REDUNDANT IMAGE REPRESENTATIONS IN SECURITY APPLICATIONS

Philippe Jost, Pierre Vandergheynst and Pascal Frossard

Signal Processing Institute
Swiss Federal Institute of Technology, Lausanne, Switzerland
{philippe.jost,pierre.vandergheynst,pascal.frossard}@epfl.ch

ABSTRACT

To be efficient, data protection algorithms should generally exploit the properties of the media information in the transform domain. In this paper, we will advocate the use of non-linear image approximations using highly redundant dictionaries, for security algorithms. We show that a flexible image representation based on a multidimensional and geometry-based coding scheme, has precious attributes for security information embedding. Redundant expansions provide very good approximation properties, as well as an increased resiliency to coding noise, and a simple stream structure enables easy manipulations. This paper describes simple examples of image scrambling and watermarking applications, based on a Matching Pursuit image coder. It illustrates the very interesting potential of redundant decompositions for data protection and security applications.

1. INTRODUCTION

Digital media handling is nowadays very easy and popular, its distribution has also become simpler due to the rapid expansion of broadband networks. In this context, where intellectual property could be violated, new security methods have been developed to protect multimedia content distributors and creators. In most cases, the security layer is mixed with the image compression layer, and uses transform domain data.

Signal expansions using redundant dictionaries have interesting properties for compression, especially at low bitrate. Such decompositions have proven to be quite efficient in representing natural images. Since the first paper by Mallat and Zhang ten years ago [1], Matching Pursuit is gaining in popularity and lots of efforts has been invested in finding good dictionaries and fast search algorithms. Using redundant dictionaries generally allows more flexibility in the representation and stream manipulation. Additionally, the increased resiliency to coding noise can be advantageously used for information embedding. This paper illustrates the potential of redundant representations for data protection schemes, and presents simple applications of scrambling and watermarking based on a Matching Pursuit coding algorithm.

This paper is structured as follows. Section 2 is a short overview of the redundant image expansions and Matching Pursuit algorithm. Section 3 presents a simple protection algorithm by image scrambling, that takes benefit of the geometric properties of redundant dictionary. Section 4 discusses information hiding in redundant expansions, and emphasizes the potential of a Matching Pursuit coder for watermarking applications. Section 5 concludes the paper.

2. REDUNDANT IMAGE EXPANSIONS

Signal expansions using redundant dictionaries is a very active domain since the introduction of the Matching Pursuit algorithm by Mallat and Zhang in 1993 [1]. They have shown that such a greedy algorithm converges exponentially in finite dimension, and thus provides a good approximation to a difficult combinatorial problem. The excellent paper of Gribonval and Nielsen [2] presents the main results in the research field during the last decade.

In general, a redundant expansion of a function f in a Hilbert space $\mathcal{H}$ is weighted sum of basis functions, also called atoms which are also functions lying in $\mathcal{H}$. The dictionary $\mathcal{D}$ is the overcomplete set of all atoms, and can be written as $\mathcal{D} = \{g_{\vec{\gamma}}\}_{\vec{\gamma}\in\Gamma}$ with $\|g_{\vec{\gamma}}\| = 1$. Matching Pursuit is a greedy algorithm that iteratively approximates the signal. It chooses $g_{\vec{\gamma_n}}$ such that the projection coefficient with the last residual is maximal. The residual signal at step n is $\mathcal{R}^n f = \mathcal{R}^{n-1} f - < \mathcal{R}^{n-1} f | g_{\vec{\gamma_n}} > g_{\vec{\gamma_n}}$. The initial residual $\mathcal{R}^0 f = f$. Thus, the function f is decomposed as follows:

$$f = \sum_{n=0}^{N-1} \langle g_{\vec{\gamma_n}} | \mathcal{R}^n f \rangle g_{\vec{\gamma_n}} + \mathcal{R}^N f \qquad (1)$$

In the case of redundant expansion of natural images, the atoms are bi-dimensional functions. The dictionary used in our coder is composed of non-separable atoms that are built on Gaussian functions along the first direction and second derivative of Gaussian functions in the orthogonal direction [3]. Each atom is fully described by a set of pa-

This work has been partly supported by the Swiss National Science Foundation.

rameters γ_i: position, rotation and scale. They uniquely represent the index $\vec{\gamma}$ of the atom $g_{\vec{\gamma}}$.

