arXiv:1211.5712v1 [cs.CV] 24 Nov 2012

Detection of elliptical shapes via cross-entropy clustering

Jacek Tabor[1] and Krzysztof Misztal[1,2]

[1] Jagiellonian University
Faculty of Mathematics and Computer Science
Łojasiewicza 6, 30-348 Kraków, Poland
tabor@ii.uj.edu.pl
[2] AGH University of Science and Technology
Faculty of Physics and Applied Computer Science
al. A. Mickiewicza 30, 30-059 Kraków, Poland
Krzysztof.Misztal@fis.agh.edu.pl

Abstract. The problem of finding elliptical shapes in an image will be considered. We discuss the solution which uses cross-entropy clustering. The proposed method allows the search for ellipses with predefined sizes and position in the space. Moreover, it works well for search of ellipsoids in higher dimensions.

Keywords: cross-entropy, MLE, EM, image processing, pattern recognition, clustering, classification

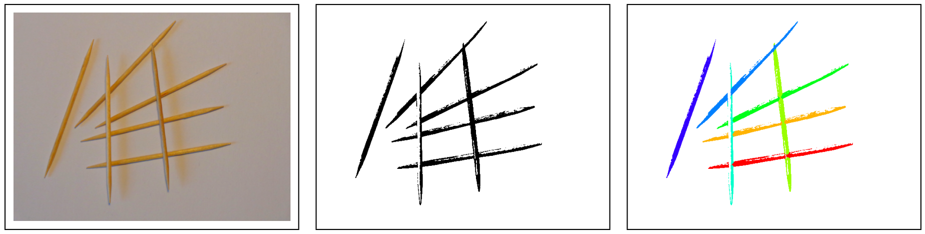
1 Introduction

Ellipse detection is one of the most important problems in image processing. It has been researched using a good variety of methods, see i.e. Tsuji and Matsumoto [14], Davies [3]. Most of the existing techniques use the Hough Transform [7] – that is very memory and time consuming.

In this paper a new approach will be presented and its advantages and disadvantages will be discussed. We show the results of the algorithm on the pictures from Fig. 1. The algorithm discussed in this paper:

- is easily adaptable, ie. if we know the expected shape of the object sought, or its position (orientation) in space, by little calculation we can prepare a proper configuration for its detection;
- can detect simultaneously multiple type of objects, ex. we can look for matches and coins at the same time;
- is rather insensitive to the disturbance of the picture (such as bluring, contrast and illumination modification, etc);
- can be used for classification (we can detect specified shapes) and for clustering (we can use it for exploring the data structure).

The acceptable disadvantage of the presented method is that to work well we need the beforehand knowledge that on the picture we study there are no other



(a) (b) (c)

Fig. 1: The result of our algorithm: Fig 1a – original image, Fig 1b – binarized image, the input for our algorithm, Fig 1a – outcome form algorithm, clusters marked in different colors.

objects than ellipse-like shapes. Consequently, our approach is well-adapted for example to the following tasks:

- count the number of ellipses on the picture;
- divide the shapes into circles of different radiuses;
- count the number of vertical and horizontal ellipses.

Our idea uses a cross-entropy clustering [13] (CEC), which from the practical point of view can be seen as joining of the k-means method with the model approach used in expectation maximization (EM). EM [9,1,10] is one of the basic and most important applications of maximal likelihood in the density estimations [8]. EM, or its variations like classification EM [12] are often applied in clustering. Although EM approach is quite general, and gives good results, to apply it we usually need to first perform complicated computations. Moreover, to accomplish the M step one commonly needs numerically consuming minimization techniques, and consequently EM is relatively slow and cannot deal well with large data.

Our aim in this paper is to show that CEC is well-adapted to classification and detection of ellipses and ellipsoids. The advantage of CEC over EM is simplicity and speed – in the case of typical Gaussian families we do not need the M-step, which enables us in particular to use fast and efficient Hartigans approach. Moreover, as the use of every cluster in CEC has its cost, contrary to classification EM, CEC reduces on-line clusters which carry no information, which in practice implies that our algorithm can find the "right" number of ellipses on the picture.

Let us discuss the contents of the paper. In the first part of our work we briefly describe the CEC algorithm. In the next section we present the basic models we use (compare with [4]). We also present results of numerical experiments. Then we describe the procedure for finding toothpicks in the image (see Fig. 1).

