

## PATTERN DETECTION BY DISTRIBUTED FEATURE EXTRACTION

*Effrosyni Kokiopoulou and Pascal Frossard*

Ecole Polytechnique Fédérale de Lausanne (EPFL)  
Signal Processing Institute - ITS  
CH- 1015 Lausanne, Switzerland

### ABSTRACT

This paper presents a distributed algorithm for the detection of patterns or their transformed versions, in noisy images. The proposed method projects the observed signal onto a redundant and structured dictionary of functions, which are distributed among general purpose vision sensors. Each of the sensors then approximates the projections on its own part of the dictionary, and transmits that short information to a central fusion center. The pattern detection problem is then cast to a parameter estimation problem, where the parameters of the geometric transformation of the pattern of interest are sought, instead of the pattern itself. The parameters of the transformation are estimated by introducing a score function over the parameter space. Such an approach allows the fusion center to directly work in the space of features computed by the sensors, without need for signal reconstruction. It advantageously provides a generic approach, where the processing of the image is directly driven by the detection task. Experimental results indicate the effectiveness of the proposed method and its resiliency to noise in the observation.

### 1. INTRODUCTION

Recent progress in processor and communication technology have motivated a lot of research efforts on sensor networks, which is an emerging and promising field in signal processing community. In the same time, pattern detection becomes an increasingly important task with numerous applications including object recognition and tracking, as well as a wide range of security applications. In that context, distributed algorithms are getting a lot of attention [1, 2], as they allow to shift the computational complexity towards the receiving end, possibly without loss in performance. In [1] the authors propose a collaborative signal processing framework for tasks such as detection and tracking and in [2] the proposed system combines wavelet filters based on soft thresholding, with distributed detection fusion, in the context of sensor networks.

In this paper, we study the distributed pattern detection problem where an observed image  $x$  contains a pattern  $s$ ,

which may be present in a geometrically transformed version. We formulate the pattern detection problem as a parameter estimation problem, where the parameters of the pattern transformation are sought. It allows to work directly in the space of geometrical features computed by the vision sensors, and avoids the need for signal reconstruction. Feature extraction therefore becomes specific to the detection task; since it does not need to provide accurate signal representation, but rather detect meaningful information for pattern detection. In the same time, it stays generic with respect to the pattern to be detected, thanks to the flexible design of the system.

We investigate the particular scenario of distributed pattern detection, in a network of inexpensive general purpose vision sensors with severe limitations on memory and power capabilities. The sensors take measurements of the observed image  $x$  by projecting it on a redundant dictionary of visual primitives [3], which is distributed among the different sensors. They each compute a crude approximation of the correlation results of  $x$  with the most important atoms from their own sub-dictionary, and send that short information to a central processing unit, usually called Fusion Center (FC). The FC uses a score function over all possible transformations and the maximum of this function is returned as the estimated parameters. We provide experimental results in both noiseless and noisy case which demonstrative the validity and the effectiveness of the proposed method.

### 2. PROBLEM FORMULATION

We consider the pattern detection problem that consists in recovering a possibly transformed version of a signal  $s$  from the Hilbert space  $\mathcal{H}$ , in a noisy observation  $x$  given as:

$$x = \beta U(\eta^*)s + e, \quad x \in \mathcal{H}, \quad (1)$$

where  $U(\eta^*)$  represents a unitary operator applied on the signal  $s$ , and  $\beta$  is a multiplicative scalar factor. The noise  $e$  represents the part of  $x$  that does not belong to the pattern of interest. We consider that the pattern can undergo any geometric transformation from the group of similitude in  $\text{SO}(2)$ . In other words, the operator  $U(\eta^*)$  can contain the following three types of operations : translations  $\vec{b}$ , isotropic scaling  $a$ , rotation  $\theta$  (in  $\text{SO}(2)$ ) or a synthesis of them.

---

This work has been partly supported by the Swiss National Science Foundation, under grants PP-002-68737, and NCCR IM2.

