

Single-frame Image Recovery using a Pearson type VII MRF[☆]

Ata Kabán^{1,*}, Sakinah Ali Pitchay¹

^a*School of Computer Science, The University of Birmingham, Edgbaston, Birmingham, B15 2TT, UK.*

Abstract

Compressive imaging and image super-resolution aim at recovering a high-resolution scene from its compressed or low resolution measurements. The main difficulty lies with the ill-posedness of the problem, and there is no consensus as to how best to formulate image models that can both impose smoothness and preserve the edges in the image. Here we develop a new image prior based on the Pearson type VII density integrated with a Markov Random Field model, which has desirable robustness properties. We develop a fully automated hyperparameter estimation procedure for this approach, which makes it advantageous in comparison with alternatives. Our recovery algorithm, although very simple to implement, it achieves statistically significant improvements over previous results in under-determined problem settings, and it is able to recover images that contain texture.

Keywords: single-frame super-resolution, compressed sensing, robust prior

1. Introduction

Image restoration and super-resolution aim to recover a high resolution scene from its low resolution measurements. The loss of resolution is often inevitable due to limitations of the camera source. In addition, the capturing process introduces additive noise. Depending on the number of low resolution frames of the scene available, we may talk about single-frame or multi-frame version of the problem. In both cases, most often the observed frames are scarce and noisy, which makes restoration an ill-posed problem. The single-frame version is necessarily under-determined too. Therefore, additional information is required to obtain an adequate solution. In a probabilistic model based framework, this additional information may be specified in the form of a prior distribution on the salient statistics that images are known to have. The two main characteristics are somewhat conflicting ones: local smoothness and the existence of edges. This makes the specification of a good image prior challenging.

Many prior models have been proposed in the literature, with no consensus, however. Gaussian Markov Random Fields represent a popular choice for its computational tractability. The Huber-MRF is robust, hence preferable, and it is considered to be a state-of-the-art approach [7, 12]. However, it requires two parameters to be tuned, and its performance depends on a good choice for these. Cross-validation could be a viable approach, but it is impractical for it is too time-consuming [12, 8, 10].

In this paper, we develop and investigate a perhaps less well-known, but quite convenient robust density, the Pearson type VII, formulated as Markov Random Field (MRF) for image recovery and super-resolution. The Pearson type VII has been used previously in situations where robust, heavy tail behaviour is required, such as in stock market modelling [5] and X-ray measurements [13], and for robust density estimation [14] as a more convenient and numerically stable alternative to the t-mixtures. In this work we exploit the robustness of this density to balance predominant smoothness of images with some allowance for edges or discontinuities.

2. The Pearson type VII density

The N -dimensional zero-mean Pearson type VII density is defined as the following [11]:

$$p(\mathbf{u}|\mathbf{C}, m) = \frac{\Gamma(m)}{\pi^{\frac{N}{2}} \Gamma(m - \frac{N}{2})} |\mathbf{C}|^{-\frac{1}{2}} [1 + \mathbf{u}^T \mathbf{C}^{-1} \mathbf{u}]^{-m} \quad (1)$$

where m is the degree of freedom that controls the degree of robustness that must satisfy $2m > N$, and N is the dimensionality of \mathbf{z} . It subsumes the Gaussian (when m approaches infinity) and the Student-t. For convenience, we will denote $\nu := 2m - 1$, so that the parameter ν is subject to positivity constraint only, and write the univariate Pearson type VII density (so, $N = 1$, $m = (\nu + 1)/2$) as:

$$p(u|\lambda, \nu) = \frac{\Gamma(\frac{1+\nu}{2}) \lambda^{\nu/2} (u^2 + \lambda)^{-\frac{1+\nu}{2}}}{\Gamma(\nu/2) \sqrt{\pi}} \quad (2)$$

where the parameter λ replaces \mathbf{C} and controls the width of the density, and ν is the degrees of freedom.

[☆]Extended version of the work presented at MLSP2010.

*Corresponding author

Email addresses: A.Kaban@cs.bham.ac.uk (Ata Kabán),
sxa814@cs.bham.ac.uk (Sakinah Ali Pitchay)

3. Image Models: Markov Random Fields

The main characteristic of any natural image is a local smoothness. That is, the intensities of neighbouring pixels tend to be similar. Any reasonable image model needs to be able to capture this property. Markov Random Fields (MRF) are well suited and widely used models that formalise this.

A very simple form of 1-st order MRF, previously employed with success for image recovery in e.g. [6, 7], is to condition each pixel intensity on its four cardinal neighbours in the following way. For any one pixel z_i define:

$$\begin{aligned} p(z_i | \mathbf{z}_{-i}) &= p(z_i | z_{4\text{neighb}(i)}) \\ &= p\left(z_i - \frac{1}{4} \sum_{j \in 4\text{neighb}(i)} z_j\right) \end{aligned} \quad (3)$$

where the notation \mathbf{z}_{-i} means all the pixels excluding the i -th, and the set of four cardinal neighbours of z_i was denoted as $4\text{neighb}(i)$. These are univariate probability distributions. We should mention that alternatives include the so-called total variation model, employed in e.g. [12], which is based on image gradients. The experimental comparison in [7] suggests that the model in eq.(3) and total variation behave in a very similar manner, the former being slightly superior though.

Using eq.(3), for an image \mathbf{z} of N pixels, the MRF represents the joint probability over all the pixels on the image — a multivariate probability distribution:

$$p(\mathbf{z}) = \frac{1}{Z} \prod_{i=1}^N p(z_i | \mathbf{z}_{-i}) = \frac{1}{Z} \prod_{i=1}^N p\left(z_i - \frac{1}{4} \sum_{j \in 4\text{neighb}(i)} z_j\right) \quad (4)$$

where $Z = \int d\mathbf{z} \prod_{i=1}^N Pr\left(z_i - \frac{1}{4} \sum_{j \in 4\text{neighb}(i)} z_j\right)$ is the normaliser (or partition function) of the MRF. This is independent of \mathbf{z} but depends on the hyperparameters of the constituent probability density building blocks.

