

# Image edges resolved well when using an overcomplete piecewise-polynomial model

Michaela Novosadová  
Signal Processing Laboratory  
Brno University of Technology  
Brno, Czech Republic  
novosadova@phd.feec.vutbr.cz

Pavel Rajmic  
Signal Processing Laboratory  
Brno University of Technology  
Brno, Czech Republic  
rajmic@feec.vutbr.cz

**Abstract**—Used in the paper is an overcomplete piecewise-polynomial image model incorporating sparsity. The paper shows that using such a model, the edges in the image can be resolved robustly with respect to noise. Two variants of the proposed approach are both shown to be superior to the use of the classic edge detecting kernels. The proposed method is in turn also suitable for image segmentation.

**Index Terms**—image segmentation, smoothing, approximation, denoising, piecewise polynomials, sparsity

## I. INTRODUCTION

Segmentation of signals, and images in particular, is an important field of research with a huge number of applications. The individual segments of an image are distinguishable due to the fact that they typically exhibit (in a wide sense) a consistent character, and this character changes at the border of segments. Models beyond the individual segmentation algorithms most often rely on the mathematical modeling of what is understood under the vague term “character”; however, there are also approaches based on machine-learning segmentation from training data [1], [2].

Within the approaches based on a priori modeling, one can distinguish explicit and implicit types of models. In the “explicit” type, the signal is modeled such that it is a composition of sub-signals which can often be expressed analytically [3]–[8]. In the “implicit” type, the signal is characterized by features that are derived from the signal by using a suitable linear operator [9]–[12]. The above differences are analogous to the so-called “synthesis” and “analysis” approaches, respectively, recognized in the sparse signal processing literature [13], [14]. The two types of models are significantly different in the way the signal is treated; however, connections between them can be found, for example see [15].

In the paper, we accept the assumption that images consist of piecewise-polynomial patches, while it is typically sufficient to consider quadratic polynomials [16]–[20]. In that regard, the segment border is a place in the image where the polynomial characterization is subjected to a change, while within the segments, the respective representation stays steady.

Research described in this paper was financed by the National Sustainability Program under grant LO1401 and by the Czech Science Foundation under grant no.GA16-13830S. For the research, infrastructure of the SIX Center was used.

With noise preset in the image, the segmentation success naturally decreases. In the paper, we propose two methods based on a single model and show that both of them perform better in edge detection than standard methods do. Our work was inspired by [3], where the authors use a similar model, which they approximately solve using a greedy algorithm. Contrariwise, our approach relies on the convex relaxation of the sparsity measure [21]. As far as we know, the present paper is the first to use convex relaxation for modeling edges in images in connection with the piecewise-polynomial assumption.

In our recent works, we dealt with various segmentation algorithms for 1D signals, all of them following the piecewise-polynomial model [5], [22], [23]. In the present work, we generalize the 1D model to 2D, derive a new numerical solver and adapt the way the model output is processed. Note that in terms of the above categorization, the proposed model belongs to the “explicit” class.

## II. OVERCOMPLETE PIECEWISE-POLYNOMIAL MODEL

The images are modeled as being composed of non-overlapping polynomial patches. To better understand the way the two-dimensional polynomials are constructed, we first present the one-dimensional model and then generalize it. For more details on the 1D model, we refer to [23].

### A. One-dimensional model

In continuous time, a polynomial of degree  $K$  is representable as a linear combination of basis polynomials:

$$y(t) = x_0 p_0(t) + x_1 p_1(t) + \dots + x_K p_K(t), \quad t \in \mathbb{R}, \quad (1)$$

where  $x_k$  are the expansion coefficients in such a basis. The system  $\{p_k(t)\}_{k=0}^K$  is only required to form a basis, but additional properties such as orthogonality can sometimes be beneficial, see [23]. In a discrete setting, the individual elements of a polynomial signal are represented as

$$y[n] = x_0 p_0[n] + x_1 p_1[n] + \dots + x_K p_K[n], \quad n = 1, \dots, N. \quad (2)$$

Given  $\{p_k(t)\}_{k=0}^K$ , every signal given by (2) is uniquely determined by the set of coefficients  $\{x_k\}_{k=0}^K$ . The uniqueness

of such an expansion is broken when we introduce a time index also in these coefficients, allowing them to change in time:

$$y[n] = x_0[n]p_0[n] + x_1[n]p_1[n] + \dots + x_K[n]p_K[n]. \quad (3)$$

This makes the representation of the signal  $\{y[n]\}_{n=1}^N$  extremely overcomplete, nevertheless, this way of handling plays a key role in the model below. It will be convenient to write the latest relation in a more compact form, for which we introduce the notation

$$\mathbf{y} = \begin{bmatrix} y[1] \\ \vdots \\ y[N] \end{bmatrix}, \quad \mathbf{x}_k = \begin{bmatrix} x_k[1] \\ \vdots \\ x_k[N] \end{bmatrix}, \quad \mathbf{p}_k = \begin{bmatrix} p_k[1] \\ \vdots \\ p_k[N] \end{bmatrix} \quad (4)$$

for  $k = 0, \dots, K$ . After this, we can write

$$\mathbf{y} = \mathbf{p}_0 \odot \mathbf{x}_0 + \dots + \mathbf{p}_K \odot \mathbf{x}_K \quad (5)$$

with  $\odot$  denoting the elementwise (Hadamard) product. To convert (5) to the form involving the standard matrix–vector product, we define

$$\mathbf{P}_k = \text{diag}(\mathbf{p}_k) = \begin{bmatrix} p_k[1] & & 0 \\ & \ddots & \\ 0 & & p_k[N] \end{bmatrix}, \quad k = 0, \dots, K, \quad (6)$$

allowing us to write

$$\mathbf{y} = \mathbf{P}_0 \mathbf{x}_0 + \dots + \mathbf{P}_K \mathbf{x}_K \quad (7)$$

or, even more compactly,

$$\mathbf{y} = \mathbf{P} \mathbf{x} = [\mathbf{P}_0 | \dots | \mathbf{P}_K] \begin{bmatrix} \mathbf{x}_0 \\ \vdots \\ \mathbf{x}_K \end{bmatrix}. \quad (8)$$

Such a description of a signal of dimension  $N$  is superfluous in terms of the number of parameters; when the polynomials are fixed (which is assumed in the rest of the paper), the total number of  $(K+1)N$  parameters are used to characterize the signal. Nevertheless, assume now that  $\mathbf{y}$  is a *piecewise* polynomial and that it consists of  $S$  independent segments. Each segment  $s \in \{1, \dots, S\}$  is then described by  $K+1$  polynomials. In the above context, this can be achieved by letting *all* the vectors  $\mathbf{x}_k$  be constant within the particular segments, with a change at the segment border. Note that the positions of the segment change-points are unknown.

