COMPRESSIVE SENSING FOR BACKGROUND SUBSTRACTION

COMPRESSED SENSING

Authored by

AMINE HAMMAMI
&
CHARLES DOGNIN

ENSAE ParisTech

2018

Table des matières

1 Introduction

Compressive Sensing for background subtraction is a technique used to efficiently detect foreground innovations in images. The idea is to subtract a background image x_b from a test image x_t to obtain a difference image x_d which only keeps the innovations we are interested in. Domain of applications can range from surveillance with tracking moving objects to teleconferencing and more. This article presents a method that only use compressive measurements (random projection of the whole image) of the test and background images to learn the innovations. It is then able to recover the location and silhouettes of the objects of interest, not the precise appearances. The main problematic of the article is twofold : how can we detect targets without reconstructing an image (ie in the random measurement space) and how can we reconstruct foreground images without using auxiliary images (ie only using the background and test random measurements) ?

The authors show that the silhouettes and positions can be recovered as a solution of a complex optimization problem : an orthogonal matching pursuit problem. To demonstrate the performance of their algorithms, the authors provide background subtraction results on SPC cameras and more traditional CCD cameras. We reproduce their results on two different dataset.

In order to detect the new objects, the authors use two distributions. The first one is a log euclidean-distance between many background training examples' compressive measurements and their mean which follows a Gaussian distribution. The second one is a log euclidean-distance between the test compressive measurements and the background compressive measurements. If the latter distribution crosses a certain threshold based on the first distribution then the algorithm detects a new object.

The other main contribution of the authors come from the adaptation of the background constraint. Indeed, the background may changes (new buildings, new trees, new car parked...) and the algorithm must be able not to flag them as innovations over time. As such, the author integrates an update rule based on two learning rates and a moving average of the compressive measurements.

2 The Compressive Sensing Theory

2.1 Sparse representation

Suppose we have an image $\mathbf{X}$ of size $N_1 \times N_2$. A vector x of size $N \times 1 (N = N_1 N_2)$ is then created from $\mathbf{X}$ by concatenating its columns in order. Suppose also that there exists a base $\Psi = [\psi_1, ..., \psi_N]$ giving a K sparse representation of x :

$$x = \sum_{n=1}^{N} \theta(n)\psi_n = \sum_{l=1}^{K} \theta(n_l)\psi_{n_l} = \Psi\theta$$

where $\theta(n)$ is the coefficient of the nth vector of the base and the coefficients indexed by n_l are the K non-zero values of the decomposition of x following this base. Let $||.||_p$ be the norm l_p where l_0 is the norm which counts the non-zero elements of θ. An image is said to be K-sparse if $||\theta||_0 = K$.

2.2 Random/Incoherent Projections

The K largest values of θ cannot be directly measured but the projection of the image x on a set of $M < N$ linear projectors $\Phi = [\phi'_1, ..., \phi'_M]'$ is measured :

$$y = \Phi x = \Phi\Psi\theta$$

where the vector y $(M \times 1)$ is the compressed sample and the matrix Φ $(M \times N)$ is the measure matrix. Given that $M < N$, the recuperation of the image x from y is under-determined. But, the sparsity assumption of x makes the recovery possible.
The CS theory states that
— the two bases must be incoherent
— M must be greater than $\mathcal{O}(K \log(\frac{N}{K}))$
Remark : incoherent means that Ψ can't give a sparsity representation of the rows of Φ. Given these two conditions, the recovery of the image x is now possible.

2.3 Signal Recovery via l_1 Optimization

We can use l_1 optimization problem (Basis Pursuit) as a recovery method :

$$\hat{\theta} = argmin||\theta||_1 \text{ knowing that } y = \Phi\Psi\theta$$

In this paper, we use Basis Pursuit Denoising :

$$\hat{\theta} = argmin||\theta||_1 + \frac{1}{2}\beta||y - \Phi\Psi\theta||_2^2$$

where $0 < \beta < \infty$. In our implementation, the recovered $\hat{\theta}$ is close enough to the original image/signal so we work directly with it.