The simplicity of (4) is also intuitively appealing. One can think of the difference between a pixel intensity and the average intensity of its neighbours, i.e. $z_i - \frac{1}{4} \sum_{j \in 4\text{neighb}(i)} z_j$, as a *feature*. However, the partition function Z is intractable to compute analytically, except for a very few specific cases. Therefore, approximations may be employed.

For notational convenience, it is handy to create the symmetric $N \times N$ matrix \mathbf{D} to encode the above neighbourhood structure, with the following entries:

$$d_{ij} = \begin{cases} 1 & \text{if } i = j; \\ -1/4 & \text{if } i \text{ and } j \text{ are neighbours;} \\ 0 & \text{otherwise.} \end{cases}$$

Then we may write the i -th feature in a vector form, with the aid of the i -th row of this matrix (denoted \mathbf{D}_i) as the following:

$$z_i - \frac{1}{4} \sum_{j \in 4\text{neighb}(i)} z_j = \sum_{j=1}^N d_{ij} z_j = \mathbf{D}_i \mathbf{z} \quad (5)$$

We now turn to instantiate the functional form of the probability densities that describe the shape of the likely values of these features. The Gaussian-MRF and the Huber-MRF represent existing choices, and we then move on to employ the Pearson type VII density in a similar role.

3.0.1. Gaussian MRF

The Gaussian MRF is frequently employed for convenience — see e.g. [6]. It has the following form:

$$p(\mathbf{z}) = \frac{1}{Z_G(\lambda)} \prod_{i=1}^N \exp\left\{-\frac{1}{2\lambda} (\mathbf{D}_i \mathbf{z})^2\right\} \quad (6)$$

where λ is the variance parameter. However, its lack of robustness is a known drawback since it tends to blur the edges of the image.

3.0.2. Huber MRF

The Huber density is defined with the aid of the Huber function:

$$H(u|\delta) = \begin{cases} u^2, & \text{if } |u| < \delta \\ 2\delta|u| - \delta^2, & \text{otherwise.} \end{cases} \quad (7)$$

where δ is a threshold parameter that needs to be adjusted. The Huber-MRF prior is then defined as:

$$p(\mathbf{z}) = \frac{1}{Z_H(\lambda, \delta)} \prod_{i=1}^N \exp\left\{-\frac{1}{2\lambda} H(\mathbf{D}_i \mathbf{z} | \delta)\right\} \quad (8)$$

where λ is similar to a variance parameter. The Huber-MRF has state-of-the-art performance, provided that its parameters are well chosen [12]. However, automating this choice in a principled way is not straightforward, and although we have been able to develop an approximate solution to estimating λ [10], the determination of the threshold parameter δ remains somewhat problematic since the pdf is not differentiable in δ .

4. The Pearson type VII MRF

We now propose to employ the Pearson type VII density with an MRF to provide a novel robust image model. One option would be to use its multivariate form as given in eq.(1) by encoding the neighbourhood structure in $\mathbf{C}^{-1} = \mathbf{D}^T \mathbf{D}$. However, this multivariate heavy-tail distribution would then be asserted on whole images (or possibly image patches) rather than tiny pixel neighbourhoods. We do not pursue this option here since our goal is to give non-zero probability to edges in the image, which requires a pixel-level modelling. Neighbourhood features that correspond to pixels that are situated at an edge may be thought of as spikes or outliers that our heavy-tail prior will account for, and this is what enables us to preserve the edges in the recovered image. To achieve this, we build up our

MRF prior from univariate Pearson VII densities, as the following:

$$p(\mathbf{z}) = \frac{1}{Z_P(\lambda, \nu)} \prod_{i=1}^N \{\lambda + (\mathbf{D}_i \mathbf{z})^2\}^{-\left(\frac{\nu+1}{2}\right)} \quad (9)$$

where $Z_P(\lambda, \nu) = \int d\mathbf{z} \prod_{i=1}^N \{\lambda + (\mathbf{D}_i \mathbf{z})^2\}^{-\left(\frac{\nu+1}{2}\right)}$ is the partition function, and this multivariate integral does not have an analytic form.

As with all MRF priors, the partition function may be neglected as long as we are interested in a *maximum a posteriori* estimate of \mathbf{z} with some known and fixed hyperparameters. However, the partition function does depend on the hyperparameters, hence for an automated estimation of these based on the model, the partition function must be approximated and taken into account. Notice that, in the case of a Pearson type VII MRF, the partition function is smooth w.r.t. both λ and ν — unlike the Huber MRF, which is non-smooth in δ . Hence, with a suitable analytic approximation of $Z_P(\lambda, \nu)$ this may be used for hyperparameter estimation.

4.1. Pseudo-likelihood approximation

We shall employ a pseudo-likelihood approximation to the partition function $Z_P(\lambda, \nu)$. Besides its simplicity, the pseudo-likelihood is known to enjoy consistency [2]. It consists of taking each $\mathbf{D}_i \mathbf{z}$ as if it were independent of $\mathbf{D}_j \mathbf{z}$, for all $j \neq i$ to break down the intractable multivariate integral into tractable univariate integrals. Thus, we have the following:

$$Z_P(\lambda, \nu) \approx \prod_{i=1}^N \int dz_i p(z_i | \mathbf{z}_{-i}) = \left\{ \frac{\Gamma(\nu/2)\sqrt{\pi}}{\Gamma\left(\frac{1+\nu}{2}\right)\lambda^{\nu/2}} \right\}^N \quad (10)$$

i.e. the inverse of the product of the normalising terms of the univariate Pearson type VII density building blocks.

Replacing this into the definition (9), we have the following approximate image model:

$$p(\mathbf{z} | \lambda, \nu) \approx \prod_{i=1}^N \frac{\Gamma\left(\frac{1+\nu}{2}\right)\lambda^{\nu/2} \left\{ (\mathbf{D}_i \mathbf{z})^2 + \lambda \right\}^{-\frac{1+\nu}{2}}}{\Gamma(\nu/2)\sqrt{\pi}} \quad (11)$$

We are now ready to employ this in the overall model for super-resolution, and use this to infer \mathbf{z} simultaneously with estimating our hyperparameters λ and ν .