The described model motivates the following idea. The finite difference operator  $\nabla$  applied to the vectors  $\mathbf{x}_k$  produces vectors that are sparse—actually,  $\nabla \mathbf{x}_k$  has  $S-1$  nonzeros at maximum. Moreover, the nonzero components of all  $\nabla \mathbf{x}_k$  occupy the same positions across  $k = 0, \dots, K$ . This property justifies the use of the group-sparse model introduced later.

### B. Two-dimensional model

The natural extension to 2D is done through the Kronecker product of 1D vectors. For brevity, the derivation of the polynomial model will be introduced under the (natural)

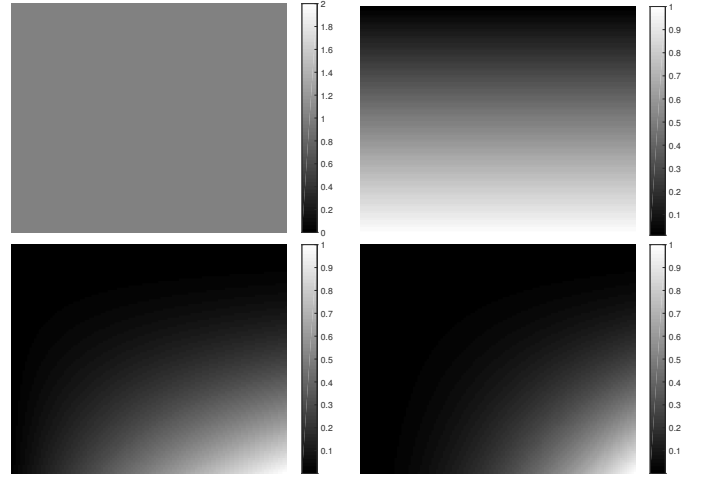


Fig. 1. Polynomials  $\mathbf{p}_{00}$ ,  $\mathbf{p}_{10}$ ,  $\mathbf{p}_{21}$ ,  $\mathbf{p}_{22}$  of size  $100 \times 120$ , generated from the standard basis of degree  $K = 2$  (i.e. from  $1, t, t^2$ ).

assumption that the degrees of polynomials in the vertical and horizontal directions are identical.

The 2D polynomials are formed by all the combinations

$$\begin{aligned} \mathbf{p}_{00} &= \mathbf{p}_{0v} \cdot \mathbf{p}_{0h}^\top \\ \mathbf{p}_{01} &= \mathbf{p}_{0v} \cdot \mathbf{p}_{1h}^\top \\ &\vdots \\ \mathbf{p}_{KK} &= \mathbf{p}_{Kv} \cdot \mathbf{p}_{Kh}^\top \end{aligned}$$

making altogether  $(K+1)^2$  polynomial basis images. The subscripts “v” and “h” distinguish the generating 1D polynomials in the vertical and horizontal direction, respectively. See Fig. 1 for examples. It is easy to show that when both the sets of vertical and horizontal prototypes form two respective orthonormal bases, then the new 2D system, too, is an orthonormal basis.

Let the image signal be  $\mathbf{y} \in \mathbb{R}^{M \times N}$  with entries  $y[m, n]$ ,  $m = 1, \dots, M$ ,  $n = 1, \dots, N$ . As in (5), the image is modeled as

$$\mathbf{y} = \mathbf{p}_{00} \odot \mathbf{x}_{00} + \mathbf{p}_{01} \odot \mathbf{x}_{01} + \dots + \mathbf{p}_{KK} \odot \mathbf{x}_{KK}, \quad (9)$$

where  $\mathbf{p}_{k\ell}$  are basis images of size  $M \times N$  as defined above, and  $\mathbf{x}_{k\ell}$  are matrices of parameters, all of the same size  $M \times N$ .

We need to identify an image with its vectorized form; this way of perception is necessary for us to be able again to express the model as a matrix–vector multiplication. Let the vectorization operator  $\text{vec}(\cdot)$  stack image columns, one after another, into a single column. The model (9) reads

$$\text{vec}(\mathbf{y}) = \mathbf{P}_{00} \cdot \text{vec}(\mathbf{x}_{00}) + \dots + \mathbf{P}_{KK} \cdot \text{vec}(\mathbf{x}_{KK}) \quad (10)$$

when we define  $\mathbf{P}_{k\ell} = \text{diag}(\text{vec}(\mathbf{p}_{k\ell}))$ . In short, convenient form,

$$\text{vec}(\mathbf{y}) = \mathbf{P} \mathbf{x} = [\mathbf{P}_{00} | \dots | \mathbf{P}_{KK}] \begin{bmatrix} \text{vec}(\mathbf{x}_{00}) \\ \vdots \\ \text{vec}(\mathbf{x}_{KK}) \end{bmatrix}. \quad (11)$$

### C. Basic idea

Following the just described image-generation model relying on  $\mathbf{P}$ , our method is based on the convex optimization problem presented in the next section. After the numerical solver has found an optimal solution to the problem, the solution is used in two possible ways to resolve the edges of the image. Both variants include the thresholding operator as the final step.

### D. Optimization problem

As in the above case, in the 2D setting, too, the parameterization coefficients  $\{\mathbf{x}_{k\ell}\}_{k,\ell}$  can be chosen such that they describe the signal perfectly and, at the same time, they are sparse under the difference operator  $\nabla$ . However, the difference operator has two forms now—the horizontal and the vertical one.

This, together with the assumption that the observed signal is corrupted by an i.i.d. Gaussian noise, motivates the following formulation of the denoising problem:

$$\hat{\mathbf{x}} = \arg \min_{\mathbf{x}} \{ \|\mathbf{L}_v \mathbf{x}\|_{21} + \lambda \|\mathbf{L}_h \text{reshape}(\mathbf{x})\|_{21} \} \\ \text{s.t. } \|\text{vec}(\mathbf{y}) - \mathbf{P}\mathbf{x}\|_2 \leq \delta. \quad (12)$$

In this optimization program, the operators  $\mathbf{L}_v$  and  $\mathbf{L}_h$  represent, respectively, the stacked vertical and horizontal differences, such that

$$\mathbf{L}_v = \begin{bmatrix} \nabla & \cdots & 0 \\ & \ddots & \\ 0 & \cdots & \nabla \end{bmatrix}, \quad (13)$$

with  $\nabla : \mathbb{R}^M \mapsto \mathbb{R}^{M-1}$  being the standard columnwise difference operator. This operator is repeated  $(K+1)^2$  times—the number of parameter maps—in (13). The operator  $\mathbf{L}_h$  is defined identically as  $\mathbf{L}_v$  in (13), but with the difference that  $\nabla$  is now the mapping  $\nabla : \mathbb{R}^N \mapsto \mathbb{R}^{N-1}$ . The operator  $\text{reshape}(\cdot)$  is used to emphasize that it is necessary to tweak the entries of  $\mathbf{x}$  such that the horizontal differences are computed using the linear operator  $\mathbf{L}_h$ .