We consider the framework of overcomplete signal expansions over structured geometric dictionaries, which has been successfully used for sparse image representations [3]. A redundant dictionary  $\mathcal{D}$  is constructed by applying geometric transformations  $\gamma$  to a mother function  $\phi$ , i.e.,

$$\mathcal{D} = \{\phi_\gamma, \phi_\gamma = U(\gamma)\phi, \gamma \in \Gamma\}, \quad (2)$$

where  $\Gamma$  is an index set on the parameter space. The components  $\phi_\gamma$  are called *atoms*. Note that all atoms must have unit norm, i.e.,  $\|\phi_\gamma\|_2 = 1, \forall \gamma$ , and that the index set has to be chosen such that  $\mathcal{D}$  spans the Hilbert space  $\mathcal{H}$ . We further assume that  $s$  can be approximated by a sparse decomposition over the redundant dictionary  $\mathcal{D}$ , as :

$$\tilde{s} = \sum_{n=0}^{N-1} c_{\gamma_n} \phi_{\gamma_n}, \quad (3)$$

where the approximation error  $\delta = \|s - \tilde{s}\|_2$  is small. Such approximations can be obtained typically by greedy algorithms like Matching Pursuit (MP) [4].

With a more detailed look in equation (1), we can now study the effect of the operator  $U(\eta)$  on the signal  $s$ . By assuming a sparse representation of  $s$ , the approximation error  $\delta$  is small, and it becomes sufficient to study the effect of the operator on the approximation  $\tilde{s}$ . The operator  $U(\eta)$  is a synthesis of translations, isotropic scalings and rotations, and the resulting parameters form a group, namely the similitude group SIM(2) on the 2D plane. Denote by

$$r_\theta = \begin{bmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{bmatrix}, \quad 0 \leq \theta < 2\pi$$

the rotation matrix in the 2D plane. If  $(\vec{b}, a, \theta)$  and  $(\vec{b}', a', \theta')$  are two elements of the group, then the group law is

$$(\vec{b}, a, \theta)(\vec{b}', a', \theta') = (\vec{b} + ar_\theta(\vec{b}'), aa', \theta + \theta'). \quad (4)$$

Now observe that using the group law, the effect of  $U(\eta)$  on  $\tilde{s}$  using equation (3) can be expressed as

$$U(\eta)\tilde{s} = \sum_{n=0}^{N-1} c_{\gamma_n} U(\eta)\phi_{\gamma_n} = \sum_{n=0}^{N-1} c_{\gamma_n} U(\eta \circ \gamma_n)\phi, \quad (5)$$

where the operation  $\eta \circ \gamma_n$  denotes the synthesis of geometric transformations  $\eta$  and  $\gamma_n$ .

It can be seen that a transformation of the signal  $s$  is equivalent to the same transformation applied to the atoms that approximate  $s$ , and equivalently to a composition of transformations applied to the generating function of the dictionary  $\mathcal{D}$ . Hence using the group law described above, the pattern detection problem can be equivalently formulated as a parameter estimation problem which is formally stated as follows:

**Problem 1.** *Given the noisy observed signal  $x$  from equation (1), the pattern  $s$  and its transformed versions  $U(\eta)\tilde{s}$  obtained from equation (5), find the parameters of the transformation  $\eta^*$  that has been most likely applied to  $s$  in  $x$ .*

### 3. PATTERN TRANSFORM PARAMETER ESTIMATION

This section describes the algorithm to solve the parameter estimation Problem 1, in its non distributed form. For each possible transformation  $\eta = (a, \vec{b}, \theta)$  we employ a (correlation based) score function

$$\zeta(\eta) = |\langle x, U(\eta)\tilde{s} \rangle|. \quad (6)$$

The estimated parameters of the exact transformation  $\eta^*$  are simply the ones which result in the highest score value, when correlation is computed between  $x$  and transformed version of the pattern  $s$ . Therefore, we need to solve an optimization problem where the maximum of  $\zeta(\eta)$  with respect to  $\eta$  (4 parameters), is sought. The objective function is clearly non-convex, which excludes the use of efficient steepest descent-type of methods for solving the optimization problem. We have rather chosen to employ a simple numerical optimization scheme where we discretize uniformly the parameter space  $H$  and evaluate the objective function in each grid point. The parameters  $\eta$  which will give the highest score are simply returned as the optimal transform parameters  $\eta^*$ .