5. The overall model for image recovery

5.1. Observation model

Denoting the vectorised high-resolution image by \mathbf{z} , as before, this is now a hidden variable. Instead, some low resolution version of it is observed. The degradation process will be taken as a linear transform, and we should note that, although this is a simplifying assumption, it

has worked well in many super-resolution application so far [6, 7, 12].

$$\mathbf{y} = \mathbf{W}\mathbf{z} + \boldsymbol{\eta} \quad (12)$$

where $\boldsymbol{\eta} \sim \mathcal{N}(\mathbf{0}, \sigma^2 \mathbf{I})$ is an additive noise. Equivalently, we can write $p(\mathbf{y} | \mathbf{z}) = \mathcal{N}(\mathbf{W}\mathbf{z}, \sigma^2 \mathbf{I})$, where \mathbf{y} is the observed version of the image, with $M < N$ pixels, and σ^2 is the observation noise variance. In single-frame super-resolution, the transform \mathbf{W} typically contains blur and down-sampling. In the multi-frame case we also have shift that varies between the observed frames and in that case \mathbf{y} is a concatenation of all the vectorised low resolution frames observed from the scene of interest. The single-frame problem is more challenging in that the system is under-determined (i.e. there are less observed pixel intensities than there are unknown ones).

5.2. Joint model

The overall working model consists of the observation model and the image prior model, so we have the pseudo-joint likelihood, assuming 0-mean σ^2 -variance i.i.d. Gaussian additive noise:

$$p(\mathbf{y}, \mathbf{z} | \mathbf{W}, \sigma^2, \lambda, \nu) = p(\mathbf{y} | \mathbf{z}, \mathbf{W}, \sigma^2) p(\mathbf{z} | \lambda, \nu) \quad (13)$$

6. MAP-based estimation in the model with Pearson type VII MRF

We will use the joint probability (13) as the objective to be maximised. Maximising this w.r.t. \mathbf{z} is also equivalent to finding the most probable image \mathbf{z} , i.e. the maximum a posteriori (MAP) estimate, since (13) is proportional to the posterior $p(\mathbf{z} | \mathbf{y})$. Equivalently, the negative log of this expression will be defined as our minimisation objective:

$$Obj(\mathbf{z}, \sigma^2, \lambda, \nu) = -\log[p(\mathbf{y} | \mathbf{z}, \sigma^2)] - \log[p(\mathbf{z} | \lambda, \nu)] \quad (14)$$

Plugging in the functional forms of the two density functions, we then minimise this w.r.t. \mathbf{z} and the hyperparameters in turn.

6.1. Estimating the most probable \mathbf{z}

The terms of the objective (14) that depend on \mathbf{z} are the following:

$$Obj_{\mathbf{z}}(\mathbf{z}) = \frac{1}{2\sigma^2} (\mathbf{y} - \mathbf{W}\mathbf{z})^2 + \frac{\nu+1}{2} \sum_{i=1}^N \log \{ \lambda + (\mathbf{D}_i \mathbf{z})^2 \} \quad (15)$$

The optimisation of (15) w.r.t. \mathbf{z} may be done employing any nonlinear optimiser, the objective is differentiable. We employed a conjugate gradient type method¹, which requires gradient information. The gradient is the following.

$$\nabla_{\mathbf{z}} Obj_{\mathbf{z}} = \frac{1}{\sigma^2} \mathbf{W}^T (\mathbf{W}\mathbf{z} - \mathbf{y}) + (\nu+1) \sum_{i=1}^N \mathbf{D}_i^T \frac{\mathbf{D}_i \mathbf{z}}{(\mathbf{D}_i \mathbf{z})^2 + \lambda} \quad (16)$$

¹We made use of the efficient implementation available from <http://www.kyb.tuebingen.mpg.de/bs/people/car1/code/minimize/>

6.2. Estimation of σ^2

Similarly writing out the terms of (14) that depend on σ^2 , taking derivative and solving, we get a closed form estimate for σ^2 :

$$\hat{\sigma}^2 = \frac{1}{M} \left(\sum_{i=1}^M (y_i - \mathbf{W}_i \hat{\mathbf{z}})^2 \right) \quad (17)$$

6.3. Estimation of λ and ν

The terms that depend on λ and ν :

$$\begin{aligned} Obj_{\lambda, \nu} &= N \log \Gamma \left(\frac{1+\nu}{2} \right) - N \log \Gamma(\nu/2) + \frac{N\nu}{2} \log \lambda \\ &- \frac{1+\nu}{2} \sum_{i=1}^N \log((\mathbf{D}_i \mathbf{z})^2 + \lambda) \end{aligned} \quad (18)$$

Both of these hyperparameters need to be positive valued. To ensure our estimates are actually non-negative, we parametrise the log probability objective (18) such as to optimise for the (\pm) square root of these parameters.

Taking derivatives w.r.t. $\sqrt{\lambda}$ and $\sqrt{\nu}$, we get:

$$\frac{d}{d\sqrt{\lambda}} Obj_{\lambda, \nu} = \sum_{i=1}^N \frac{\nu(\mathbf{D}_i \mathbf{z})^2 - \lambda}{((\mathbf{D}_i \mathbf{z})^2 + \lambda)\sqrt{\lambda}} \quad (19)$$

$$\begin{aligned} \frac{d}{d\sqrt{\nu}} Obj_{\lambda, \nu} &= \left[N \log \lambda - \sum_{i=1}^N \log((\mathbf{D}_i \mathbf{z})^2 + \lambda) \right. \\ &\left. + N\psi \left(\frac{1+\nu}{2} \right) - N\psi \left(\frac{\nu}{2} \right) \right] \sqrt{\nu} \end{aligned} \quad (20)$$

where $\psi(\cdot)$ is the digamma function. The zeros of these functions give us the estimates of $\pm\sqrt{\lambda}$ and $\pm\sqrt{\nu}$. Although there is no closed-form solution, these can be obtained numerically using any unconstrained nonlinear optimisation method. The square of these estimates give us the estimates of λ and ν respectively.