3. IMAGE SCRAMBLING

Scrambling is a well-known technique to introduce disorder in digital data. Applied to images, it lowers the quality of the whole or part of the image. It often happens that some scrambling methods lead to images that are too distorted. We now present an algorithm that adaptively scrambles the encoded image. It introduces perturbation into the parameters of the atoms. A parameter γ_i of an atom can take any integer value from 0 up to γ_i^{max}. Let Γ_s be the set of all parameters. Let us define a squeezing function S introducing a perturbation p into the parameter γ_i as follows:

$$S(\gamma_i, p) = (\gamma_i + p + \gamma_i^{max} + 1) \mod (\gamma_i^{max} + 1) \quad (2)$$

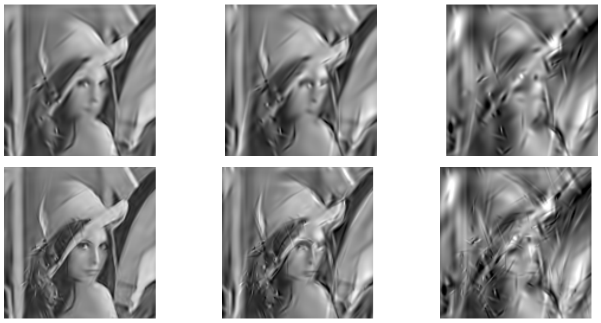
To ensure reconstruction at the decoding, p has to be the same as for the encoding. The number p is the result of a pseudo-random generator; to be able to decode correctly, it should depend only on an initial seed. Let R be such a pseudo-random number generator, the function $R(x)$ returns an integer between 0 and x. Algorithm 1 uniformly scrambles the parameters of the atoms:

Algorithm 1 Uniform Scrambling of atoms parameters

```
$\Gamma_s$ the set of parameters to modify for each atom.
To each $\gamma_i$ in $\Gamma_s$ assign $p_{\gamma_i}^{max}$ the maximal deviation.
for all atom $g_{\vec{\gamma_n}}$ in the redundant expansion do
    for all $\gamma_i$ in $\Gamma_s$ do
        $r = -p_{\gamma_i}^{max} + R(2 * p_{\gamma_i}^{max} + 1)$
        $\gamma_i = S(\gamma_i, r)$
    end for
end for
```

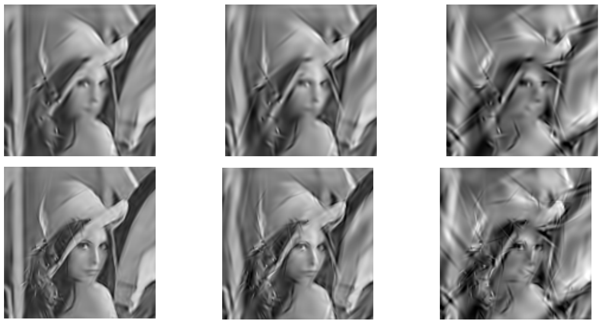
This flexible scrambling algorithm allows to add perturbation independently on position, scales or rotation parameters, or to any combination of them. Figure 1 shows images encoded with our Matching Pursuit encoder, after scrambling has been applied to the positions of the atoms. The rows contain the approximations for respectively 200 and 500 atoms. The first column are the images without scrambling and the others shows the results for different maximal allowable shifts on the positions. These images illustrate the fact that we can easily achieve different levels of scrambling. Even for small deviations, the visual impact is already important.

The scrambling can also be applied on the rotation parameters and the corresponding images are shown in Figure 2. Since the dictionary is built on 18 different angles, a unit shift of the rotation parameter corresponds to a physical rotation of 10 degrees of the atoms. Interestingly, we can see that small shifts bring minimal visual distortion to the images, and that the degradation is less important than for similar shifts on the position parameter.