3 CS for Background Subtraction

With background subtraction the purpose is to find the location, the shape and the appearance of any object using a known background.
Notation :
— x_b background image
— x_t test image
— x_d difference image
the difference image is simply the subtraction between the test and the background image respectively.

3.1 Sparsity of Background Subtracted Images

x_b and x_t differ only on the foreground's support. Since $x_d = x_t - x_b$, the difference is in the cardinality of $P = |\mathcal{S}_d|$ pixels $<< N$. The sparsity of the real world images is defined as $K_{\text{scene}} = K_b = K_t = (\lambda_0 \log N + \lambda_1)N$ where $(\lambda_0, \lambda_1) \in \mathbb{R}^2$. The difference image is also a real world image with a smaller support and its sparsity can be approximated by $K_d = (\lambda_0 \log P + \lambda_1)P$.
M_{scene} is the number of compressive samples. $M_{\text{scene}} = M_b = M_t \approx K_{\text{scene}} \log(\frac{N}{K_{\text{scene}}})$

and $M_d \approx K_d \log(\frac{N}{K_d})$. More generally $M_d < M_{\text{scene}}$, indeed the support of the difference image is smaller than the support of the background and the test image, so it needs smaller number of samples

3.2 The Background Constraint

Assume that we have multiple compressive measurements $y_{bi}(M \times 1, i = 1, ..., B)$ extracted from training background images x_{bi} (x_b is their mean). We approximate the distribution of y_{bi} as an i.i.d Gaussian distribution $y_{bi} \sim \mathcal{N}(y_b, \sigma^2 I)$.
We have $y_b = \Phi x_b$ and if any changes occur in the image that is not included in the background model, we obtain a compressive measurement of a test vector $y_t = \Phi x_t$. We can then obtain for the difference image, a background subtracted vector $y_d = y_t - y_b \sim \mathcal{N}(\mu_d, \sigma^2 I)(M \times 1)$.
The size of the vector y_b, M is greater than M_d but may not be greater than M_b and M_t, that's why it is generally very hard to reconstruct the background but x_b still follow the equation $y_b = \Phi x_b$.
In order to be robust against noise and small variations, we introduce the l_2 distance between y_{bi} and their mean y_b :

$$||y_{bi} - y_b||_2^2 = \sigma^2 \sum_{n=1}^{M} (\frac{y_{bi}(n) - y_b(n)}{\sigma})^2$$

If $M > 30$, the sum can be approximated using the central limit theorem $||y_{bi} - y_b||_2^2 \sim \mathcal{N}(M\sigma^2, 2M\sigma^4)$.

Proof Since $y_{bi} \sim \mathcal{N}(y_b, \sigma^2 I)$, we have $\frac{y_{bi} - y_b}{\sigma} \sim \mathcal{N}(0, I)$ and so $\frac{y_{bi}(n) - y_b(n)}{\sigma} \sim \mathcal{N}(0, 1)$.
The sum of $(\frac{y_{bi}(n) - y_b(n)}{\sigma})^2$ follow then a chi2 distribution with $\mathbb{E}(\sum_{n=1}^{M} (\frac{y_{bi}(n) - y_b(n)}{\sigma})^2) = M$ and $\text{Var}(\sum_{n=1}^{M} (\frac{y_{bi}(n) - y_b(n)}{\sigma})^2) = 2M$, we can now conclude that $\frac{||y_{\{bi\}} - y_b||_2^2}{\sigma^2}$ follow a chi2 distribution with the same mean and variance.
M is greater than 30 so by the central limit theorem and because the chi-squared distribution is the sum of M independent random variables with finite mean and variance, it converges to a normal distribution :

$$\frac{||y_{bi} - y_b||_2^2}{\sigma^2} \sim \mathcal{N}(M, 2M)$$
$$||y_{bi} - y_b||_2^2 \sim \mathcal{N}(M\sigma^2, 2M\sigma^4)$$

when we have a test images with new objects to identify, the distribution becomes $||y_t - y_b||_2^2 \sim \mathcal{N}(M\sigma^2 + ||\mu_d||_2^2, 2M\sigma^4 + 4\sigma^2||\mu_d||_2^2)$.