6.4. The algorithm

- Initialise the estimate $\hat{\mathbf{z}}$, e.g. as some combination of the solution of a Gaussian MRF and random noise.
- Iterate until convergence:
 - Estimate σ^2 using (17).
 - Perform iterations to update λ and ν in turn, using (19) and (20), keeping the current estimate $\hat{\mathbf{z}}$ fixed.
 - Perform iterations to update $\hat{\mathbf{z}}$ using (16)

Note that, the inner loops need not completely converge. It is sufficient to increase, not necessarily to minimise the objective at each intermediate step. However, we observed faster overall convergence by letting the inner iterations make more progress. The reason is probably that the overall objective is complex, with multiple local optima, while the individual updates break it down into simpler objectives in a greedy manner. Our MatLab implementation is available from http://www.cs.bham.ac.uk/~axk/Sakinah/PearsonMRF_code/

7. Experiments and results

We conducted experiments with both classical super-resolution (SR) matrices where \mathbf{W} comprises blur and down-sampling, as well as with random Gaussian compressive sensing (CS) matrices where \mathbf{W} has random entries sampled i.i.d. from a standard Gaussian. The latter is of interest in the light of new research in compressed sensing and signal processing [1, 3] directed towards devising hardware that can exploit some good theoretical properties of certain random matrices.

The observation data was generated starting from a ground truth real image via the matrix \mathbf{W} and additive noise. Working on synthetic data allows us to compare the recovered image against the ground truth, so that we can measure our recovery performance quantitatively.

7.1. Illustrative experiments

We start by demonstrating the working of our algorithm. Figure 1 shows an example of under-determined case, where we recover a 80×70 'cameraman' image (that is, $N = 5600$ pixels) from its $M = 4000$ randomly compressed measurements and additive noise of $\sigma = 0.5$.

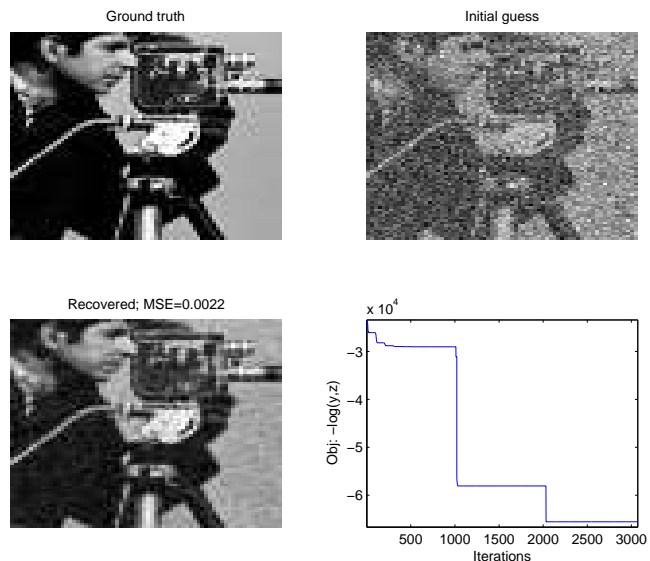


Figure 1: Example recovery of 'cameraman' (5600 pixels) from random linear mapping to 4000 pixels and additive noise with $\sigma = 0.5$.

We will be concerned with under-determined systems in this paper, however for the sake of completeness, we next show an overdetermined case as well, derived from a classical multi-frame super-resolution task, i.e. the transformation (or measurement) matrix consists of random shifts, Gaussian blur with point spread function set to 0.4 and down-sampling. Here we generated 18 low resolution images with a zoom factor of 3, so the overall system is overdetermined in this case. Figure 2 shows the ground truth, a straw-man recovery by averaged bi-cubic interpolation from the individual low resolution frames (which we use

as an initial guess to seed our algorithm in this experiment), and the obtained recovered image, along with the evolution of the objective over the iterations. It is easy to

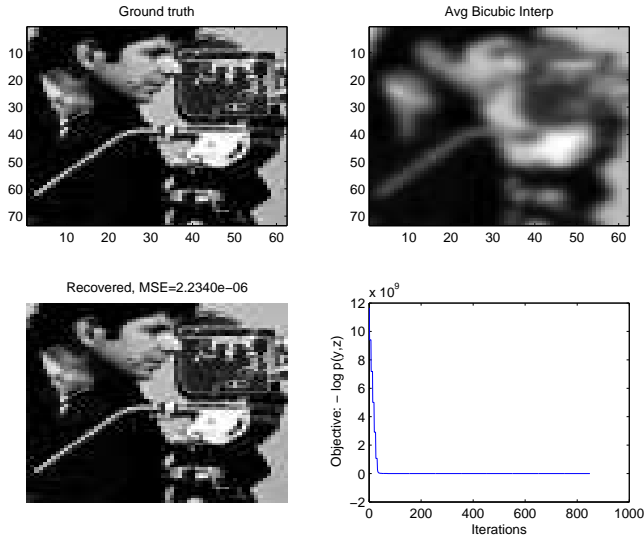


Figure 2: Example recovery from multiple (18) low resolution (zoom factor of 3) frames, which together represent an over-determined system.

see from the evolution of the objective function over the iterations (and the quality of recovered image) that having access to more observation frames makes the recovery task much easier.

In the remainder of the paper we will focus on recovery from a single frame, i.e. under-determined systems — such as recovering a 5600 pixels (80×70) image from a single $M \leq 4000$ -pixel frame. Quite obviously, without the specification of a prior, such a system would have infinitely many solutions, hence under-determined systems are much more reliant on the prior image model. In addition, the observations are subject to Gaussian additive noise, and this makes the recovery problem even harder.

In the case of CS-type \mathbf{W} , the noise standard deviations that we tested were $\sigma \in \{4 \times 10^{-5}, 0.5, 1, 2\}$. In the case of SR-type \mathbf{W} these values were divided by $0.8\sqrt{N}$ to make the signal-to-noise ratios roughly the same for the two matrix types. This is still relatively high noise, considering that we scaled the pixel intensities in the generating ground truth image to the interval $[-0.5, 0.5]$.