The first term of (12) is the penalty. Piecewise constant  $\mathbf{x}_{k\ell}$  suggest that these vectors are sparse under the difference operation. The usual convex substitute of the true sparsity measure is the  $\ell_1$ -norm [21], [24]. Recall, however, that in addition to sparsity, the nonzero differences should appear at the same spots, as argued in Sec. II-A. Therefore, our optimization should contain a penalty that promotes *group* sparsity, whose convex substitute is the  $\ell_{21}$ -norm [25]. The  $\ell_{21}$ -norm acts on a matrix and is formally defined by

$$\|\mathbf{Z}\|_{21} = \|\mathbf{Z}_{1,:}\|_2 + \dots + \|\mathbf{Z}_{p,:}\|_2, \quad (14)$$

i.e. the  $\ell_2$ -norm is applied to the individual rows of  $\mathbf{Z}$  and the resulting vector is measured by the  $\ell_1$ -norm. In our case, the matrix is formed such that it is of size  $(M-1)(N-1) \times (K+1)^2$  and as such, the  $\ell_{21}$  penalty promotes sparsity across differentiated pixels, in line with the justification of the model. For details, see [5].

The scalar parameter  $\lambda$  optionally weights the vertical vs. horizontal differences. This parameter can differ from unity in the case that the user wants to emphasize horizontal over vertical structures and vice versa. Note, however, that this parameter should not change when the size of the image should “grow” proportionally in both directions, since in such a case both the penalties grow proportionally as well; only  $\delta$  should scale with the image size.

The second term in (12) is the data fidelity term. The Euclidean norm reflects the fact that gaussianity of noise is assumed. Consider however, that  $\delta$  accounts not only for the noise power, but also for model imperfections, since real images do not follow the piecewise-polynomial assumption exactly. As such, this parameter has to be tuned. Finally, the vector  $\hat{\mathbf{x}}$  contains the obtained optimizers.

## III. NUMERICAL SOLVER

The optimization problem (12) requires minimizing a sum of convex functions within a convex set of feasible solutions. Proximal splitting algorithms are suitable for such a class of tasks. We utilize the Condat proximal algorithm [26], which is able to solve (among more general formulations) problems of the type

$$\text{minimize } h_1(\mathbf{L}_1 \mathbf{x}) + h_2(\mathbf{L}_2 \mathbf{x}) + h_3(\mathbf{L}_3 \mathbf{x}), \quad (15)$$

over  $\mathbf{x}$ , where  $h_i$  are convex functions and  $\mathbf{L}_i$  are linear operators.

The connection between (15) and (12) is provided via assigning  $h_1 = \|\cdot\|_{21}$ ,  $h_2 = \lambda \|\cdot\|_{21}$ ,  $\mathbf{L}_1 = \mathbf{L}_v$ ,  $\mathbf{L}_2 = \mathbf{L}_h \text{reshape}(\cdot)$ . To fit the formulation (15), the feasible set is recast in the unconstrained form using the indicator function  $h_3 = \iota_{\{\mathbf{z} : \|\mathbf{y} - \mathbf{z}\|_2 \leq \delta\}}$ , where  $\iota_C$  denotes the indicator function of a convex set  $C$  [27]. Finally,  $\mathbf{L}_3 = \mathbf{P}$ .

The algorithm that solves (12) is described in Alg. 1. Therein, two operators are involved which have not been introduced yet: The operator  $\text{soft}_{\tau}^{\text{group}}$  performs “group” soft thresholding with the threshold  $\tau$ , where there are  $(M-1)(N-1)$  disjoint groups, each containing  $(K+1)^2$  coefficients. The projector  $\text{proj}_{B_2(\mathbf{y}, \delta)}$  finds the closest point in the  $\ell_2$ -ball  $\{\mathbf{z} : \|\mathbf{y} - \mathbf{z}\|_2 \leq \delta\}$  with respect to the input point. Both the operations are computationally cheap, for details see for example [5].

The convergence of the algorithm is guaranteed when it holds  $\xi \sigma \|\mathbf{L}_1^T \mathbf{L}_1 + \mathbf{L}_2^T \mathbf{L}_2 + \mathbf{L}_3^T \mathbf{L}_3\| \leq 1$ , where  $\|\cdot\|$  denotes the operator norm. The operator norm of the sum can be bounded such that  $\|\mathbf{L}_1^T \mathbf{L}_1 + \mathbf{L}_2^T \mathbf{L}_2 + \mathbf{L}_3^T \mathbf{L}_3\| \leq \|\mathbf{L}_1\|^2 + \|\mathbf{L}_2\|^2 + \|\mathbf{L}_3\|^2$ .

---

**Algorithm 1:** The Condat Algorithm solving (12)

---

**Input:**  $\mathbf{P}$ ,  $\mathbf{y}$ ,  $\delta$ **Output:**  $\hat{\mathbf{x}} = \mathbf{x}^{(i+1)}$ Set the parameters  $\xi, \sigma > 0$  and  $\rho \in (0, 2)$ .Set the initial primal variable  $\mathbf{x}^{(0)}$  and dual variables $\mathbf{u}_1^{(0)}, \mathbf{u}_2^{(0)}, \mathbf{u}_3^{(0)}$ .**for**  $i = 0, 1, \dots$  **do**

$$\begin{aligned} \bar{\mathbf{x}}^{(i+1)} &= \\ \mathbf{x}^{(i)} - \xi \left( \mathbf{L}_v^\top \mathbf{u}_1^{(i)} + \text{reshape}^\top \mathbf{L}_h^\top \mathbf{u}_2^{(i)} + \mathbf{P}^\top \mathbf{u}_3^{(i)} \right) \\ \mathbf{x}^{(i+1)} &= \rho \bar{\mathbf{x}}^{(i+1)} + (1 - \rho) \mathbf{x}^{(i)} \\ \mathbf{p}_1 &= \mathbf{u}_1^{(i)} + \sigma \mathbf{L}_v (2\bar{\mathbf{x}}^{(i+1)} - \mathbf{x}^{(i)}) \\ \bar{\mathbf{u}}_1^{(i+1)} &= \mathbf{p}_1 - \text{soft}_{\lambda}^{\text{group}}(\mathbf{p}_1) \\ \mathbf{u}_1^{(i+1)} &= \rho \bar{\mathbf{u}}_1^{(i+1)} + (1 - \rho) \mathbf{u}_1^{(i)} \\ \mathbf{p}_2 &= \mathbf{u}_2^{(i)} + \sigma \mathbf{L}_h \text{reshape}(2\bar{\mathbf{x}}^{(i+1)} - \mathbf{x}^{(i)}) \\ \bar{\mathbf{u}}_2^{(i+1)} &= \mathbf{p}_2 - \text{soft}_{\lambda}^{\text{group}}(\mathbf{p}_2) \\ \mathbf{u}_2^{(i+1)} &= \rho \bar{\mathbf{u}}_2^{(i+1)} + (1 - \rho) \mathbf{u}_2^{(i)} \\ \mathbf{p}_3 &= \mathbf{u}_3^{(i)} + \sigma \mathbf{P} (2\bar{\mathbf{x}}^{(i+1)} - \mathbf{x}^{(i)}) \\ \bar{\mathbf{u}}_3^{(i+1)} &= \mathbf{p}_3 - \sigma \text{proj}_{B_2(\mathbf{y}, \delta)}(\mathbf{p}_3 / \sigma) \\ \mathbf{u}_3^{(i+1)} &= \rho \bar{\mathbf{u}}_3^{(i+1)} + (1 - \rho) \mathbf{u}_3^{(i)} \end{aligned}$$