(a) Original (b) Shift = 2 (d) Shift = 8

Fig. 1. Scrambling of the position parameters of the atoms, with different maximal shifts, for 200- and 500-atom expansions of the 128 x 128 *Lena* image.



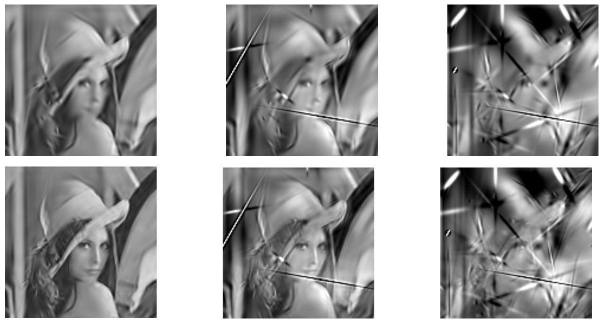
(a) Original (b) Shift = 1 (d) Shift = 4

Fig. 2. Scrambling of the rotation parameter of the atoms, with different maximal shifts, for 200- and 500-atom expansions of the 128 x 128 *Lena* image.

Finally, Figure 3 shows the results obtained with scrambling of the scale parameters. Scale scrambling is much more sensitive than noise on the position or rotation parameters. This is due to two main factors. First, the $\mod$ operators in the scrambling algorithm causes abrupt scale changes. When substituting a small scale parameter to a large scale one, annoying *lines* appear. Such big changes do not happen in the case where the parameters are cyclic, like the rotation parameters. Second, the norm of the atom is not conserved any more when scales change. The luminance of the image is therefore clearly degraded.

Figure 4 shows the evolution of the PSNR given the number of atoms in the image decomposition, for different scrambling strategies. At the beginning, the quality increases quasi normally since the first atoms are not extremely sensitive to coding noise. At a given stage, adding more scrambled atoms leave the error almost constant, since atoms do not contribute anymore to the true image representation. Figure 4 also confirms that the scale is the most sensitive parameter, and that the smallest degradations generally occur when scrambling the rotation parameter.

Based on the previous simple examples, it is possible to design more complicated scrambling strategies. It is straightforward to apply the previous algorithm only on some re-



(a) Original (b) Shift = 1 (d) Shift = 4

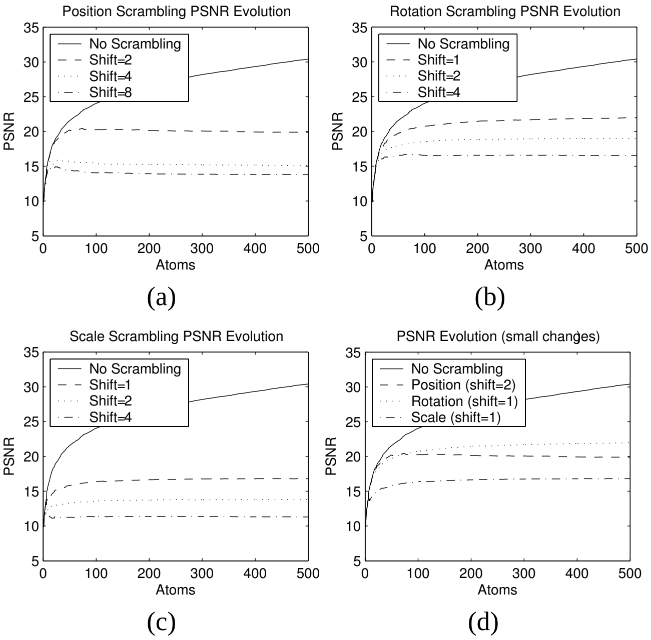
Fig. 3. Scrambling of the scale parameters of the atoms, with different maximal shifts, for 200- and 500-atom expansions of the 128 x 128 *Lena* image.

gions of the image and it could be of big interest, for given applications, to scramble only some regions of interest. Another possible application is to scramble only the last M atoms. The progressive order of the atoms within the Matching Pursuit stream guarantees that a reasonable quality of the image is already available with the first few atoms. Thus, a low quality image would be publicly available, and the high quality stream would be available only a subset of decoders, aware of the scrambling key.