Due to the group law described in the previous section, the score function can be evaluated without need for the signal  $x$ , nor the reconstruction of transformation of  $s$ . That results in a reduced complexity detection algorithm, that works as follows. First we project  $x$  on  $\mathcal{D}$ , to compute all inner products of the form  $\langle x, \phi_\gamma \rangle, \gamma \in \Gamma$ . Then, using equation (3), the score function  $\zeta(\eta)$  can be evaluated as :

$$\begin{aligned} \zeta(\eta) &= |\langle x, U(\eta)\tilde{s} \rangle| \\ &= \left| \sum_{n=0}^{N-1} c_{\gamma_n} \langle x, U(\eta)\phi_{\gamma_n} \rangle \right| \\ &= \left| \sum_{n=0}^{N-1} c_{\gamma_n} \langle x, \phi_{\eta \circ \gamma_n} \rangle \right| = |\langle c, q \rangle|. \end{aligned} \quad (7)$$

By design assumptions, the atoms  $\phi_{\eta \circ \gamma_n}$  exist in the initial full dictionary  $\mathcal{D}$  and can be localized using the group law given in equation (4). Hence, once we have projected  $x$  on the dictionary, the score evaluation for each  $\eta$ , involves a simple inner product between  $c = [c_{\gamma_0}, \dots, c_{\gamma_{N-1}}]$  and  $q = [\langle x, \phi_{\eta \circ \gamma_0} \rangle, \dots, \langle x, \phi_{\eta \circ \gamma_{N-1}} \rangle]$ .

It is important to note that the projection and the score evaluation parts are well separated, and this is crucial for the distributed algorithm described in the next section. It has the advantage that the search for another pattern  $s'$  only requires the repetition of the score evaluation part. As opposed to [5], the projection part does not need to be repeated, which makes the proposed approach appropriate for use in network of general purpose vision sensors.

A small computational trick allows to further reduce the complexity of the solution to the optimization problem. Recall that each transformation  $\eta = [a, \vec{b}, \theta]$  consists of 4 parameters. We reduce the search from a four dimensional to

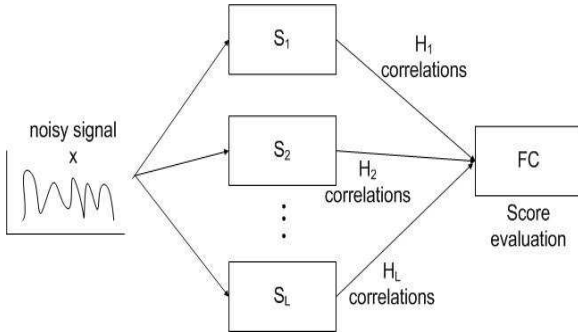


Fig. 1. Sensor network architecture.

two dimensional parameter space, by discretization of the parameter space with respect to scaling  $a$  and rotation  $\theta$  only. For each grid point  $(a_i, \theta_j)$ , we can then compute the block of scores  $\zeta(\eta_{ij})$ , where  $\eta_{ij} = [a_i, \vec{b}, \theta_j]$  corresponds to all possible translations  $\vec{b} = [b_1 \ b_2]^T$ , all at once. Observe that when  $(a_i, \theta_j)$  are fixed, the scores  $\zeta(\eta)$  are the result of correlating  $x$  with  $U([a_i, \vec{0}, \theta_j])\tilde{s}$ . From equation (7), we need the correlation matrices of  $x$  with each one of the atoms  $\phi_{\eta \circ \gamma_n}$ . However recall that these correlation matrices have already been computed in the first part of projecting  $x$  on the dictionary. These are finally combined according to equation (7) to yield the scores for all  $\vec{b}$ .