**return**  $\mathbf{x}^{(i+1)}$ 

---

We have

$$\begin{aligned} \|\mathbf{L}_1\|^2 &= \|\mathbf{L}_v\|^2 = \max_{\|\mathbf{x}\|_2=1} \|\mathbf{L}_v \mathbf{x}\|_2^2 \\ &= \max_{\|\mathbf{x}\|_2=1} \left( \sum_{k=0}^K \sum_{\ell=0}^K \|\nabla \text{vec}(\mathbf{x}_{k\ell})\|_2^2 \right) \\ &\leq \sum_{k=0}^K \sum_{\ell=0}^K \left( \max_{\|\mathbf{x}\|_2=1} \|\nabla \text{vec}(\mathbf{x}_{k\ell})\|_2^2 \right) \\ &\leq \sum_{k=0}^K \sum_{\ell=0}^K \|\nabla\|^2 = \|\nabla\|^2 (K+1)^2 \leq 4(K+1)^2 \end{aligned}$$

and the same holds for  $\|\mathbf{L}_2\|$ . As regards  $\|\mathbf{L}_3\| = \|\mathbf{P}\|$ , we have  $\|\mathbf{P}\|^2 = \|\mathbf{P}\mathbf{P}^\top\|$  and thus it suffices to find the maximum eigenvalue of  $\mathbf{P}\mathbf{P}^\top$ . Since  $\mathbf{P}$  has a multi-diagonal structure (cf. (11)),  $\mathbf{P}\mathbf{P}^\top$  is diagonal and thus it suffices to find the maximum on its diagonal. Altogether, the convergence is guaranteed when  $\xi\sigma (\max \text{diag}(\mathbf{P}\mathbf{P}^\top) + 8(K+1)^2) \leq 1$ .

## IV. EXPERIMENTS

The goal of the experiment was to establish whether the proposed problem (12) can help in indicating edges in noisy images better than the standard edge detection methods.

The experiment was designed similar to that of the evaluation in [3]. The starting signals for the test were the standard grayscale photographs taken from Matlab, the range of whose values was 0 to 255. The images were contaminated with zero mean i.i.d. gaussian noise.

1) *Coins, lower noise:* The first experiment was performed with the photo depicting a number of coins, see Fig. 2. The image is of size  $246 \times 300$  px. We superposed the noise over it with the standard deviation  $\sigma = 20$ .

As the ground truth edges we consider the result of the Sobel operator [28] applied to the original, clean image, subsequent normalization of the result such that the maximum absolute value is one, and finally the thresholding of the absolute values. The output, i.e. the binary map, is supposed to indicate the true positions of edges. We chose a threshold value of 0.15 for this particular image, which retained the important details at the sides of the coins. See Fig. 2.

As the standard way of edge detection we consider the application of the Sobel operator to the noisy image and thresholding (the normalized absolute values). In this process, we tuned the threshold to a compromise such that false edges do not appear too often while some of the details are still preserved in the coins. In our particular case, the threshold was 0.24.

The newly proposed approach was as follows: In the optimization problem (12), the standard basis of degree  $K = 2$  was used as the generating polynomial basis, corresponding to the examples in Fig. 1. The basis images  $\mathbf{p}_{00}, \dots, \mathbf{p}_{22}$  were used to form the model matrix  $\mathbf{P}$  as described in Sec. II-B. The model error  $\delta = 4000$  was allowed.

We let Alg. 1 run for 500 iterations with this setting (which took 3 minutes on a PC equipped with Intel i7 processor and 16 GB of RAM). This way we obtained the solution to (12), namely  $\hat{\mathbf{x}}$ . This vector is used in two distinct ways to get an estimate of the image edges.

The first way simply synthesizes the denoised image, i.e. it forms the image by  $\hat{\mathbf{y}} = \mathbf{P}\hat{\mathbf{x}}$  and reshapes the resulting column into a matrix of the size  $264 \times 300$ . Then, the above described convolution with the Sobel kernel and thresholding is applied as in the above cases. Again, the threshold was found (0.14) to produce the best-looking compromise between false alarms and edge preservation.

The second way exploits the multiple information contained in  $\hat{\mathbf{x}}$ , in contrast to using the single synthesized image alone. In particular, recall that the edges are (at least in the model) indicated in all the sets of parameters  $\mathbf{x}_{k\ell}$  in the same places, cf. (9). The space of  $\mathbf{x}$  is much larger than the image space and therefore the parameters could potentially be helpful in detecting the edges. We first compute the Sobel operator on all the individual parameter maps  $\hat{\mathbf{x}}_{k\ell}$ . Then we compute pixelwise  $\ell_2$ -norms along the  $(K+1)^2$  parameters. The output is a single matrix and is subject to thresholding (in our case with the threshold 0.12).

In total, we compare three methods (Sobel and that proposed in two versions) with the ‘‘ground truth’’. The results can be seen in Fig. 2.

2) *Coins, higher noise:* The same experiment but with greater noise is presented in Fig. 3 (the respective thresholds are 0.15, 0.29, 0.16 and 0.18;  $\delta = 6000$ ).

3) *Coins, higher noise, orthonormal basis:* The same image was processed using the orthonormal basis instead of the standard basis [23]. This time, the model error was set to  $\delta = 8000$  and the respective thresholds are 0.15, 0.29, 0.14 and 0.14. To reach convergence, we had to let Alg. 1 run for 10000 iterations. The experiment is presented in Fig. 4.

4) *Cameraman*: The cameraman image of the size  $256 \times 256$  (Fig. 5) was processed using the standard basis and with the model error  $\delta = 7000$  and the respective thresholds are 0.25, 0.29, 0.18 and 0.16. To reach convergence, we had to let Alg. 1 run for 500 iterations.

5) *Rice*: The rice image of the size  $256 \times 256$  (Fig. 6) was processed using the standard basis and with the model error  $\delta = 8000$  and the respective thresholds are 0.24, 0.4, 0.2 and 0.18. To reach convergence, we had to let Alg. 1 run for 500 iterations.