From each low-resolution data set, we then try to recover the ground truth image, and we assess the performance by measuring the mean square error between the recovered image $\hat{\mathbf{z}}$ and the ground truth \mathbf{z} — that is, $MSE = mean((\mathbf{z} - \hat{\mathbf{z}})^2)$.

7.1.1. Initialisation

Given that we optimise a non-convex objective, the initialisation scheme may impact the solution and the speed of convergence. Empirically we found that using CS-type matrices \mathbf{W} the quality of the solution is much less sensitive to initialisation than it is in the case of SR-type

matrices. The main issue, for SR-type, is to avoid starting it off from a neighbourhood of a local optimum. Therefore, in the case of under-determined problems we need to avoid using the output of a simpler superresolution recovery method as an initial guess, as it often turns out to lead to an unwanted local optimum. On the other hand, a completely random initialisation would take longer to converge. Based on these considerations, in all our experiments we adopted the following scheme. In experiments with SR-type matrices we initialise \mathbf{z} with the average between the minimum energy estimate ($\mathbf{W}^T \mathbf{y}$) and a random guess drawn from standard Gaussians. For CS-type matrices, since we did not experience any local optima issues, we opted to use a ridge regression to produce the initial guess (although other schemes that we tried did not make any noticeable difference). In both cases, we initialised the hyperparameters with $\nu = 10, \lambda = 1, \sigma^2 = 0.001$.

7.2. Assessment of the modelling power of the Pearson-type-VII image prior

Before diving into the assessment of our full algorithm, we switch off the automated hyperparameter estimation in this subsection. Here we assess the Pearson type VII based image model by comparing it with state-of-the-art alternative image priors when each of the competing prior models is supplied their best hyperparameters. For this purpose we select the best hyperparameters for each competing model based on the MSE with the ground truth. This, of course is not feasible in practice since the ground truth is not available, but it provides us information on what each model can achieve at its best. We will then move on to assess our automated hyperparameter estimation procedure against these idealised best results in section 7.3. We used CS-type matrices \mathbf{W} in this set of experiments. The competing methods are: Gaussian-MRF, a multivariate-Student-t based MRF that we also experimented with, and the Huber-MRF.

Figure 3 summarises the results obtained for the 'church' image² against varying noise conditions. Figure 4 shows the best recovery for the setting with $\sigma = 0.5$.

We see from the Figure 3 that the Pearson type VII based MRF model can achieve state-of-the-art performance in all noise levels tested, comparable to that of Huber-MRF, while the other priors tested perform worse. However, as already mentioned, for the Huber-MRF, a principled determination of both of its hyperparameters would not be straightforward. The next question is then, how does the automated hyperparameter estimation of our Pearson type VII based MRF prior compare to these hand-picked best results?

7.3. Assessment of the automated hyperparameter estimation procedure

Keeping the same experimental conditions set out in the previous section, Figure 5 shows the MSE achieved by

²<http://www.robots.ox.ac.uk/~vgg/research/SR/synthdata.html>

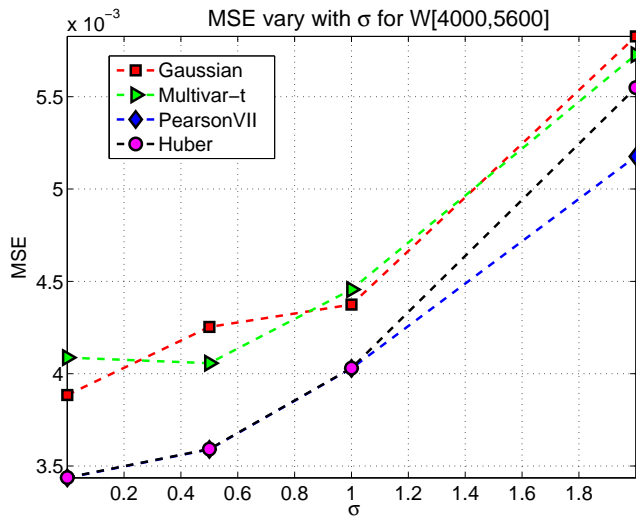


Figure 3: Comparative MSE performance for the under-determined system in progressively increasing noise conditions, using the best hyperparameter values (i.e. the value that produces the smallest MSE).

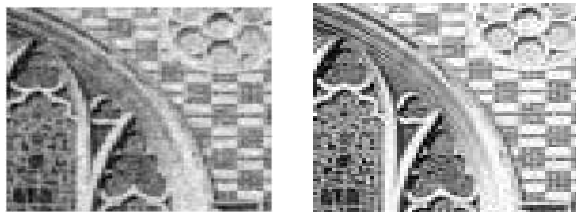


Figure 4: Left: Best recovered with manual tuning; Right: Ground truth.

our recovery algorithm that includes automated hyperparameter estimation, superimposed with the best manually picked results (of the same prior) from Figure 3 for reference. We see that, except for very high levels of noise, the agreement is remarkable. In fact, the MSE at the highest of the noise levels is still comparable with that of the best manual tuning of Huber MRF. Hence, we can conclude that, in these experiments the Pearson type VII based MRF is preferable as a fully automatic method.

In addition, it is notable the good agreement between the MSE and the values taken by the objective function. Note that the calculation of the MSE requires access to the ground truth image, while the Objective does not. Hence, the agreement between these two quantities represent further evidence for the appropriateness of our proposed model and automated estimation procedure. In other words, the best (or close to best) results in terms of agreement with the ground truth can be found by accessing the objective function independently of the ground truth. Indeed, the MSE of the recovered image selected solely on the basis of the objective function (MSE=0.0036) is not far off from the best MSE across the 10 repeats (MSE=0.0034).