Data protection by scrambling in the transform domain has the advantage to stay very simple. Starting from an existing decomposition, our algorithm adds a random value to a parameter. There exist very fast pseudo-random number generators. Thus, the complexity of our system is low enough to fit real-time constraints. The complexity of descrambling at decoder is the same as the one of the scrambling algorithm. Finally, note that the goal of this section is not to prove the robustness of the data protection scheme against potential attacks. This section however points out the benefits that can be offered by flexible streams, generated by Matching Pursuit, in the design of scrambling algorithms.

4. INFORMATION HIDING

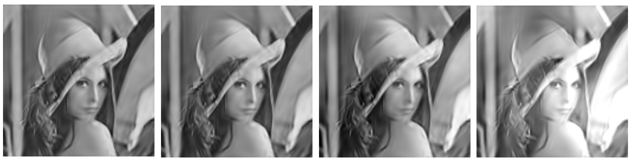
This section discusses the potential of redundant approximations in order to hide information within the coded image streams. Data hiding, in watermarking or steganography applications, relies on properly identifying redundancy in the image information, that can be used to hide a message without degrading the image representation. Redundant decompositions are natural candidates to hide messages, due to their inherent resiliency to coding noise. Geometrical redundancy is generally captured by atom indexes. In the same time, the importance of the coefficients in carrying information, is somehow smaller than in coding schemes based on orthogonal transforms. In these coders, the coefficients may even carry all the information: in a wavelet coder, the value of the coefficients and their position suffice to completely describe the image. Messages could thus be easily added into the redundant expansion of the image, where small and controlled variations in the parameter or coefficient values may be unnoticeable on the decoded image.



(a) (b)

(c) (d)

Fig. 4. PSNR evolution given the number of atoms and a scrambling strategy. (a) scrambling the positions, (b) scrambling the rotation, (c) scrambling the scaling, (d) comparison of the slightest scrambling for all illustrated methods.



(a) Original (b) a = 0.1 (c) a = 0.3 (d) a = 0.5

Fig. 5. Example of reconstructed images when adding randomly values to the projection parameter.

Figure 5 represents decoded images, after a uniform noise has been added to the projection parameters in the Matching Pursuit stream. The maximal magnitude of the noise, $|A|$ is proportional to the coefficient value, i.e., $|A| = a \times |c_{\bar{\gamma}}|$. Visually, when adding values that can be up to plus or minus 10 percents, it is difficult to decide which image could be the original one. On the two last columns, a lot of noise has been added, respectively up 30 and 50 percent of the absolute value of the projection. Even in those extreme cases, the user can still distinguish the content of the image. The projection parameter could thus be used to hide information without an important visual impact. Similarly, messages can be hidden in well chosen atom parameters, or in any combination of atom parameter and coefficients. It has been shown in the previous section that small variations of the rotation parameter, for example, bring a controlled degradation on the decoded image.

A simple algorithm is now presented, that hides a secret message $\mathcal{S}$ in the projection parameters. The list s_h is the binary version of $\mathcal{S}$. The projection coefficients are quantized and coded using DPCM; it gives a list of integers q_i. A pseudo-random number generator gives us a list r_i of numbers. Each *jump* in q_i holds one bit of $\mathcal{S}$ as described in algorithm 2.

Algorithm 2 Steganography Encryption

```
$h = 0$
**for** $i = 1$ to $N - 2$ **do**
   **if** $q_i > 1$ **then**
      **if** $q_i + r_i \neq s_h$ **then**
         $q_i = q_i - 1$ ; $q_{i+1} = q_{i+1} + 1$ ; $h = h + 1$
      **end if**
   **end if**
**end for**
```

In the here-above discussion, the information hiding process takes place after the image expansion have been generated, i.e., after all projection parameters and atoms have been found. The information hiding stage could take place during the search, in taking benefit of the redundancy of the decomposition. One could force the presence of an a priori list of atoms in the decomposition. The presence or the absence of the chosen atoms in the expansion represents the hidden watermark. Due to the properties of the Matching Pursuit algorithm and the overcomplete dictionary, these atoms stay indistinguishable from the other atoms in the expansion. The coding error they introduce in the stream is also diluted by successive iterations of Matching Pursuit, so that the hidden information cannot be discovered.