#### A. Discussion

It is clear that with both of our methods we were able to indicate the edges significantly better than the standard detection kernels. Note, however, that the price paid is vastly longer computation times. The detection based on parameter maps has beaten the simpler synthesizing approach.

### V. SOFTWARE AND DATA

The data and the implementation for Matlab can be downloaded from the URL [http://www.utko.feec.vutbr.cz/~rajmic/software/edge\\_detect.zip](http://www.utko.feec.vutbr.cz/~rajmic/software/edge_detect.zip). The archive is prepared in the sense of reproducible research; note, however, that the noise is generated each time the script is run, and that the orthonormal polynomials in experiment IV-3 are based on randomness too.

### VI. CONCLUSION

We have presented a model based on overcomplete piecewise-polynomial image assumption with group sparsity. The proposed model provides the basis for two simple methods for identifying image edges. We have shown that the two methods resolve image edges robustly with respect to noise, and that they exhibit a significant improvement over the performance of the standard Sobel edge detecting kernel.

### REFERENCES

- [1] F. Milletari, N. Navab, and S.-A. Ahmadi, "V-net: Fully convolutional neural networks for volumetric medical image segmentation," in *2016 Fourth International Conference on 3D Vision (3DV)*, Oct 2016, pp. 565–571.
- [2] K. Fritscher, P. Raudaschl, P. Zaffino, M. F. Spadea, G. C. Sharp, and R. Schubert, "Deep neural networks for fast segmentation of 3D medical images," in *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2016*, S. Ourselin, L. Joskowicz, M. R. Sabuncu, G. Unal, and W. Wells, Eds. Cham: Springer International Publishing, 2016, pp. 158–165.
- [3] R. Giryes, M. Elad, and A. M. Bruckstein, "Sparsity based methods for overparameterized variational problems," *SIAM journal on imaging sciences*, vol. 8, no. 3, pp. 2133–2159, 2015.
- [4] S. Shem-Tov, G. Rosman, G. Adiv, R. Kimmel, and A. M. Bruckstein, *Innovations for Shape Analysis*, ser. Mathematics and Visualization. Springer, 2012, ch. On Globally Optimal Local Modeling: From Moving Least Squares to Over-parametrization, pp. 379–405.
- [5] P. Rajmic, M. Novosadová, and M. Daňková, "Piecewise-polynomial signal segmentation using convex optimization," *Kybernetika*, vol. 53, no. 6, pp. 1131–1149, 2017. [Online]. Available: <https://www.kybernetika.cz/content/2017/6/1131>
- [6] G. Ongie and M. Jacob, "Recovery of discontinuous signals using group sparse higher degree total variation," *Signal Processing Letters, IEEE*, vol. 22, no. 9, pp. 1414–1418, 2015.
- [7] I. W. Selesnick, S. Arnold, and V. R. Dandam, "Polynomial smoothing of time series with additive step discontinuities," *IEEE Transactions on Signal Processing*, vol. 60, no. 12, pp. 6305–6318, Dec 2012.

- [8] B. Zhang, J. Geng, and L. Lai, "Multiple change-points estimation in linear regression models via sparse group lasso," *IEEE Transactions on Signal Processing*, vol. 63, no. 9, pp. 2209–2224, May 2015.
- [9] K. Bleakley and J.-P. Vert, "The group fused Lasso for multiple change-point detection," Jun. 2011, technical report. [Online]. Available: <https://hal.archives-ouvertes.fr/hal-00602121>
- [10] S.-J. Kim, K. Koh, S. Boyd, and D. Gorinevsky, " $\ell_1$  trend filtering," *SIAM Review*, vol. 51, no. 2, pp. 339–360, 2009. [Online]. Available: <http://dx.doi.org/10.1137/070690274>
- [11] I. Selesnick, "Sparsity-assisted signal smoothing (revisited)," in *Acoustics, Speech and Signal Processing (ICASSP), 2017 IEEE International Conference on*. IEEE, 2017, pp. 4546–4550.
- [12] R. J. Tibshirani, "Adaptive piecewise polynomial estimation via trend filtering," *Annals of Statistics*, vol. 42, no. 1, pp. 285–323, 2014.
- [13] M. Elad, P. Milanfar, and R. Rubinstein, "Analysis versus synthesis in signal priors," in *Inverse Problems 23 (200)*, 2005, pp. 947–968.
- [14] S. Nam, M. Davies, M. Elad, and R. Gribonval, "The cosparsity analysis model and algorithms," *Applied and Computational Harmonic Analysis*, vol. 34, no. 1, pp. 30–56, 2013.
- [15] M. Unser, J. Fageot, and J. Ward, "Splines are universal solutions of linear inverse problems with generalized TV regularization," *SIAM Review*, vol. 59, no. 4, pp. 769–793, 2017.
- [16] M. Elad, J. Starck, P. Querre, and D. Donoho, "Simultaneous cartoon and texture image inpainting using morphological component analysis (mca)," *Applied and Computational Harmonic Analysis*, vol. 19, no. 3, pp. 340–358, 2005.
- [17] K. Bredies and M. Holler, "A TGV-based framework for variational image decomposition, zooming, and reconstruction. part I," *Siam Journal On Imaging Sciences*, vol. 8, no. 4, pp. 2814–2850, 2015.
- [18] M. Holler and K. Kunisch, "On infimal convolution of tv-type functionals and applications to video and image reconstruction," *SIAM Journal On Imaging Sciences*, vol. 7, no. 4, pp. 2258–2300, 2014.
- [19] F. Knoll, K. Bredies, T. Pock, and R. Stollberger, "Second order total generalized variation (tgv) for mri," *Magnetic Resonance in Medicine*, vol. 65, no. 2, pp. 480–491, 2011. [Online]. Available: <http://dx.doi.org/10.1002/mrm.22595>
- [20] G. Kutyniok and W.-Q. Lim, "Compactly supported shearlets are optimally sparse," *Journal of Approximation Theory*, vol. 163, no. 11, pp. 1564–1589, 2011. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S0021904511001067>
- [21] D. L. Donoho and M. Elad, "Optimally sparse representation in general (nonorthogonal) dictionaries via  $\ell_1$  minimization," *Proceedings of The National Academy of Sciences*, vol. 100, no. 5, pp. 2197–2202, 2003.
- [22] M. Novosadová and P. Rajmic, "Piecewise-polynomial curve fitting using group sparsity," in *Proceedings of the 8th International Congress on Ultra Modern Telecommunications and Control Systems*, Lisbon, 2016, pp. 317–322.
- [23] M. Novosadová, P. Rajmic, and M. Šorel, "Orthogonality is superiority in piecewise-polynomial signal segmentation and denoising," *submitted*, 2018. [Online]. Available: <http://www.utko.feec.vutbr.cz/~rajmic/papers/NovosadovaRajmicSorel-OrthogonalitySuperiority.pdf>
- [24] E. J. Candes, M. B. Wakin, and S. P. Boyd, "Enhancing sparsity by reweighted  $\ell_1$  minimization," *Journal of Fourier Analysis and Applications*, vol. 14, pp. 877–905, 12 2008.
- [25] M. Kowalski and B. Torrèsani, "Structured Sparsity: from Mixed Norms to Structured Shrinkage," in *SPARS'09 – Signal Processing with Adaptive Sparse Structured Representations*, R. Gribonval, Ed. Inria Rennes – Bretagne Atlantique, 2009, pp. 1–6. [Online]. Available: <http://hal.inria.fr/inria-00369577/en/>
- [26] L. Condat, "A generic proximal algorithm for convex optimization—application to total variation minimization," *Signal Processing Letters, IEEE*, vol. 21, no. 8, pp. 985–989, Aug 2014.
- [27] P. Combettes and J. Pesquet, "Proximal splitting methods in signal processing," *Fixed-Point Algorithms for Inverse Problems in Science and Engineering*, pp. 185–212, 2011.
- [28] W. Burger and M. J. Burge, *Digital image processing: an algorithmic introduction using Java*. Springer, 2008.