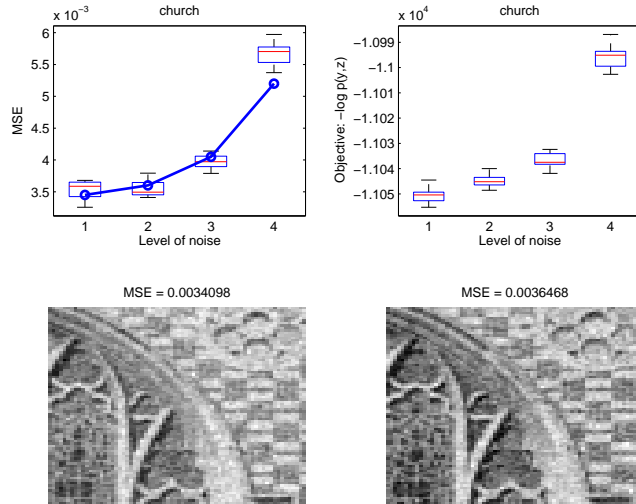


Figure 5: Comparing the performance of the fully automated Pearson type VII based MRF approach with the best result found by manual tuning of the hyperparameters. *Top left*: The distribution of MSE; *Top right*: The distribution of the values of the objective function; The boxplots represent 10 independent repeats where in each trial the additive noise and the transform \mathbf{W} were randomly drawn anew. *Bottom left*: Best result out of the 10 repeats with $\sigma = 0.5$, picked by lowest MSE. *Bottom right*: Best result out of the 10 repeats, picked by lowest values of the objective function. We see the MSE of the latter is very close to that of the former.

7.4. Comparison with Bayesian Compressed Sensing (BCS)

A recent technique that is also fully automated has been proposed in the field of compressed sensing [3], called Bayesian Compressed Sensing (BCS), which is based on the Automatic Relevance Determination (ARD) principle. It is interesting to compare our results with those of this method, since modelling-wise BCS is somewhat related to our approach in that it targets the solution of an under-determined linear system with the use of a probabilistic model and a prior. The prior they employ is the improper uninformative limit of a Student-t prior. To make the link, we note that the Pearson type VII density subsumes the Student-t if we set λ to $\lambda\nu$. However, the algorithmic solution of BCS differs from ours, and so does the authors choice to use the non-informative limit of the prior in [3, 4]. Hence, it is of interest to see the effect of these differences comparatively³. Unless stated otherwise, we will use the authors improved version of BCS from [4].

Figure 6 shows results obtained with BCS on the same data and experimental setup as we used in Figure 5. Although the performance seems to be not hugely different, it is still worse than what we had achieved. From the figure we see the MSE values over 10 repeats are higher, and the best MSE (0.0081) is still worse than the MSE of our choice based solely on our objective function in Figure 5 (MSE=0.0036). Looking at the recovered im-

³We used the authors implementation that is available from <http://people.ee.duke.edu/~lcarin/BCS.html>

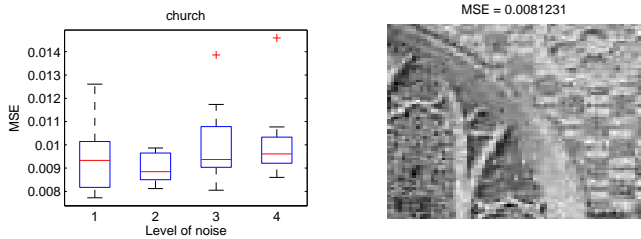


Figure 6: Results obtained with the Bayesian Compressed Sensing (BCS) algorithm of [3], to be compared with Figure 5 and 4. Left: The distribution of MSE over 10 independent repeats; Right: The best recovery across all these repeats. Observe the best MSE is still higher than the MSE of the Pearson-VII result picked cf. the best value of the objective function.

age (Fig.6) we notice that BCS tends to discard part of the edges in favour of more homogeneous areas, and this degrades performance when there is insufficient data. It should be pointed out that the same over-sparsifying problem associated with the use of the non-informative limit of Student-t priors in under-determined systems has been also reported in [15] in the context of logistic regression based classification problems. A general theory that explains the behaviour of promoting local strong homogeneity by log-priors that are non-smooth at zero can be found in [16]. It may be interesting to note that our log Pearson prior is actually smooth everywhere, but may exhibit a sharp curvature at zero when ν is small. Hence it is flexible enough to be able to promote local homogeneity without over-emphasising it.

We observed similar results on several different images in our experiments, with both CS-type and SR-type instantiations of \mathbf{W} . In Figures 7-8 we give results on another image ('castle') where we used SR-type \mathbf{W} and the same noise conditions as before. The observations made earlier are apparent again. From these results it seems that our algorithm is indeed able to recover a good quality high resolution image even when the image contains more texture.

7.4.1. Systematic experimental validation

Next we validate our findings by performing a battery of comparative experiments between our Pearson-type-VII based recovery algorithm and BCS. Before doing this on image data, we find it instructive to take the one dimensional sparse spiky signal used as a first test benchmark for compressed sensing algorithms (e.g. in [3] and [4]). It is known that the less sparse the signal is and the less observation measurements we have the more difficult the recovery problem. As in [4] (Fig.2 in [4]), we take signals of length $N = 512$ having 20 non-zero entries of ± 1 , random Gaussian compressive transform, we vary the number of observations, and measure the reconstruction error of the two recovery algorithms.

Figure 9 shows an example of recovery, where the number of observations are too few for BCS to cope with. In turn, our algorithm manages to recover the signal to a

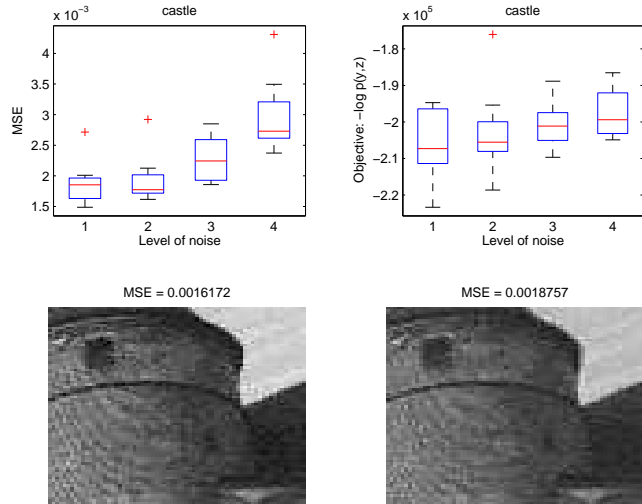


Figure 7: *Top left*: The distribution of MSE for our Pearson-VII based approach; *Top right*: The distribution of the objective functions values for our Pearson-VII based approach; The boxplots represent 10 independent repeats. *Bottom left*: Best result with Pearson-VII, out of the 10 repeats, picked by lowest MSE at noise level $\sigma = 0.5$. *Bottom right*: Best result with Pearson-VII, out of the 10 repeats, picked by lowest values of the objective function at noise level $\sigma = 0.5$.