Algorithm 3 presents an algorithm making use of the previously described principle. It introduces atoms from a mark into the image. The decomposition of the mark $\{g_{\vec{\gamma_m}}\}_{\gamma_m \in \mathcal{M}}$ has to be known. The algorithm will introduce these atoms at given positions in the redundant image approximation of the image.

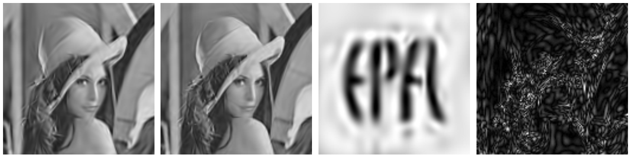

(a) No Mark (b) Watermarked (c) Mark (d) Difference

Fig. 6. Example of watermarking during search. The difference is normalized to a maximal value of 40.

Algorithm 3 Watermarking

```
$\{g_{\vec{\gamma_m}}\}_{\gamma_m \in \mathcal{M}}$ the redundant image approximation of the mark and $\mathcal{M} \in \Gamma$
$o_1$ and $o_2$ two positive numbers greater than 0.
Choose a set $r_j$ of $k$ integers between $o_1$ and $N - 1 - o_2$
$m = 0$, $a \approx 0.8$
$\mathcal{R}^0 f = f$
**for** $i = o_1$ to $N - 1 - o_2$ **do**
   **if** $i \in \{r_j\}$ **then**
      $g_{\vec{\gamma_i}} = g_{\vec{\gamma_m}}$ ; $c_i = ac_{i-1}$ ; $m = m + 1$
   **else**
      choose $g_{\vec{\gamma_i}}$ such that $< \mathcal{R}^{i-1} f | g_{\vec{\gamma_i}} >$ is maximal.
   **end if**
   $\mathcal{R}^i f = \mathcal{R}^{i-1} f - < \mathcal{R}^{i-1} f | g_{\vec{\gamma_i}} > g_{\vec{\gamma_i}}$
**end for**
```

Figure 6 shows the results obtained by applying algorithm 3. Atoms from the image to hide, i.e., the mark, have been added during the Matching Pursuit image expansion. The error they generate is compensated during the search and spread over the whole image. As the atoms to hide are placed deterministically in the stream, it is possible to recover the watermark. With some assumptions on the probabilities of an atom to take part in the sparse approximation, it is possible to compute the probability of such a list to be accidentally present. It is obvious that this probability can be made arbitrarily small when increasing the size of the dictionary. On the other hand, the complexity to find the sparse approximation depends directly on the size of the dictionary. It has to be noted finally that this section does not deal with the robustness of information hiding scheme, but rather shows the potential of redundant expansions for data embedding.

5. CONCLUSIONS

This paper advocated the use of redundant expansions in media security applications. Simple methods are proposed for data protection and information hiding in images encoded with a Matching Pursuit algorithm. We have shown that simple geometric manipulations on the atoms lead to interesting and promising results. The presented sketch of algorithms have however to be seen as possible paths to explore rather than real solutions. The flexibility of the Matching Pursuit streams, and their increased resiliency to coding noise, allows to foresee an interesting potential for redundant expansions in security algorithms.

6. REFERENCES

[1] S. Mallat and Z. Zhang, “Matching pursuit with time-frequency dictionaries,” *IEEE Transactions on Signal Processing*, vol. 41, no. 12, pp. 3397–3415, Dec 1993.

[2] R. Gribonval and M. Nielsen, “Approximation with highly redundant dictionaries,” in *Wavelets: Applications in Signal and Image Processing, Proc. SPIE’03*, 2003.

[3] R. Figueras i Ventura, P. Vandergheynst, and P. Frossard, “Highly flexible image coding using non-linear representations,” Tech. Rep. EPFL TR-ITS-2003.002, 1015 Lausanne, June 2003.