In Appendix we provide the proof of the only cross-entropy formula from section which is essentially new. In our opinion its proof is worth including as in fact it given a method which can be easily used in search for cross-entropy in other Gaussian subfamilies.

2 Theoretical background of CEC

In this section we give a short introduction to CEC, for more detailed explanation we refer the reader to [13]. To explain CEC we need to introduce the "energy function" we want to minimize. By the cross-entropy of the probability measure μ (which represent the data-set we study) with respect to density f we understand

$$H^{\times}(\mu\|f) = -\int_{\mathbb{R}^N} \ln f(y)\, d\mu(y).$$

The above cross-entropy corresponds to the theoretical code-length of compression of μ-randomly chosen element of $\mathbb{R}^N$ with the code optimized for density f [2]. In a more general case when one is interested in (best) coding for μ by densities chosen from family $\mathcal{F}$, we arrive at *the cross-entropy of* μ *with respect to a family of coding densities* $\mathcal{F}$

$$H^{\times}(\mu\|\mathcal{F}) := \inf_{f\in\mathcal{F}} H^{\times}(\mu\|f).$$

In the case of splitting of $\mathbb{R}^N$ into pairwise disjoint sets $U_1,\ldots,U_N$ such that elements of U_i we "code" by optimal density from family $\mathcal{F}_i$, the mean code-length of randomly chosen element x equals

$$E_\mu(U_1,\mathcal{F}_1;\ldots;U_n,\mathcal{F}_n) := \sum_{i=1}^{k} \mu(U_i)\cdot(-\ln(\mu(U_i)) + H^{\times}(\mu_{U_i}\|\mathcal{F}_i)), \tag{1}$$

where μ_U denotes the normalized restriction of μ to the set U and is given by $\mu_U(A) := \frac{1}{\mu(U)}\mu(A\cap U)$.

The aim of CEC is to find splitting of $\mathbb{R}^N$ into pairwise disjoint sets U_i which minimize the function given in (1). In this paper we restrict for the sake of simplicity to clusters generated by Gaussian densities (although one can easily use any density family for which MLE can be performed).

Now we proceed with discussion of the Gaussian models we will use in CEC. We consider following density families:

1. $\mathcal{G}_\Sigma$ – Gaussian densities with covariance Σ. The clustering will have the tendency to divide the data into clusters resembling the unit circles in the Mahalanobis distance given by $\|x-y\|^2_\Sigma := (x-y)^T\Sigma(x-y)$. Its particular important subfamily is given by $\mathcal{G}_{r\mathrm{I}}$, where $r>0$ is fixed (in this case we will have tendency to divide the data into "circles" with approximate radius of $\sqrt{r}$).
2. $\mathcal{G}_{(\cdot I)}$ – spherical Gaussian densities, which covariance is proportional to identity. The clustering will try to divide the data into circles of arbitrary sizes.
3. $\mathcal{G}_{\mathrm{diag}}$ – Gaussians with diagonal covariance. The clustering will try to divide the data into ellipsoid with radiuses parallel to coordinate axes.
4. $\mathcal{G}$ – all Gaussian densities. In this case we divide dataset into ellipsoid-like clusters without any preferences concerning the size or shape or position in space of the ellipsoid.

We need a result which says what is the cross-entropy of the probability measure μ with respect to coding adapted for the respective Gaussian subfamilies. A basic role is played by the following observation.

Observation 21 *Let μ be a discrete or continuous probability measure in $\mathbb{R}^N$ with well-defined mean $m_\mu := \int x d\mu(x)$ and covariance matrix $\Sigma_\mu := \int (x - m_\mu)(x - m_\mu)^T d\mu(x)$. Let a fixed positive-definite symmetric matrix Σ be given.*

Then $H^\times\big(\mu \| \mathcal{G}_\Sigma\big) = H^\times\big(\mu_\mathcal{G} \| \mathcal{N}(\mathrm{m}_\mu, \Sigma)\big)$, where $\mu_\mathcal{G}$ denotes the probability measure with Gaussian density of the same mean and covariance as μ. Consequently