#### 4. DISTRIBUTED PATTERN DETECTION

This section now considers the distributed version of the parameter estimation algorithm, in the context of a visual sensor network as shown in Figure 1. The scenario consists of a number of sensors,  $S_i$ ,  $i = 1, \dots, L$ , which observe a noisy image  $x$ . In the proposed distributed algorithm, we assume that the initial full dictionary  $\mathcal{D}$  is split into  $L$  sub-dictionaries  $\mathcal{D}_i$ ,  $i = 1, \dots, L$  that are as disjoint as possible for increased resiliency in the detection step. Each sub-dictionary is assigned to a sensor. The  $i$ -th sensor projects  $x$  on its sub-dictionary by correlating  $x$  with the atoms that it has been assigned. For computational complexity reasons described above, atoms that differ only in the translation parameters, are assigned to the same sensor, and the dictionary partition (clustering) is therefore driven by the characteristics of centered atoms (i.e., atoms whose translation parameter  $\vec{b} = \vec{0}$ ). This step yields a set of correlation matrices, one for each centered atom assigned to sensor  $i$ :

$$C_j = x * \phi_j, \quad j \in \mathcal{D}_i, \quad (8)$$

where  $*$  denotes 2D correlation.

Next, the sensor  $S_i$  picks  $H_i$  atoms, whose correlation matrices gave the highest response, which means in our case the maximum element (in absolute value) within  $C_i$ . It com-

putes a crude approximation of the correlation matrix, in order to limit the size of the information to be transmitted. One straightforward solution consists in low-pass filtering of the matrices, which will highlight the regions of high correlation. We have chosen to fit one atom  $\psi_i$  from a very small dictionary, for each of the selected correlation matrix. Altogether, the sensor  $S_i$  sends out an atom index that identifies each of the  $H_i$  correlation matrices, along with the approximation of the correlation matrix, represented by the index of the fitting atom  $\psi_i$ .

Finally, the Fusion Center (FC) collects the total correlation representations from the sensors and proceeds to the score evaluation part using the formula (7). Note that only those atoms that were extracted from a sensor participate in the computation of the score formulas. Those atoms who are missing are simply ignored. However, since the sensors have represented only the high response atoms, we can expect that the missing atoms should not be the important ones.

Note that the partitioning of the full dictionary between the individual sensors is an important step in the system described above. We choose to distribute the initial full dictionary to the sensors so that the produced sub-dictionaries are as disjoint as possible. The motivation for having disjoint sub-dictionaries is to yield correlation matrices from very different atoms that capture the basic parts of  $x$ . Representing the correlation matrices of two very similar atoms results in sending almost the same information twice. By clustering the atoms of the initial dictionary into disjoint groups we tend to avoid this situation. In our algorithm, we partition the full dictionary into smaller parts using K-Means and the clustering is performed on the non-translated atoms (i.e.,  $\vec{b} = \vec{0}$ ).

#### 5. EXPERIMENTAL RESULTS

We provide experimental results that demonstrate the validity and the effectiveness of the proposed distributed algorithm. For the dictionary construction, we use the Anisotropic Refinement (AR) atoms which have been successfully used in image coding [3]. These are zero-mean edge-like atoms, which are obtained by a Gaussian function, and the partial second derivative of a Gaussian function in the orthogonal direction. We use 10 rotation angles in  $[0, \pi]$ , and 10 scales for each direction, which are logarithmically equi-distributed in the in-

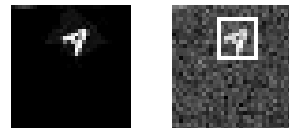


Fig. 2. The test image  $x$  in the noiseless and noisy case (at SNR=0 db) shown after correct recovery.

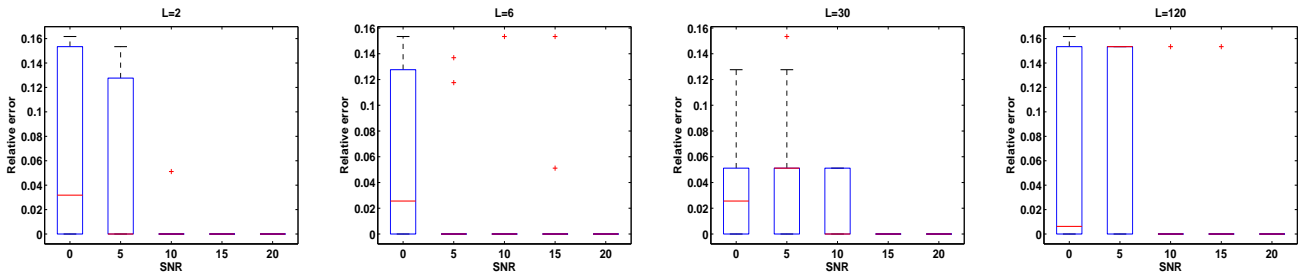


Fig. 3. Error behavior of the method for various number of sensors  $L$ .

interval  $[1, N]$ , where  $N$  is the size of the image. Thus, the full dictionary consists of 1000 atoms, which can take any spatial position on the image. Additionally, the small dictionary for approximating the correlation matrices at the sensors consists of 45 atoms, produced by 3 scales and 5 orientations, using the same generating function.