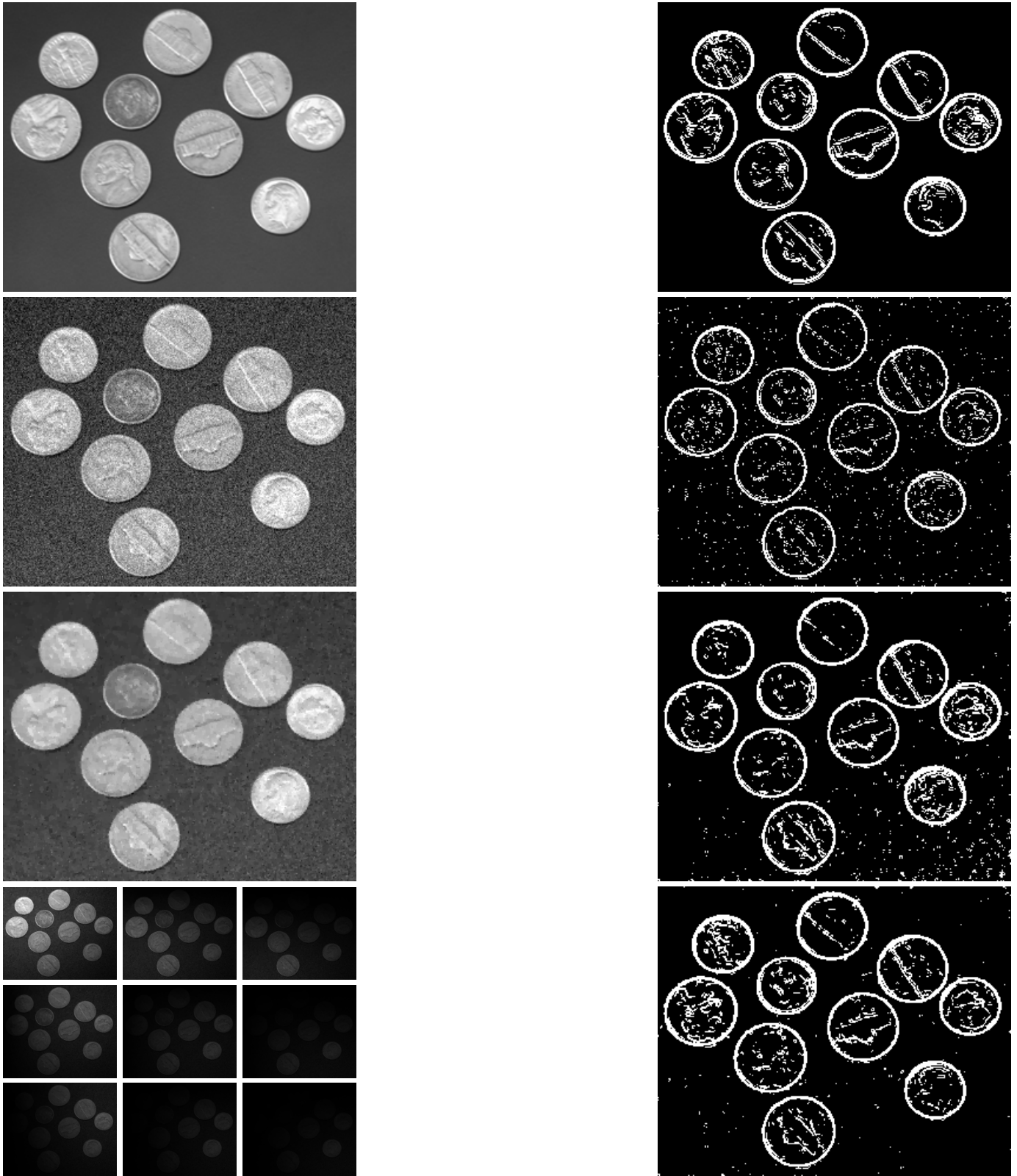


Fig. 2. Results of the edge-detecting experiment. The left column shows the images, from top to bottom: original; noisy ( $\sigma = 20$ ); synthesized by taking  $\hat{y} = P\hat{x}$  (i.e. denoised); parameter maps  $\hat{x}_{00}, \hat{x}_{01}, \dots, \hat{x}_{22}$  (from left to right and from top to bottom, in absolute values and jointly scaled to fit the black–white range). The right column shows the respective edge maps obtained by thresholding gradient images. Only the last image (which gives best results) is constructed from the parameter maps  $\hat{x}_{k\ell}$ .