Proof Since $\frac{y_t - y_b}{\sigma} \sim \mathcal{N}(\frac{\mu_d}{\sigma}, I)$, $||y_t - y_b||_2^2 = \sigma^2 \sum_{n=1}^{M} (\frac{y_t(n) - y_b(n)}{\sigma})^2$ follow a non central chi-squared distribution. the non centrality parameter λ is :

$$\lambda = \sum_{n=1}^{M} \mathbb{E}(\frac{y_t(n) - y_b(n)}{\sigma})^2 = \sum_{n=1}^{M} \frac{\mu_d(n)^2}{\sigma^2} = \frac{||\mu_d||_2^2}{\sigma^2}$$

So, we can state now that :

$$\mathbb{E}[\sum_{n=1}^{M}(\frac{y_t(n)-y_b(n)}{\sigma})^2] = M + \frac{||\mu_d||_2^2}{\sigma^2}$$
$$\mathrm{Var}[\sum_{n=1}^{M}(\frac{y_t(n)-y_b(n)}{\sigma})^2] = 2M + 4\frac{||\mu_d||_2^2}{\sigma^2}$$

Finally, we can conclude that :

$$||y_t - y_b||_2^2 = \sigma^2 \sum_{n=1}^{M}(\frac{y_t(n)-y_b(n)}{\sigma})^2 \sim \mathcal{N}(M\sigma^2 + ||\mu_d||_2^2, 2M\sigma^4 + 4\sigma^2||\mu_d||_2^2)$$

Knowing that σ^2 scales the distribution, $\frac{1}{M} << 1$ and if $u << 1$ we have $1+u \approx e^u$, we obtain $\mathcal{N}(M\sigma^2, 2M\sigma^4) = M\sigma^2\mathcal{N}(1, \frac{2}{M}) = M\sigma^2(1+\sqrt{\frac{2}{M}}\mathcal{N}(0,1)) \approx M\sigma^2 \exp\{\sqrt{\frac{2}{M}}\mathcal{N}(0,1)\}$. Finally we can state that :

$$\log ||y_{bi} - y_b||_2^2 \sim \mathcal{N}(\mu_{bg}, \sigma_{bg}^2)$$

μ_{bg} and σ_{bg}^2 are learned using maximum likelihood techniques.

Maximum likelihood For a practical reason, let us note $X = log||y_{bi} - y_b||_2^2$. The likelihood is :

$$L(X_1, ..., X_B; \mu_{bg}, \sigma_{bg}) = \prod_{i=1}^{B} \frac{1}{\sqrt{2\pi}\sigma_{bg}} e^{-\frac{1}{2}(\frac{x_i-\mu_{bg}}{\sigma_{bg}})^2}$$
$$\log L(X_1, ..., X_B; \mu_{bg}, \sigma_{bg}) = \sum_{i=1}^{B} -\frac{1}{2}(\frac{x_i-\mu_{bg}}{\sigma_{bg}})^2 - \frac{B}{2}\log(\sigma_{bg}^2) + cste$$
$$\frac{\partial \log L}{\partial \mu_{bg}} = \sum_{i=1}^{B} \frac{1}{\sigma_{bg}^2}(x_i - \mu_{bg})$$
$$\frac{\partial \log L}{\partial \sigma_{bg}^2} = \sum_{i=1}^{B} \frac{1}{2\sigma_{bg}^4}(x_i - \mu_{bg})^2 - \frac{B}{2}\frac{1}{\sigma_{bg}^2}$$

so,

$$\frac{\partial \log L}{\partial \mu_{bg}} = 0 \Rightarrow \hat{\mu}_{bg} = \frac{1}{B}\sum_{i=1}^{B} x_i$$
$$\frac{\partial \log L}{\partial \sigma_{bg}^2} = 0 \Rightarrow \hat{\sigma}_{bg}^2 = \frac{1}{n}\sum_{i=1}^{B}(x_i - \hat{\mu}_{bg})$$

3.3 Object Detector based on CS

Before trying to reconstruct any object, it is useful to see if there is a difference between the test and the background images. To do so, the l_2 distance between y_t and y_b can also be approximated by :

$$\log ||y_t - y_b||_2^2 \sim \mathcal{N}(\mu_t, \sigma_t^2)$$

When the difference between the test image and the background image is small, $\sigma_t^2 \approx \sigma_{bg}^2$ but $\mu_t \neq \mu_{bg}$. So we state that if $|\log||y_t - y_b||_2^2 - \mu_{bg}| \geq c\sigma_{bg}$, a new object is detected in the test image.