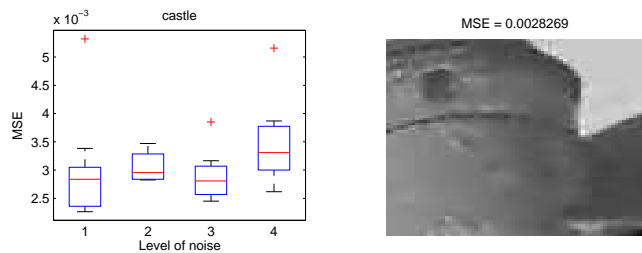


Figure 8: *Left*: The distribution of MSE for the BCS approach. *Right*: Best result with BCS, out of the 10 repeats, picked by lowest MSE at noise level $\sigma = 0.5$. The best recovery from BCS (MSE=0.0028) is higher than the pick that only uses the Objective (MSE=0.0018). BCS tends to discard part of the edges in favour of strong local homogeneity.

great extent.

In Figure 10 we give the full picture of this comparison for the recovery of the spike signal. As before, *BCS* refers to the improved version of BCS described in [4], and we also tried the previous version of this method, described in [3], which is referred to as *BCS_o* in the legend of our figure. We did not consider multi-task settings in this work though.

We see that our Pearson-VII is able to recover the signal from fewer measurements than BCS can. This also means that given the same number of measurements it can recover signals up to a larger number of spikes. This in turn implies in an image-reconstruction context that it can recover more edges, i.e. it can deal with more textured images.

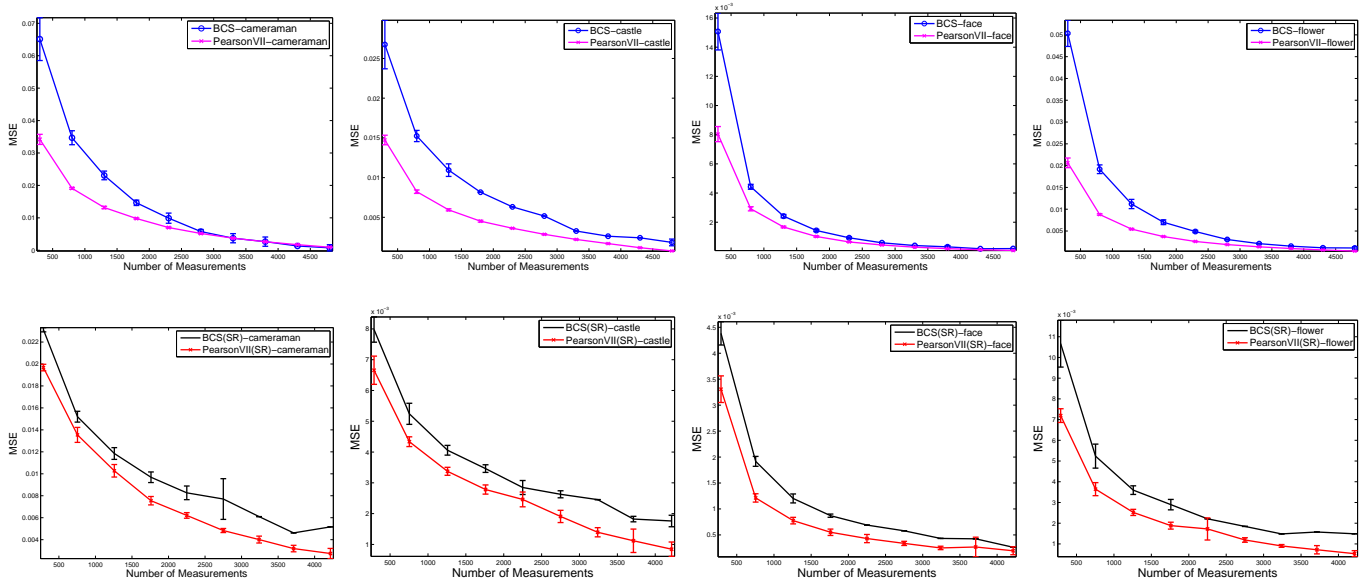


Figure 11: Comparative results on four different images, when \mathbf{W} is CS-type (upper row) and when \mathbf{W} consists of blur & downsampling (bottom row), and the number of observation is varied. The error bars represent one standard deviation from 10 independent repeats. We see the Pearson-based algorithm performs better than BCS in the under-determined regime.

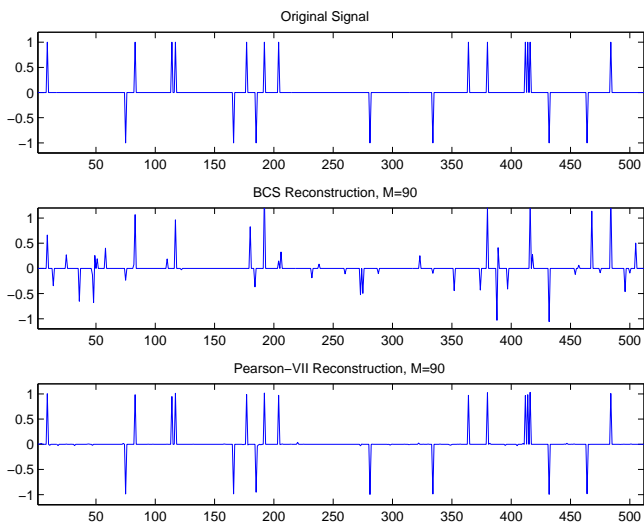


Figure 9: Reconstruction of the ‘spikes’ signal of length $N = 512$ having 20 non-zero entries from only $M = 90$ random compressive measurements. The first two subplots are reproduced from Fig.2 of [4] whereas the last subplot shows our recovery result.

