conseille très fortement de dissoudre votre groupe de projet, de consulter un ophtalmologiste et de négocier avec un groupe ne présentant pas de défaut de vision une réorganisation de personnel.

Il existe plusieurs images de test dont les couleurs sont similaires à celle-ci (Cf. les URLs). Cette image en particulier n'est qu'une instance d'image pour qu'un patient passe ce test de vision (mesure de l'acuité à distinguer le verdâtre du rougeâtre). Le terme "instance" s'apparentant à celui de "réalisation", nous considérerons que le logiciel prend en entrée une réalisation y d'une image aléatoire Y . Du point de vue du logiciel, sur l'image en entrée, il est probable qu'il y soit écrit "73" ou "51", ou n'importe quel nombre stupide, mais le logiciel ne le sait pas *a priori*. Nous dirons que cette image en couleur s'appelle les *données*. Le logiciel connaît les valeurs y_i des points de l'image ; ce sont des vecteurs RGB (*red-green-blue*).

L'image y a été générée à partir d'une image xs dont les points ont comme valeur "fond", "disque" ou "chiffre" : $xs_i \in \Lambda$ où $\Lambda = \{\text{fond}, \text{disque}, \text{chiffre}\}$. Cette image xs ressemble à celle de la figure 2 où les trois valeurs (on parle d'*étiquettes*) sont respectivement représentées par du blanc, du gris et du noir.

Nous considérerons que xs est une réalisation de l'image aléatoire X . Le logiciel ne connaît que y , données (image) en entrée. Le but du projet est bien sûr de retrouver xs , image en sortie du logiciel. Nous pourrions alors donner cette image à Yves-Jean Daniel pour qu'un de ses réseaux de neurones effectue la reconnaissance de caractères et identifie sans problème le nombre inscrit dans le disque. Nous dirons qu'une image x d'étiquettes, comme l'est xs , s'appelle un *résultat*. Pour coder ces étiquettes, nous prendrons les conventions suivantes : *fond* = 0, *disque* = 1 et *chiffre* = 2, bref $\Lambda = \{0, 1, 2\}$.

Résumé. On cherche la "meilleure" réalisation xs d'une image aléatoire X d'étiquettes. Il nous faudra bien sûr modéliser la notion de "meilleure"! On désignera par x un *résultat*. Pour trouver xs , on dispose de y , notre image en entrée, nos *données*, réalisation de Y .



Figure 2: L'image originale recherchée (xs).

3 Questions de droit

Le projet que vous allez réaliser pourra faire l'objet d'une exploitation commerciale. Plusieurs possibilités peuvent être envisagées. Vous pourrez le céder définitivement et en perdre la maîtrise. Vous préférerez peut-être conserver un regard sur votre travail et ne permettre son utilisation que ponctuellement et dans un cadre bien défini.

Vous devez traduire par des contrats adéquats ces différentes hypothèses.

4 Questions de maths

Ces questions ont pour but de vous conduire "pas à pas" à une solution mathématique et algorithmique du problème de reconnaissance. Elles ne sont pas difficiles ! Pour la plupart d'entre elles, trouver leur réponse ne nécessite que de faire preuve d'esprit logique... De plus, sécher sur une question ne vous bloquera pas pour répondre aux questions suivantes (les questions numérotées différemment sont indépendantes des autres).

4.1 Partie modélisation

Question 1a

Nous cherchons à modéliser ce problème d'un point de vue probabiliste. Cherche-t-on $xs = \arg \max_x P(Y = y | X = x)$ ou $xs = \arg \max_x P(X = x | Y = y)$?

Justifiez-vous !

Question 1b

La formule de Bayes s'écrit :

$$P(a|b) = \frac{P(b|a)P(a)}{P(b)}.$$

Au bout du compte, que cherche-t-on à maximiser ?

Question 1c

Au fait, la question 1a est-elle vraiment pertinente ?

Justifiez-vous !

Question 2a

On va supposer par la suite que nous avons l'égalité suivante :

$$P(Y = y | X = x) = \prod_i P(Y_i = y_i | X_i = x_i). \quad (1)$$

Quelle hypothèse permet / quelles hypothèses permettent d'écrire cette égalité ?

Question 2b

Concrètement, que signifie/nt cette/ces hypothèse/s ? (nous parlons d'images dans ce problème...)

Question 2c

Dans le cadre de notre problème, cette hypothèse est-elle très raisonnable ou critiquable / ces hypothèses sont-elles très raisonnables ou critiquables ?

Justifiez-vous !

Quelques notations supplémentaires

Dans la suite, nous allons noter X^i l'image aléatoire privée de la variable aléatoire du $i^{\text{ème}}$ point ; nous avons :

$$X^i = X - \{X_i\} = \{X_1, \dots, X_{i-1}, X_{i+1}, \dots, X_n\}$$

et x^i sera une réalisation de X^i . x^i est l'image x mais dont le $i^{\text{ème}}$ point n'existe pas (il a disparu !)