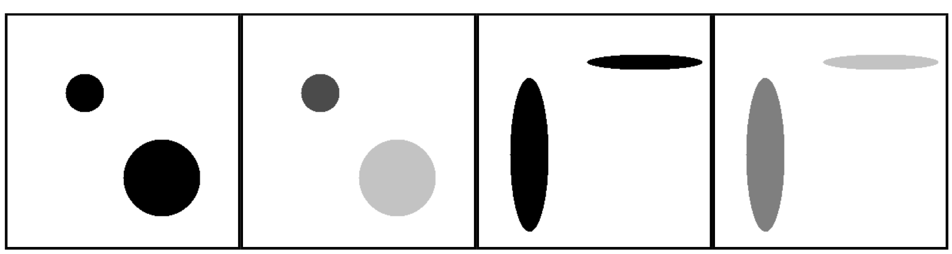
$$H^\times\big(\mu \| \mathcal{G}_\Sigma\big) = \frac{N}{2}\ln(2\pi) + \frac{1}{2}\mathrm{tr}(\Sigma^{-1}\Sigma_\mu) + \frac{1}{2}\ln\det(\Sigma). \tag{2}$$

By applying the above proposition one can easily deduce[3] the formulas for cross-entropy given the Table 1.

$\mathcal{F}$	cov. matrix	$H^\times\big(\mu\|\mathcal{F}\big)$
$\mathcal{G}_\Sigma$	Σ	$\frac{N}{2}\ln(2\pi) + \frac{1}{2}\mathrm{tr}(\Sigma^{-1}\Sigma_\mu) + \frac{1}{2}\ln\det(\Sigma)$
$\mathcal{G}_{r\mathrm{I}}$	$r\mathrm{I}$	$\frac{N}{2}\ln(2\pi) + \frac{1}{2r}\mathrm{tr}(\Sigma_\mu) + \frac{N}{2}\ln r$
$\mathcal{G}_{(\cdot\mathrm{I})}$	$\frac{\mathrm{tr}(\Sigma_\mu)}{N}\mathrm{I}$	$\frac{N}{2}\ln(2\pi e/N) + \frac{N}{2}\ln(\mathrm{tr}\Sigma_\mu)$
$\mathcal{G}_{\mathrm{diag}}$	$\mathrm{diag}(\Sigma)$	$\frac{N}{2}\ln(2\pi e) + \frac{1}{2}\ln(\det(\mathrm{diag}(\Sigma_\mu)))$
$\mathcal{G}$	Σ_μ	$\frac{N}{2}\ln(2\pi e) + \frac{1}{2}\ln\det(\Sigma_\mu)$

Table 1: Table of cross-entropy formulas with respect to Gaussian subfamilies.

In the second column we give the formula for the covariance matrix of the Gaussian density which realizes the desired minimum of cross-entropy (obviously the mean is always the mean of the measure). Simple applications of the formulas given above can be found on the Figure 2.



(a) (b) (c) (d)

Fig. 2: The simplest case: input and outcome for our algorithm applied to $\mathcal{G}_{r\mathrm{I}}$ (Fig. 2a and 2b) and $\mathcal{G}_{\mathrm{diag}}$ (Fig. 2c and 2d).

[3] In practice all the formulas given in the are known, see for example [13].

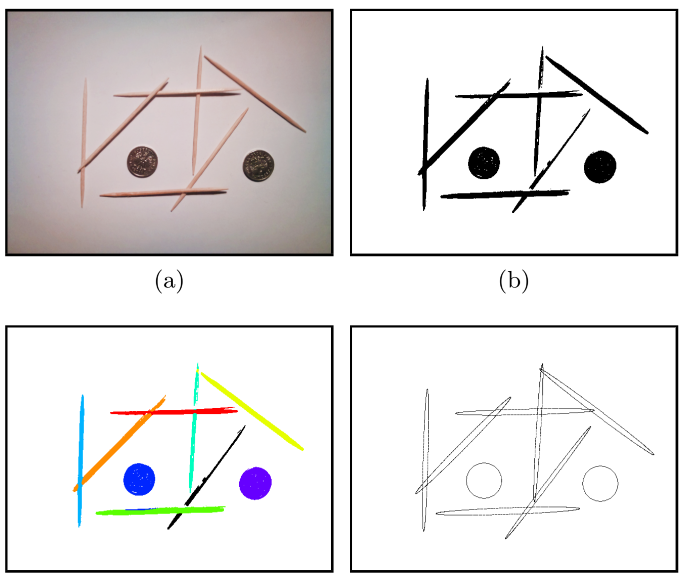
3 Case study

Let us explain the method on the following simple problem: assume that we want to count the toothpicks on the Fig. 3. To do so we take a particular object and compute its covariance matrix. We have obtained a covariance with eigenvalues

$$\lambda_1 = 4938.5 \text{ and } \lambda_2 = 5.7.$$

Since we want to allow the toothpick to have any position in space, we introduce the set $\mathcal{G}_{\lambda_1,\lambda_2}$ to consist of all Gaussian densities on the plane with covariance matrix having eigenvalues λ_1 and λ_2 (observe that this set is rotation and translation invariant, but not scale invariant).