Returning to image recovery, we now conduct experiments varying the number of measurements, and fixing the noise level to $\sigma = 0.005$. Figure 11 shows the comparative results obtained on four different natural images⁴ for both types of \mathbf{W} (CS-type, and SR-type). The images are: ‘cameraman’ (104×94 pixels), ‘castle’, ‘face’, and ‘flower’ (90×90 pixels each). We vary the number of compressive / low resolution observations down to 300 pixels. From

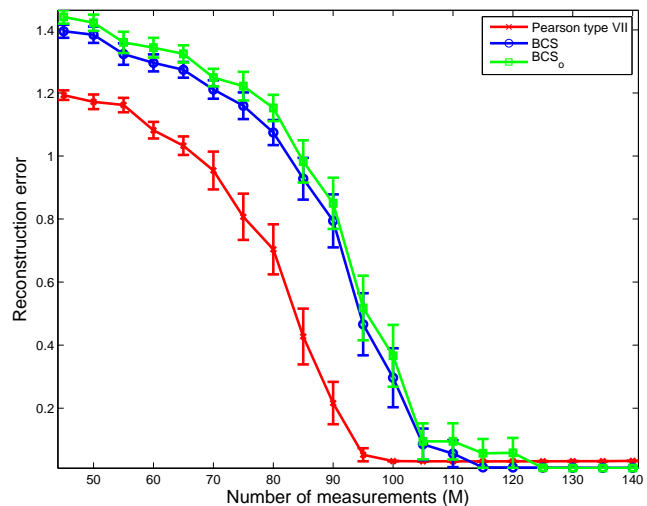


Figure 10: Comparison of recovering the ‘spikes’ signal from its CS measurements and additive noise of $\sigma = 0.005$ (the same setting as in Fig.2 of [4]). The error bars represent one standard error about the mean, from 25 independent repeats. We see that PearsonVII can recover the signal from fewer measurements than BCS.

these figures, in each experiment our algorithm achieves a statistically significant improvement over BCS in severely under-determined problem settings.

8. Conclusions

In this paper we formulated a new image prior based on Pearson type VII densities integrated with a MRF. Our main motivation has been to exploit the heavy tail property of this density, which indeed seems to be a good

⁴http://www.cs.bham.ac.uk/~axk/Sakinah/PearsonMRF_code/

way of preserving edges while imposing smoothness. The form of this prior has the additional advantage of allowing us to perform a fully automated hyperparameter estimation. Our recovery algorithm, although very simple to implement, it achieves statistically significant improvements over BCS in under-determined problem settings, and is able to recover more textured images than BCS can. Future work includes multi-task extensions of this approach, where the hyperparameters would be shared across tasks while other similarities may or may not be present.

References

- [1] M. A. Davenport, P. T. Boufounos, M. B. Wakin, and R. G. Baraniuk Signal Processing with Compressive Measurements, To appear in IEEE Journal of Selected Topics in Signal Processing, 2009.
- [2] S. Geman and C. Gragne. Markov random field image models and their applications to computer vision. Proc. Intl Congress of Mathematicians, pp 1496–1517, Berkeley, California, 1986.
- [3] S. Ji, Y. Xue, and L. Carin, Bayesian Compressive Sensing IEEE Trans. Signal Processing, vol. 56, no. 6, pp. 2346–2356, June 2008.
- [4] S. Ji, D. Dunson, and L. Carin, Multi-Task Compressive Sensing IEEE Trans. Signal Processing, vol. 57, no. 1, pp. 92–106, Jan. 2009.
- [5] Y. Nagahara. Non-gaussian distribution for stock returns and related stochastic differential equation, Asia-Pacific Financial Markets, vol. 3, no. 2, pp. 121–149, 1996.
- [6] R. C. Hardie, K. J. Barnard. Joint MAP Registration and High-Resolution Image Estimation Using a Sequence of Undersampled Images, IEEE Trans. Image Processing, Vol. 6, No. 12, Dec. 1997, pp. 621–633.
- [7] H. He and L.P. Kondi. MAP Based Resolution Enhancement of Video Sequences Using a Huber-Markov Random Field Image Prior Model, IEEE Conference of Image Processing, 2003, pp. 933–936.
- [8] H. He and L.P. Kondi, Choice of Threshold of the Huber-Markov Prior in MAP Based Video Resolution Enhancement, IEEE Electrical and Computer Engineering Canadian Conference, 2004.
- [9] A. Kabán and S.Ali Pitchay. Single-frame Image Super-resolution using a Pearson type VII MRF. Proc. IEEE International Workshop on Machine Learning for Signal Processing (MLSP 2010). August 29 - September 1, 2010, Kittila, Finland.
- [10] A Kabán. Estimation of the regularization parameter in Huber-MRF for image resolution enhancement. Technical Note. <http://www.cs.bham.ac.uk/~axk/huber.pdf>
- [11] K. Pearson, Mathematical contributions to the theory of evolution, xix: Second supplement to a memoir on skew variation. Philosophical Transactions of the Royal Society of London, Series A, Containing Papers of a Mathematical or Physical Character, vol. 216, no. 429–457, 1916.
- [12] L. C. Pickup, D. P. Capel, S. J. Roberts, A. Zissermann, Bayesian Methods for Image Super-Resolution, The Computer Journal, 2007.
- [13] P. Prévý. The use of Pearson VII distribution functions in X-ray diffraction residual stress measurement, Advances in X-Ray Analysis, vol. 29, pp. 103–111, 1986.
- [14] J. Sun, A. Kabán, and J. Garibaldi, Robust Mixture Modeling using the Pearson Type VII Distribution. Pattern Recognition Letters. Vol. 31, Issue 16, Dec 2010, pp. 2447–2454.
- [15] Y. Qi, T.P. Minka, R.W. Picard, Z. Ghahramani, Predictive automatic relevance determination by expectation propagation. In: Proc. ICML, 2004.
- [16] M. Nikolova. Local Strong Homogeneity of a Regularised Estimator. Siam Journal of Applied Mathematics, Vol. 61, No. 2, 2000, pp. 633–658.