3.4 Foreground Reconstruction

In this section, the BPDN algorithm with a fixed point continuation method and total variation optimization with an interior point method on the background subtracted compressive measurements were used.
The recovery of the appearance of the object is very difficult because the measurement contains also information about the background. So if the aim is to reconstruct the appearance of the object, we have to take enough compressive samples in order to reconstruct the test image and apply then a binary foreground image as a mask. Another way is also to reconstruct the background image, mask it and then add it into the foreground estimate.

Total variation optimization We define here the discrete gradient at location $i = (i_1, ..., i_n)$ as the vector with its n entries given by

$$((\nabla\theta)_i)_j = \theta_{i+e_j} - \theta_i$$

where $j = 1, ..., n$ and e_j is the j-th standard basis vector.
The total variation problem is :

$$\text{minimize}||\theta||_{\text{TV}} = ||\nabla\theta||_1 \text{ such that } ||\Psi\theta - y||_2 \leq \epsilon$$

3.5 Adaptation of the Background Constraint

It exists two types of changes in a background :
— drifts such as illuminations
— shifts sudden and major change
Remark : Detecting shifts is difficult because it has to handle complex operations (detecting and finding if it is interesting or not for instance).
The goal is to be able to find drifts and create an algorithm that is robust to these changes. To do so y_b needs to be continuously updated : when we obtain an estimate of the difference image $\hat{x}_d$, we can have then the compressive sample that need to be generated $\hat{y}_d = \Phi\hat{x}_d$. After that, we can determine the background estimate since $\hat{y}_b = y_t - \hat{y}_d$. Then the update equation is defined as follows :

$$y_b^{\{j+1\}} = \alpha(y_t^{\{j\}} - \hat{y}_d^{\{j\}}) + (1-\alpha)y_b^{\{j\}}$$

where α is a learning rate $\in (0, 1)$ and j is the time index.
This update does not take into account a sudden and major change in the background image such as new stationary targets. This change will be considered by this update as part of the foreground for all future frames. In order to fix this problem, we use this kind of updates :

$$y_{ma}^{\{j+1\}} = \gamma y_t^{\{j\}} + (1-\gamma)y_{ma}^{\{j\}}$$
$$y_b^{\{j+1\}} = \alpha(y_t^{\{j\}} - \hat{y}_e^{\{j\}}) + (1-\alpha)y_b^{\{j+1\}}$$

where y_{ma} is the simple moving average, γ is the moving average learning rate $\in [0, 1]$ and $\hat{y}_e = \Phi \hat{x}_{ma}$. Consider a shift in the image, at the beginning, this sudden change will be integrated in the foreground, then the value of the moving average will approximate the intensity of the pixel and won't be considered as a foreground but as a background.
the main disadvantage is that if the object to detect stays too long (depend on the value of the learning rate) without moving, it will be considered by the update as a background.

4 Limitations

The main limitation is related to embedded hardware limitations at the time of the article. In order to obtain directly a CS framework for imaging, the authors use an SPC (Single Pixel Camera). It differs from standard cameras by reconstructing an image using only a single optical photodiode along with a digital micromirror device (DMD) and by combining the sampling and compression into a single single linear (scalar product) measurement process. The current DMD arrays can change the way they pick the measurements at at most 40K per second which is too slow to process higher than 300x300 resolution background subtraction (with 1% compression ratios at 30fps). It means that at the time of writing of this article, the SPC embedded hardware was not efficient enough for this algorithm to work for real time high quality video object detection. Recently, researchers from the MIT Lab [2] came out with an ultrafast SPC camera which needs 50 times less measurements than traditional SPC camera. With such an efficient hardware, this limitation should be probably raised.