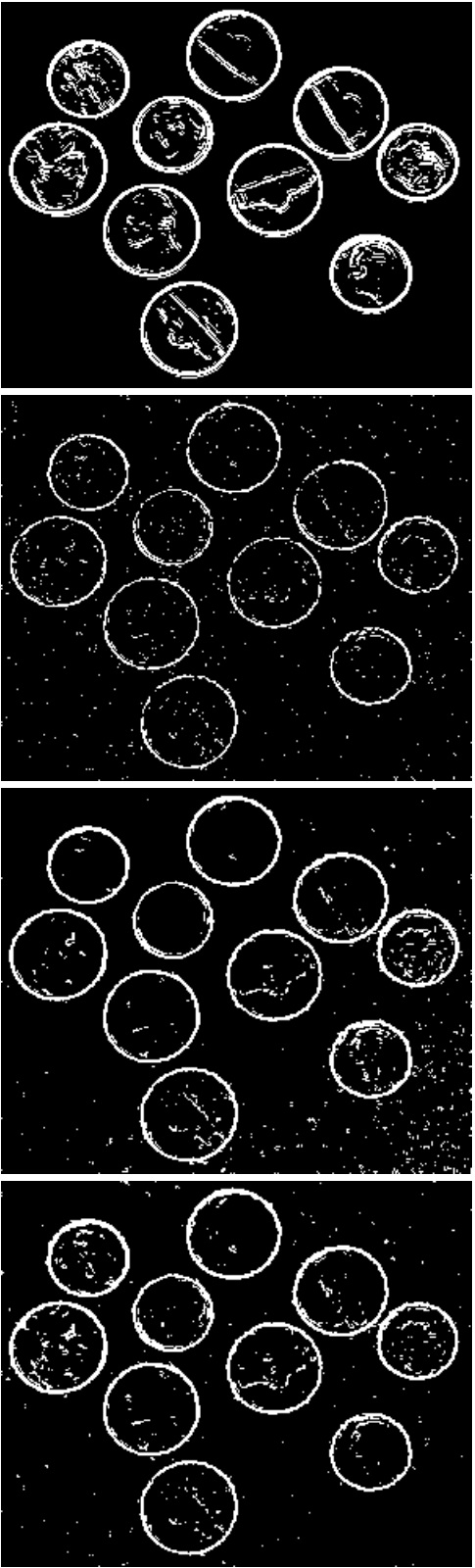
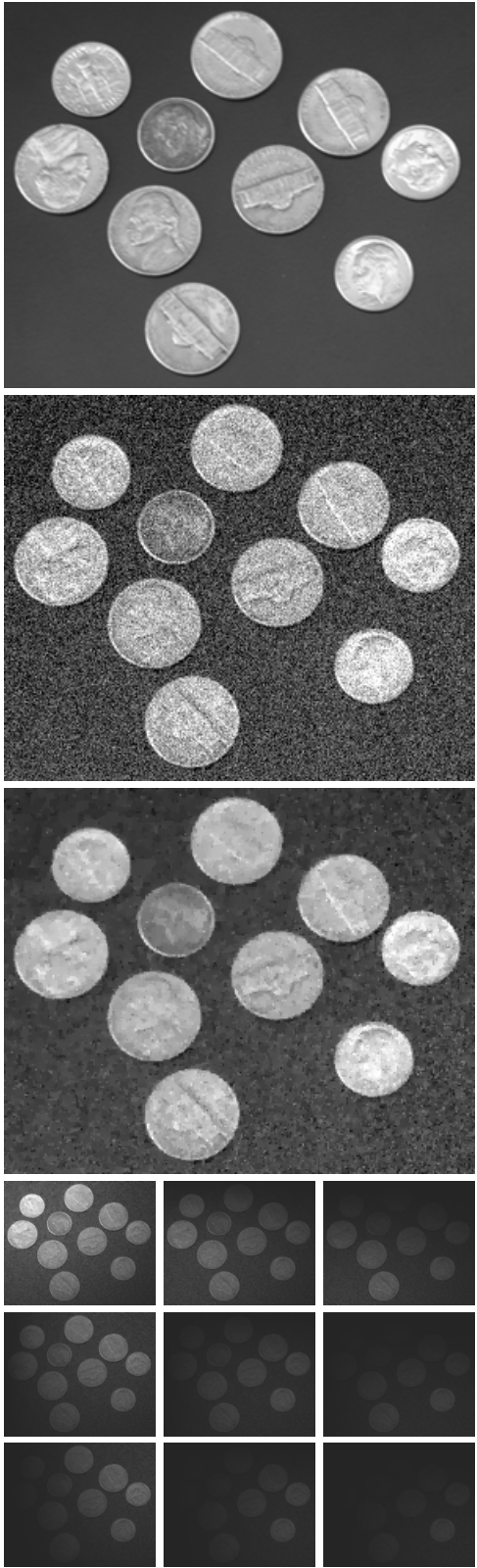


Fig. 3. Results of the experiment, similar to the one presented in Fig. 2, but for noise standard deviation  $\sigma = 30$ .

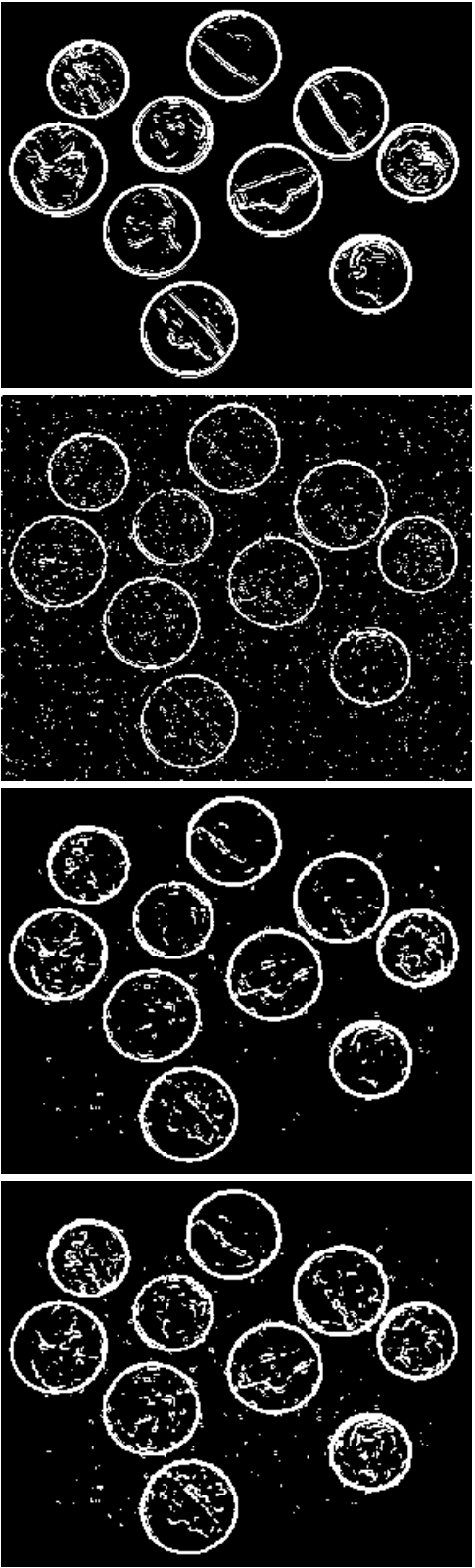
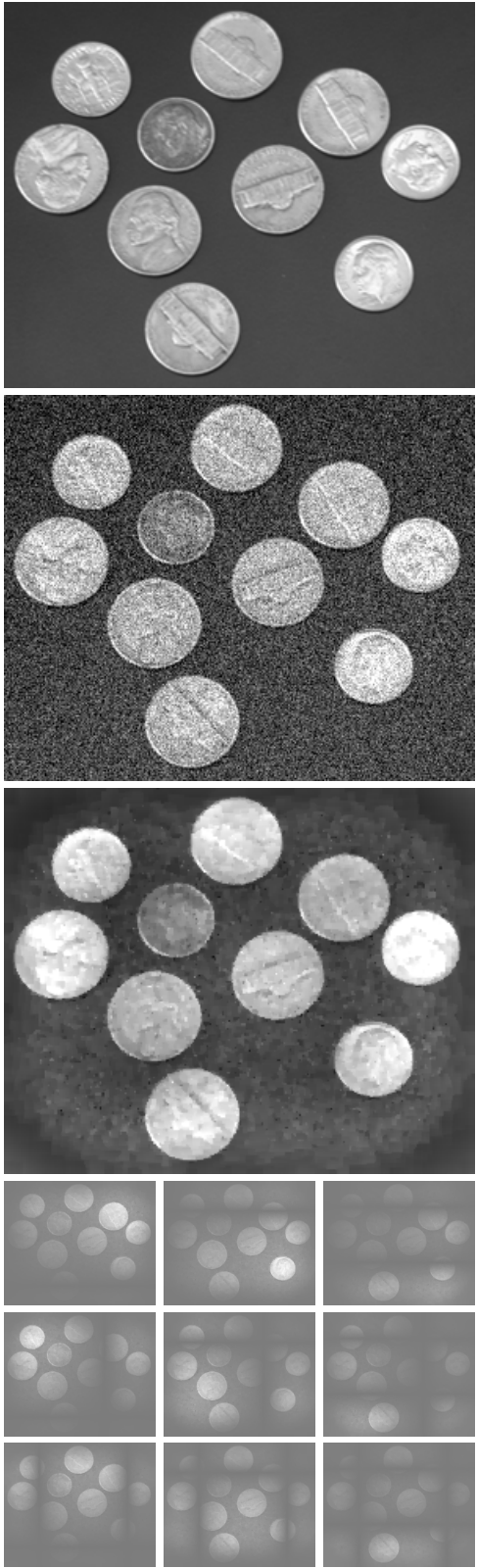