Consider now a probability measure μ, representing our data, with covariance Σ_μ, with eigenvalues $\lambda_1^\mu > \lambda_2^\mu > 0$. By applying Proposition 1 (see Appendix) jointly with Observation 21 we easily conclude that the best approximation (understood in the maximal likelihood or equivalently cross-entropy, sense) of μ in $\mathcal{G}_{\lambda_1,\lambda_2}$ is given by the Gaussian density with covariance matrix with the same eigenvectors as Σ_μ and eigenvalues λ_1 and λ_2. Consequently, the cross-entropy,



(a) (b)

(c) (d)

Fig. 3: The result of our algorithm: Fig 3a – original image, Fig 3b – binarized image, the input for our algorithm, Fig 3c – outcome form algorithm, clusters marked in different colors, Fig 3d – outcome form algorithm, ellipses with the same mean and covariance as calculated by algorithm densities.

which plays the role of energy, $H^{\times}(\mu\|\mathcal{G}_{\lambda_1,\lambda_2})$ thanks to (2) is given by

$$H^{\times}(\mu\|\mathcal{G}_{\lambda_1,\lambda_2}) = \frac{N}{2}\ln(2\pi) + \frac{1}{2}(\lambda_1^{\mu}/\lambda_1 + \lambda_2^{\mu}/\lambda_2) + \frac{1}{2}(\ln(\lambda_1) + \ln(\lambda_2)).$$

By applying Hartigan approach we can now find the splitting of the data into pairwise disjoint sets $U_1, \ldots, U_k$ which minimizes the value of (1). Results of our method can be seen on Figure 3 (we omit here the natural preliminary binarization procedure).

To visualize the found clusters, we draw the boundary of an ellipse with the same mean and covariance as a given density estimator[4].

4 Conclusion

We have proposed a new method, which uses cross-entropy clustering approach, to classification and detection of ellipse-like shapes. The main advantage of the method lies in the fact that it can be easily adapted to finding ellipses of desired shape and position in space. The basic disadvantage is that in current algorithm configuration (basic approach) we can deal only with pictures which contain only ellipse-like shapes (for example we cannot discover ellipses in a picture with ellipses and rectangles). Our further work will consist on elimination of this inconvenience.

5 Appendix: how to compute MLE for Gaussian families

The situation is very simple if we search for the MLE, or in other words for the minimum in (2) in the class of diagonal matrices (subclass consisting of Gaussians with independent variables). A more requiring and difficult question is to find the desired minimum in the class of all Gaussians. Below we present an approach which allows to do this.

We will use the well-known von Neumann trace inequality [6,11]:

Theorem [von Neumann trace inequality]. *Let E, F be complex $N \times N$ matrices. Then*

$$|\mathrm{tr}(EF)| \leq \sum_{i=1}^{N} s_i(E) \cdot s_i(F), \tag{3}$$

where $s_i(D)$ denote the ordered (decreasingly) singular values of matrix D.

Let us recall that for the symmetric positive matrix its eigenvalues coincide with singular values.

Given $\lambda_1, \ldots, \lambda_N \in \mathbb{R}$ by $S_{\lambda_1,\ldots,\lambda_N}$ we denote the set of all symmetric matrices with eigenvalues $\lambda_1, \ldots, \lambda_N$. The following proposition plays the basic role in the search for optimal Gaussian densities, as it reduces the search from all symmetric matrices to search in the set of eigenvalues. Since its proof is short, we provide it for the sake of completeness.

[4] We recall that covariance matrix of a uniform density of an ellipse with radiuses r_1, r_2 is given by $[r_1^2/4, 0; 0, r_2^2/4]$, that is we draw the ellipse with radiuses $2\sqrt{\lambda_i}$.