The images used for testing are depicted in Figure 2, where we are trying to detect  $s$  represented by the letter “A”. The exact transformation is  $\eta^* = [0.5, 9, 18, \pi/4]^T$  i.e.,  $s$  is scaled down by half, rotated by  $\pi/4$  and translated to position (9,18). We evaluate the detection performance for both the noiseless, and noisy scenarios.

$L$	2	3	4	6	10	15	30	60	120
$H$	280	160	110	50	28	12	4	4	1

Table 1. Minimum number of atoms per sensor, required for exact recovery in the noiseless case.

**Noiseless case** We first investigate the minimum number  $H$  of atoms per sensor that are required for correct recovery of  $\eta^*$  in the noiseless case. Table (1) reports the measured  $H$  for various number of sensors  $L$ . This experiment essentially studies the effect of missing atoms. If we use a smaller  $H$ , then the missing atoms are overwhelming and they lead to wrong  $\eta$ . Notice, that as the number of sensors increases, the number of atoms per sensor  $H$  whose correlation matrix is approximated, decreases. Indeed, in that case, the meaningful components of  $s$  are likely to be distributed in different sub-dictionaries.

**Noisy case** Then, we study the behavior of the algorithm in the presence of additive white Gaussian noise. Figure 3 illustrates the variation of the relative error  $e_r = \frac{\|\hat{\eta} - \eta^*\|_2}{\|\eta^*\|_2}$ , for various levels of SNR and for some representative values  $L$ . The estimated parameters are denoted by  $\hat{\eta}$ . Note that each experiment is repeated 10 times (for each random realization of noise) and the results are reported in boxplot notation. We have used  $H = 600$ , so each sensor keeps  $\lfloor \frac{H}{L} \rfloor$  number of atoms. We can observe that the algorithm is quite resilient to noise, even for a small number of sensors,  $L$ .

Note that in the case where the image content is more complex, the correlation matrices at the sensors are more complicated and a couple of atoms do not suffice for their approximation. One approach to alleviate this problem is to do a weak pre-detection locally at the sensors, where only the candidate transformations with the highest scores will be forwarded to the FC. This is to be investigated in future work.

## 6. CONCLUSIONS

We introduced a distributed algorithm for pattern detection in noisy images in the context of visual sensor networks. The proposed algorithm is based on the partitioning of a structured dictionary of geometric features. Each sensor computes an approximation of the correlation of the observed signal with the most prominent atoms in its dictionary. The pattern detection problem is casted into a parameters estimation problem and provides a generic method that is efficient even in noisy environments. Such a parameter estimation method may prove to be useful in numerous pattern detection application, from security schemes to tracking tasks.

## 7. REFERENCES

- [1] Dan Li, Kerry Wong, Yu Hen Hu, and Akbar Sayeed, “Detection, classification and tracking of targets in distributed sensor networks,” *IEEE Signal Processing Magazine*, vol. 19, no. 2, pp. 17–29, March 2002.
- [2] M. Li, Y. Wu, and S. Wu, “Multisensor distributed detection fusion based on wavelet domain filter,” in *Proceedings of ICSP*, 2004.
- [3] Figueras i Ventura R, Vandergheynst P, and Frossard P, “Low rate and flexible image coding with redundant representations,” *IEEE Transactions on Image Processing*, February 2006.
- [4] S. G. Mallat and Z. Zhang, “Matching pursuits with time-frequency dictionaries,” *IEEE Transactions on Signal Processing*, vol. 41, no. 12, pp. 3397–3415, December 1993.
- [5] F. Mendels, P. Vandergheynst, and J-P Thiran, “Rotation and scale invariant shape representation and recognition using matching pursuit,” *Int. Conf. on Patt. Rec.*, vol. 4, pp. 326–329, 2002.