De plus, nous noterons ν_i le voisinage du point i , c'est-à-dire, les points qui entourent le point i . Pour simplifier, nous considérerons que le voisinage du point de coordonnées

ligne/colonne (l, c) est constitué des 4 points de coordonnées respectives $(l - 1, c)$, $(l, c - 1)$, $(l, c + 1)$ et $(l + 1, c)$. Nous aurons donc $\forall i, \text{card}(\nu_i) = 4$.

Dans la suite, nous allons noter X_{ν_i} l'image aléatoire telle que :

$$X_{\nu_i} = \{ X_j, j \in \nu_i \},$$

et x_{ν_i} sera une réalisation de X_{ν_i} . x_{ν_i} est l'image x restreinte aux quatre points voisins du $i^{\text{ème}}$ point (c'est l'ensemble des quatre étiquettes des voisins du $i^{\text{ème}}$ point de x).

Remarquons que nous avons les propriétés suivantes :

$$\begin{aligned} X_i &\notin X^i \\ X_i &\notin X_{\nu_i} \\ X_{\nu_i} &\in X^i. \end{aligned}$$

Question 3a

Nous pourrions avoir :

$$P(X = x) = \prod_i P(X_i = x_i).$$

Dans le cadre de nos images x , est-ce bien réaliste / raisonnable ?

Justifiez-vous !

Question 3b

Nous aimerions bien avoir :

$$P(X = x) = \prod_i P(X_i = x_i | X^i = x^i) = \prod_i P(X_i = x_i | X_{\nu_i} = x_{\nu_i}). \quad (2)$$

Concrètement, qu'est-ce que cela signifie ?

Justifiez-vous !

4.2 Partie énergie

Question 4

Nous retenons finalement les équations (1) et (2). Montrez (pas à pas !) que xs vérifie finalement :

$$xs = \arg \min_x \left(\sum_i U_a(x_i, y_i) + U_b(x_i, x_{\nu_i}) \right) \quad (3)$$

où U_a et U_b sont deux fonctions-énergies distinctes associées respectivement à deux lois de probabilité.

Nota bene : en fin de section 2.3, nous avons introduit la notation U_T ; ici, les indices "a" et "b" n'ont rien à voir avec l'indice T ; ils servent seulement à différencier des formes différentes d'énergies.

Question 5a

Que représentent les énergies U_a et U_b ?

Justifiez-vous !

Forme des énergies

Dans la suite, nous noterons :

$$U_i(x_i, x_{\nu_i}, y_i) = U_a(x_i, y_i) + U_b(x_i, x_{\nu_i})$$

et :

$$U(x, y) = \sum_i U_i(x_i, x_{\nu_i}, y_i).$$

Notre problème est la recherche de xs qui minimise l'énergie $U(x, y)$:

$$xs = \arg \min_x U(x, y)$$

soit :

$$xs = \arg \min_x \left(\sum_i U_i(x_i, x_{\nu_i}, y_i) \right). \quad (4)$$

Dans la suite, nous considérerons que U_a sera définie à partir de trois fonctions-énergies suivant :

$$U_a(x_i, y_i) = \begin{cases} U_{fond}(y_i) & \text{si } x_i = 0 \\ U_{disque}(y_i) & \text{si } x_i = 1 \\ U_{chiffre}(y_i) & \text{si } x_i = 2 \end{cases} \quad (5)$$

et nous considérerons que U_b sera définie à partir d'une matrice suivant :

$$U_b(x_i, x_{\nu_i}) = \beta \sum_{j \in \nu_i} B_{x_i, x_j} \quad (6)$$

avec $\beta \in \mathbb{R}^+$ et B matrice 3×3 telle que :

$$B = \begin{bmatrix} 0 & 1 & 2 \\ 1 & 0 & 1 \\ 2 & 1 & 0 \end{bmatrix}. \quad (7)$$

Question 5b

Comment s'écrivent les énergies U_{fond} , U_{disque} et $U_{chiffre}$ (Cf. équation (5)) si elles traduisent des lois "gaussiennes" (on dit aussi "normales") ?

Par la suite, nous supposerons que les centres de ces lois sont les couleurs suivantes :

$$\begin{aligned} \bar{y}_{fond} &= (248, 252, 248) \\ \bar{y}_{disque} &= (235, 194, 155) \\ \bar{y}_{chiffre} &= (206, 200, 149). \end{aligned}$$

Question 5c

Quel sens donner à la définition de U_b (équation (6)) et à la définition de B (équation (7)) ?