Proposition 1. *Let B be a symmetric nonnegative matrix with eigenvalues $\beta_1 \geq \ldots \geq \beta_N \geq 0$. Let $0 \leq \lambda_1 \leq \ldots \leq \lambda_N$ be fixed. Then*

$$\min_{A \in S_{\lambda_1,\ldots,\lambda_N}} \mathrm{tr}(AB) = \sum_i \lambda_i \beta_i.$$

Proof. Let e_i denote the orthogonal basis build from the eigenvectors of B, and let operator $\bar{A}$ be defined in this base by $\bar{A}(e_i) = \lambda_i e_i$. Then trivially

$$\min_{A \in S_{\lambda_1,\ldots,\lambda_N}} \mathrm{tr}(AB) \leq \mathrm{tr}(\bar{A}B) = \sum_i \lambda_i \beta_i.$$

To prove the inverse inequality we will use the von Neumann trace inequality. Let $A \in S_{\lambda_1,\ldots,\lambda_N}$ be arbitrary. We apply the inequality (3) for $E = \lambda_N \mathrm{I} - A$, $F = B$. Since E and F are symmetric nonnegatively defined matrices, their eigenvalues $\lambda_N - \lambda_i$ and β_i coincide with singular values, and therefore by (3)

$$\mathrm{tr}((\lambda_N \mathrm{I} - A)B) \leq \sum_i (\lambda_N - \lambda_i)\beta_i = \lambda_N \sum_i \beta_i - \sum_i \lambda_i \beta_i. \tag{4}$$

Since $\mathrm{tr}((\lambda_N \mathrm{I} - A)B) = \lambda_N \sum_i \beta_i - \mathrm{tr}(AB)$, from inequality (4) we obtain that $\mathrm{tr}(AB) \geq \sum_i \lambda_i \beta_i$.

Corollary 1. *Assume that we want to find the best fit of μ with covariance Σ_μ in the class $\mathcal{G}_{\lambda_1,\ldots,\lambda_n}$, where $\lambda_1 \geq \ldots \geq \lambda_n > 0$.*

To do so we take the eigenvalues $\lambda_1^\mu \geq \ldots \geq \lambda_n^\mu$ corresponding to orthonormal eigenvectors $e_1^\mu, \ldots, e_n^\mu$, and then Σ is given in the base as a diagonal matrix with $\lambda_1, \ldots, \lambda_n$ on the diagonal.

References

1. Celeux, G. and Govaert, G.: Gaussian parsimonious clustering models. Pattern Recognition, 28(5):781–793, 1995.
2. Cover, T.M. and Thomas, J.A. and Wiley, J. et al.: Elements of information theory, volume 6. Wiley Online Library, 1991.
3. Davies, E.R.: Finding ellipses using the generalised Hough transform. Pattern Recognition Letters, 9 (2), 87–96, 1989.
4. Fraley, C.: Algorithms for model-based Gaussian hierarchical clustering. SIAM Journal on Scientific Computing, 20(1):270–281, 1998.
5. Free Stock Images. `http://www.stockfreeimages.com/`
6. Grünwald, P.D. and Myung, I.J. and Pitt, M.A.: Advances in minimum description length: Theory and applications. the MIT Press, 2005.
7. Illingworth, J. and Kittler, J.: A survey of the Hough transform Computer vision, graphics, and image processing, 44 (1), 87–116, 1988.
8. Lehmann, E.L. and Casella, G.: Theory of point estimation, volume 31. Springer Verlag, 1998.
9. McLachlan, G.J. and Krishnan, T.: The EM algorithm and extensions. 274, Wiley New York, 1997.

10. Mcnicholas, P.D. and Murphy, T.B.: Parsimonious gaussian mixture models. Statistics and Computing, 18(3):285–296, 2008.
11. Mirsky, L.: A trace inequality of John von Neumann. Monatsh. Math., 79 (4):303–306, 1975.
12. Samé, A. and Ambroise, C. and Govaert, G.: An online classification em algorithm based on the mixture model. Statistics and Computing, 17(3):209–218, 2007.
13. Tabor, J. and Spurek, P.: Cross-entropy clustering. Available from `http://arxiv.org/pdf/1210.5594.pdf`, 2012.
14. Tsuji, S. and Matsumoto, F.: Detection of ellipses by a modified Hough transformation IEEE Trans. Comput., C-27 (1978), pp. 777–781