Fig. 4. Results of the experiment for  $\sigma = 30$  as in Fig. 3, but now with orthonormal polynomial basis.



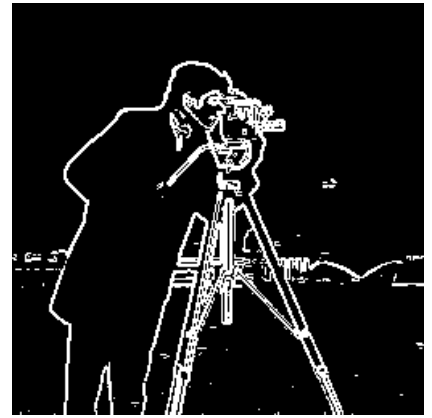


Fig. 5. Results of the experiment for the cameraman image,  $\sigma = 30$ .

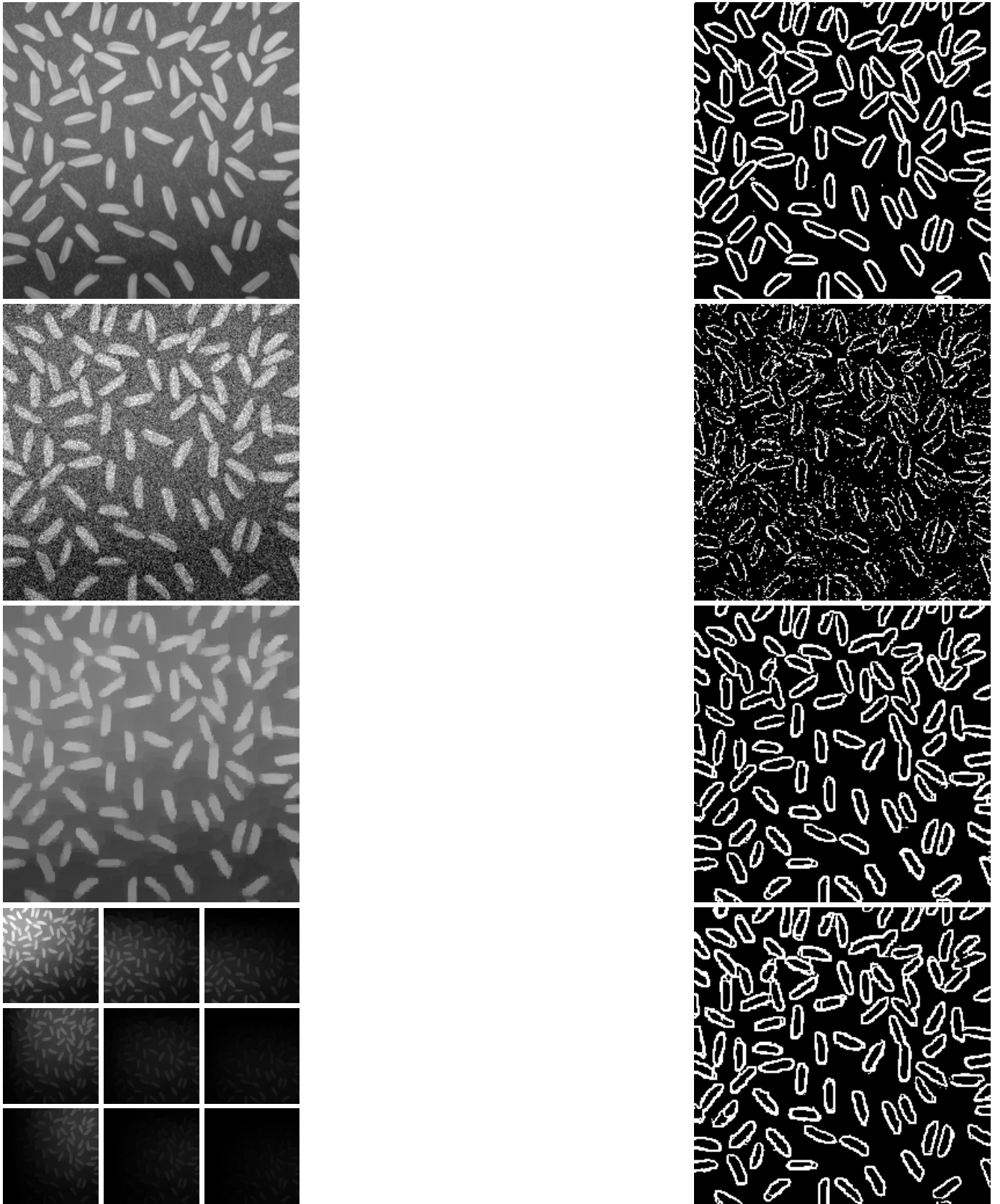


Fig. 6. Results of the experiment for the rice image,  $\sigma = 30$ .