5 Experiments

5.1 Background subtraction with an SPC

Definition of SPC Instead of using multiple sensors to measure the incident light from the image to be taken, SPC measures a linear superposition of light intensities from different pixels using a single photo-diode. By measuring multiple such linear superpositions and exploiting the spatial smoothness in the image it reconstructs the original image

The author use an SPC for their experiments. They use the standard Mandrill test image, the foreground being a white rectangular patch. The resolution is 64x64 pixel. Both the background and foreground (y_b and y_t) were acquired using pseudo-random measurements, based on the Mersenne Twister algorithm. The traditional subtraction image (different from the difference image) is naturally obtained with $y_d = y_t - y_b$. The reconstruction of the background, test and difference images are obtained using the TV minimization algorithm. The FIGURE 1 shows the results. The top row is the background image, the second row the test image. The third row represents the traditional subtraction of both images and the fourth row represents the background subtraction (difference image) from compressive measurements. The columns corresponds to measurement rates $\frac{M}{N}$ of 50%, 5%, 2%, 1%, 0.5% from left to right. What is worth noting is that the innovation is recovered with the proposed algorithm with rates as low as 1%.

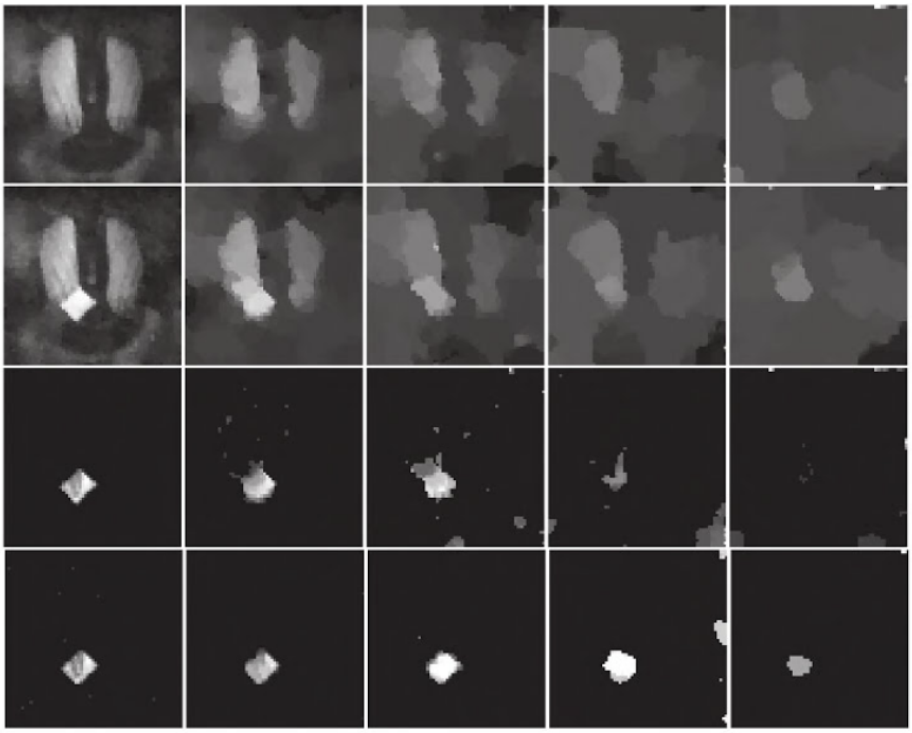

FIGURE 1 – First Experiment's results

5.2 The Sparsity Assumption

The author tests their assumption that the sparsity of natural images has the form $K = (\lambda_0 log(N) + \lambda_1)N$. Using various images from the Berkeley Segmentation Data Set and computing various wavelet approximations of various block sizes (from 2x2 to 256x256 pixels), the result of their experiments show that the necessary number of compressive samples is monotonic with the tile size. To approximate K they determined the minimum number of wavelet coefficients that results in a compression with -40 decibels distortion with respect to the original image. As a result, if the innovations in the image are smaller than the image, fewer compressive measurements are necessary to recover them.