Question 6a

Pour faciliter la recherche de xs , une première idée serait de simplifier notre modélisation du problème. Pouvons-nous écrire :

$$\min_x U(x, y) = \sum_i \min_x U_i(x_i, x_{\nu_i}, y_i) \quad ?$$

Justifiez-vous !

Question 6b

Quelle/s condition/s nous permettrait/aient d'écrire à partir de l'équation (4) :

$$xs_i = \arg \min_{x_i} U_i(x_i, x_{\nu_i}, y_i) \quad ?$$

Justifiez-vous !

Question 7a

Soit une loi de probabilité $P(var) = \frac{1}{N} e^{-U(var)}$ dont nous ne connaissons rien. Notons $P_\infty(var) = \lim_{T \rightarrow +\infty} P_T(var)$.

Démontrer que la loi P_∞ est une loi uniforme.

Question 7b

Supposons que la loi P présente un maximum global pour $var = vs$. Dit autrement :

$$vs = \arg \max_{var} P(var).$$

Posons $P_0(var) = \lim_{T \rightarrow 0} P_T(var)$. Quelle est la loi P_0 ?

Question 7c

Supposons maintenant que la loi P présente un même maximum global pour plusieurs valeurs de var : vs^1, \dots, vs^m . Dit autrement :

$$P(vs^1) = \dots = P(vs^m) = \max_{var} P(var).$$

Quelle est maintenant la loi P_0 ?

Algorithme A1

```
// initialisation
pour tout point  $i$ 
  • affecter une étiquette au hasard, soit :  $x_i = rand_\Lambda$ 
// corps
itérer
  • tirer un point au hasard, soit :  $i = rand_n$ 
  • déterminer  $e_i = \arg \min_{e \in \Lambda} U_i(e, x_{\nu_i}, y_i)$ 
  • réaliser l'affectation  $x_i = e_i$ 
```

Algorithme A2

```
// initialisation
pour tout point  $i$ 
  • affecter une étiquette au hasard, soit :  $x_i = rand_\Lambda$ 
// corps
itérer
  • tirer un point au hasard, soit :  $i = rand_n$ 
  • effectuer un tirage suivant la loi de proba  $P_i(x_i) = \frac{1}{N_i} e^{-U_i(x_i, x_{\nu_i}, y_i)}$ 
```

Algorithme A3

```
// initialisation
pour tout point  $i$ 
  • affecter une étiquette au hasard, soit :  $x_i = rand_\Lambda$ 
// corps
itérer
  • tirer un point au hasard, soit :  $i = rand_n$ 
  • tirer une étiquette au hasard, soit :  $e = rand_\Lambda$ 
  • calculer  $\delta = U_i(e, x_{\nu_i}, y_i) - U_i(x_i, x_{\nu_i}, y_i)$ 
  • effectuer l'affectation  $x_i = e$ 
    systématiquement si  $\delta < 0$ 
    ou avec la probabilité  $p = e^{-\delta}$  si  $\delta > 0$ 
```

Figure 3: Trois algorithmes.

4.3 Partie algorithmique

La figure 3 présente trois algorithmes. La notation *rand* effectue un tirage uniforme ; par exemple, *rand_{Lambda}* donne 0, 1 ou 2 avec la même probabilité.

Les questions 8* portent sur l'algorithme **A1**. Les questions 9* portent sur l'algorithme **A2**. Enfin, les questions 10* portent sur l'algorithme **A3**.

Question 8a

Vers quoi converge cet algorithme ?

Justifiez-vous !

Question 8b

Si nous modifions le corps de l'algorithme en :

pour tout point $i = 1 \dots n$

- déterminer $e_i = \arg \min_{e \in \Lambda} U_i(e, x_{\nu_i}, y_i)$
- réaliser l'affectation $x_i = e_i$

que se passe-t-il ?

Question 8c

Si nous modifions le corps de l'algorithme en :

itérer

- pour tout point $i = 1 \dots n$
 - déterminer $e_i = \arg \min_{e \in \Lambda} U_i(e, x_{\nu_i}, y_i)$
 - réaliser l'affectation $x_i = e_i$

quel problème survient ?

Question 8d

Au bout du compte, nous ne modifierons pas l'algorithme **A1**. Quelle est la condition d'arrêt de la boucle ?

Justifiez-vous !

Question 8e

Pour accélérer la convergence, une *autre* initialisation serait la bienvenue ; laquelle ?

Justifiez-vous !

Question 9a

Les fonctions de tirage aléatoires $rand_\Lambda$ et $rand_n$ sont très faciles à simuler avec la fonction `rand` du langage C. Pour l'algorithme **A2**, nous devons effectuer un tirage suivant la loi de probabilité $P_i(x_i) = \frac{1}{N_i} e^{-U_i(x_i, x_{\nu_i}, y_i)}$. Nous pouvons calculer $P_i(0)$, $P_i(1)$ et $P_i(2)$.

Mais comment réaliser un tirage d'étiquette suivant cette loi ? (Donner un pseudo-code de l'algorithme qui réalise ces tirages.)

Question 9b

En fait, l'algorithme **A2** ne converge pas et ce n'est pas grave du tout ! Si nous l'interrompons au bout de t itérations, nous obtenons une réalisation $x^{(t)}$ (lire : un x issu de l'itération numéro t).

Deux remarques : $x^{(t)}$ a été obtenu sachant que l'initialisation nous a fourni $x^{(0)}$; pour tout t , nous noterons $X^{(t)}$ l'image aléatoire dont $x^{(t)}$ est une réalisation.

À votre avis, quelle propriété vérifie $x^{(t)}$ lorsque t tend vers $+\infty$? (Aucune démonstration formelle ne vous est demandée.)

Question 9c

Justifiez l'allure de cet algorithme et dites pourquoi il ne peut pas s'écrire plus simplement.

Question 10a

L'algorithme **A3** a, à peu de choses près, le même comportement que l'algorithme **A2** : x au bout d'un nombre infini d'itérations a la même propriété sympathique.

Que se passe-t-il si nous modifions l'algorithme **A3** en **A3bis** décrit en figure 4 ?

Question 10b

Quelle condition d'arrêt est raisonnable ?

Justifiez-vous !

Question 10c

Y a-t-il un lien entre cet algorithme et le métier de forgeron ? Si oui, lequel ? Si non, lequel ?

Question 73 (bonus)

Comment améliorer *significativement* la modélisation mathématique de notre problème pour obtenir de *très* bons résultats ?

(Bien sûr, cela signifie que des hypothèses n'étaient pas très justifiées...)

5 Implémentation

Il s'agit de mettre en œuvre l'algorithme **A3bis**. Je vous conseille de prendre comme initialisation $T = 666$ et comme modification de T : $T = 0,95 \times T$. Vous pouvez vous amuser à faire varier β (Cf. l'équation (6)).

Enfin, effectuer tout d'abord vos tests sur l'image `~theo/pub/maths/73_small_part.ppm` puis sur `~theo/pub/maths/73_small.ppm` et finalement sur `73.ppm`. À titre indicatif, j'obtiens l'image de la figure 5.

```

// initialisation
pour tout point  $i$ 
  • affecter une étiquette au hasard, soit :  $x_i = rand_\Lambda$ 
  • donner à  $T$  une grande valeur
  •  $t = 0$ 
// corps
itérer
  • incrémenter  $t$  ; soit :  $t = t + 1$ 
  • tirer un point au hasard, soit :  $i = rand_n$ 
  • tirer une étiquette au hasard, soit :  $e = rand_\Lambda$ 
  • calculer  $\delta = U_i(e, x_{\nu_i}, y_i) - U_i(x_i, x_{\nu_i}, y_i)$ 
  • effectuer l'affectation  $x_i = e$ 
    systématiquement si  $\delta < 0$ 
    ou avec la probabilité  $p = e^{-\frac{\delta}{T}}$  si  $\delta > 0$ 
  • si  $t \text{ div } n = 0$ 
    diminuer légèrement  $T$ 

```

Figure 4: L'algorithme **A3bis**.

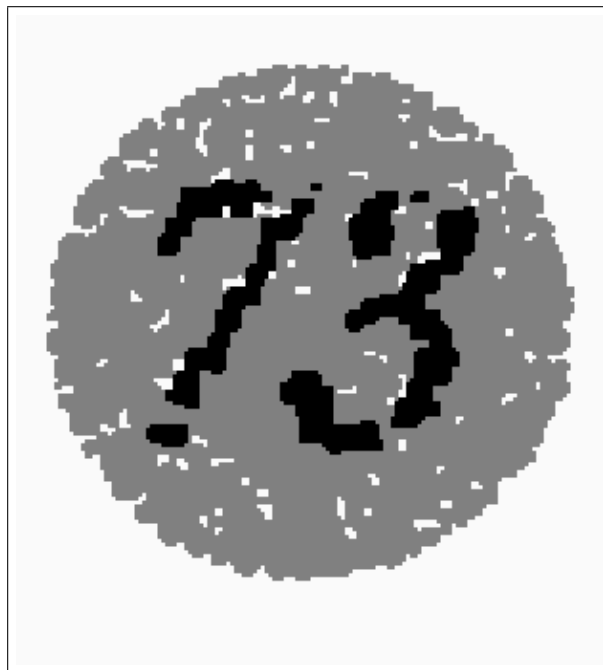


Figure 5: Une image résultat.