5.3 Multi-view Ground Plane Tracking

The third experience is based on a surveillance video sequence of 300 frames. On the figure 2, the first two rows show images and background subtraction results using the compressive measurements. The right figures shows the detected points using the authors CS algorithm in blue and the detected points using full images in black. As a result, the background subtracted images are sufficient to generate detections that compare well against classical detection algorithm.

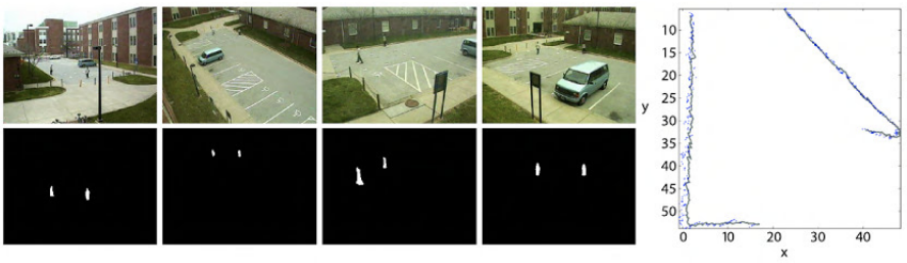

FIGURE 2 – Multi-view Ground Plane Tracking

5.4 Adaptation to Illumination Changes

In the third experiment the authors test the performances of the two background constraint adaptation algorithms. The quite noisy background subtraction result comes from the choice of the sparsifying basis (delta in that case). The first row shows the original images, the middle row corresponds to a background update using the second (best) algorithm and the last row corresponds to a background update using the first algorithm. The second algorithm with the moving average part allows the background constraint to keep track of the sudden change in illumination. As a result, the images are cleaner and keeps improving. The right figure correspond to the ROC AUC curve of the detection rate using both algorithms. We observe that for the same false negative rate, the second algorithm offers a much higher true positive rate.

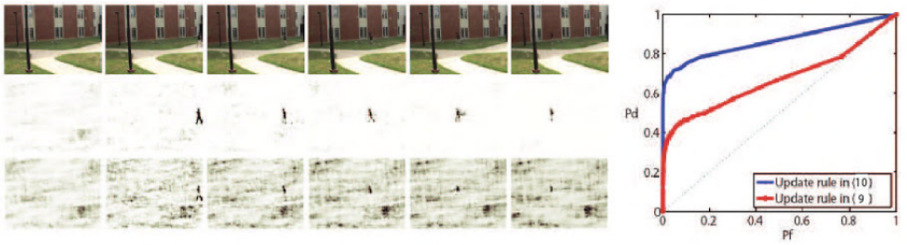

FIGURE 3 – Adaptation to Illumination Changes

5.5 Silhouettes vs. Differences Images

For their last experiment, they use a set up for a 3D reconstruction using compressive measurements. The left figures represent from left to right the ground truth and the background subtracted 3D image using CS. The result is not very satisfying because it integrates elements from the background. The right images show the 3D reconstruction of the ground truth.

6 The related more recent works

We decided to develop on the article 'Compressive Sensing Approaches for Autonomous Object Detection in Video Sequences'.
This paper focuses on the Bayesian method for compressed sensing. Typically,

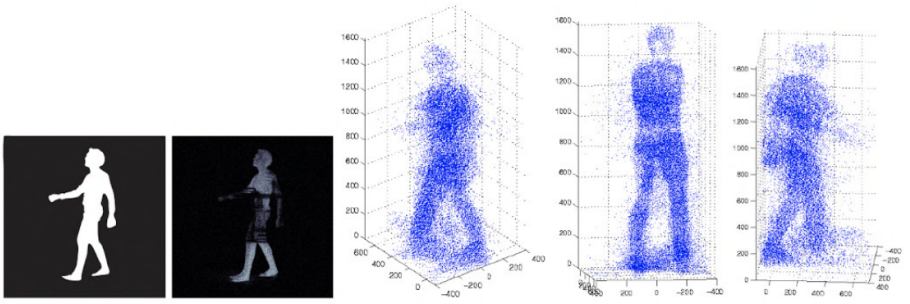

FIGURE 4 – Silhouettes vs. Difference Images

new objects take place only in a small part of the image so the difference mask x_d contains a lot of values close to zero. We estimate then x_d thanks to $y_d = \Phi x_d$. This paper introduces new method based on Bayesian method :
1) Bayesian compressive sensing : $y_d = \Phi x_d$ is reformulated as a linear regression model :

$$y_d = \Phi x_d + \xi$$

where $\xi_i \sim \mathcal{N}(0, \beta^{-1})$. The likelihood is expressed as follow :

$$p(y_d|x_d, \beta) = \prod_{n=1}^{M} \mathcal{N}(y_d(n); \Phi_n x_d, \beta^{-1})$$

where Φ_n is the n-th row of the matrix Φ.
Also, the prior distribution is imposed on all parameters :

$$p(x_d|\alpha) = \prod_{n=1}^{M} \mathcal{N}(x_d(n); 0, \alpha^{-1})$$

α and β are imposed to follow a gamma distribution :

$$p(\alpha) = \prod_{n=1}^{M} \Gamma(\alpha(n); a, b)$$

$$p(\beta) = \prod_{n=1}^{M} \Gamma(\beta(n); c, d)$$

where a, b, c, d are chosen uniform and close to zero. The Bayesian rule is then defined as :

$$p(x_d, \alpha, \beta|y_d) = \frac{p(y_d|x_d, \alpha, \beta)p(x_d, \alpha, \beta)}{p(y_d)}$$

the evidence $p(y_d)$ is intractable and so some approximation need to be used. Next we decompose the posterior as :

$$p(x_d, \alpha, \beta|y_d) = p(x_d|y_d, \alpha, \beta)p(\alpha, \beta|y_d)$$

the first term in the product can be expressed as :

$$p(x_d|y_d, \alpha, \beta) = \frac{p(y_d|x_d, \beta)p(x_d|\alpha)}{p(y_d|\alpha, \beta)}$$

We can conclude that $p(x_d|y_d, \alpha, \beta)$ is a Gaussian distribution with the parameters :

$$\Sigma = (\beta \Phi^T \Phi + A)^{-1}$$
$$\mu = \beta \Sigma \Phi^T y_d$$

where $A = diag(\alpha_1, ..., \alpha_M)$. The second term can be expressed as :

$$p(\alpha, \beta|y_d) = \frac{p(y_d|\alpha\beta)p(\alpha)p(\beta)}{p(y_d)}$$

and we need only to maximize $p(y_d|\alpha\beta) = \int p(y_d|x_d, \beta)p(x_d|\alpha)\,\mathrm{d}x_d$. The maximization with respect to α, β is given by this iterative process :

$$\alpha_i^{\text{new}} = \frac{\gamma_i}{\mu_i^2}$$
$$(\beta^{-1})^{\text{new}} = \frac{||y_d - \Phi\mu||_{l_2}^2}{s - \Sigma_{ii}\gamma_i}$$

where $\gamma_i = 1 - \alpha_i \Sigma_{ii}$ and s represents the sparsity. This process with the formula of Σ and μ converges to the optimal estimates.

7 Conclusion

The authors demonstrate in this article, that it is possible to reconstruct sparse innovations on a background scene with significantly less measurements than the traditional methods. Thus the authors provide a method to reduce the sampling cost and mitigate the communication and storage requirements for efficient object detection algorithms.

Références

[1] Danil Kuzin, Olga Isupova, and Lyudmila Mihaylova. Compressive sensing approaches for autonomous object detection in video sequences. In *Sensor Data Fusion : Trends, Solutions, Applications (SDF), 2015*, pages 1–6. IEEE, 2015.

[2] Guy Satat, Matthew Tancik, and Ramesh Raskar. Lensless imaging with compressive ultrafast sensing. *IEEE Transactions on Computational Imaging*, 3(3) :398–407, 2017.