DATA DEPENDENT MULTISCALE TOTAL VARIATION BASED IMAGE DECOMPOSITION AND CONTRAST PRESERVING DENOISING*

TONY F. CHAN† AND FREDERICK E. PARK†

April 2, 2004

Abstract. It is known that the regularization parameter in total variation denoising problems defines a scale space. More recently, however, such a parameter has been shown to have both a causality and critical behavior. We utilize both of these features along with a technique and framework called *parameter marching* to propose new multi-scale data dependent methods for (1) Image Decomposition and (2) Contrast Preserving Image Denoising. The method is based on region merging driven by the minimization of the total variation energy under the assumption that minimizing functions are constrained to be monotonicity preserving. The proposed algorithm in this framework is efficient with a computational complexity on the order of N log(N) with N the number of image pixels.

Key words. Total variation, parameter marching, causality, image processing, denoising, decomposition

AMS subject classifications. 94A08, 94A12, 90C20

1. Introduction. Image decomposition and denoising are two important image processing tasks with a wide range of useful applications. Recently, multiscale decompositions have gained a particular level of interest with the advancements in scale space methods. In the nonlinear setting, a recent trend has been the use of diffusion equations like those seen in the work by Perona and Malik [17] and the various sequels, to generate scale spaces. Here, a given observed image $\mathbf{f} = \mathbf{u}(0, \mathbf{x})$ is used as an initial value to a nonlinear time dependent evolution equation. The subsequent solutions $\mathbf{u}(t, \mathbf{x})$ then, become the resulting scale spaces for each time parameter t.

One of the nonlinear evolution equations that has drawn a lot of attention recently arises from the *Total Variation* (TV) model proposed by Rudin, Osher and Fatemi [19]. This is a very useful model for its ability to preserve edge (discontinuity) information. Precisely, given an observed signal $u^{(0)}(\mathbf{x})$ defined on an image domain Ω, the TV model seeks for the solution to the following minimization problem

$$\min_{u \in BV(\Omega)} \frac{1}{2} \int_\Omega \left[u^{(0)}(\mathbf{x}) - u(\mathbf{x}) \right]^2 d\mathbf{x} + \lambda \int_\Omega |\nabla u(\mathbf{x})| d\mathbf{x} \tag{1.1}$$

where BV is the Banach space of all functions with bounded variation. The corresponding Euler-Lagrange equation is given by,

$$u^{(0)}(\mathbf{x}) - u(\mathbf{x}) + \lambda \nabla \cdot \left(\frac{\nabla u(\mathbf{x})}{|\nabla u(\mathbf{x})|} \right) = 0. \tag{1.2}$$

Associated to the above TV model (1.1) is the notion of scale which is introduced in the work [21] by T.F. Chan and D. Strong where scale is defined as the area to perimeter ratio of image features. The same authors go on to show the important property that the regularization parameter λ in the above model (1.1) and equation

*This work was supported in part by NSF contract ACI-0072112 and NSF contract DMS-9973341.

†Department of Mathematics, University of California, 405 Hilgard Avenue, Los Angeles, CA 90095-1555 ({chan,fpark}@math.ucla.edu).

(1.2) is proportional to the scale of image features. Consequently, a solution $\mathbf{u}(\lambda)$ to the above equation (1.2) generates a scale space for each of the parameters λ.

Of course there are many ways to view scale spaces for a given image and to form the subsequent decompositions from these scale spaces. Fourier, wavelet, linear, and nonlinear filtering decompositions are some popular methods and some references to these types of methods include [20] (add some references). However, the scope of this paper is to address variational and PDE based image decomposition methods. We now shift our focus to some key image decomposition methods which are related to the proposed one.

We need briefly mention the linear wavelet approaches to image decomposition. Here, images may be decomposed according to dyadic scales in a fast and efficient manner. We reference the work by G. Strang [20] for an introduction to wavelet decompositions and for references therein.

A recent TV based image decomposition model is introduced by Osher et al. in [16]. This model arises from a variant of the TV model where a given image $u^{(0)}$ can be decomposed as $u^{(0)} = u + v$ with u and v the cartoon and texture part of the image respectively. When used in conjunction with the TV model, the above decomposition becomes the *Rudin-Osher-Fatemi* (ROF) decomposition introduced in the work [15] by Y. Meyer. In [16], u can either be a minimizer to the standard ROF model or to a variant using the H^{-1} norm. This is an important result that allows the decomposition of images into the separate components of texture and cartoon and allows for the simultaneous processing of images according to these two components.

A multiscale TV decomposition is introduced in the work by E. Tadmor, S. Nezzar, and L. Vese in [22]. In this work, the approach is to decompose an image with respect to dyadic scales, $2^j\lambda$, defined by the regularization parameter λ seen in the TV model (1.1). The initial choice of the parameter λ is based on a heuristic of the initial image to capture the smallest oscillation. This is an interesting approach that allows for a broad range of dyadic scales yielding fine or coarse decompositions.

Some important applications of the aforementioned image decompositions include but are not limited to image compression(cite), image denoising [16], object recognition, and feature extraction. In this paper, we propose the use of a monotonicity preserving variant of the above TV model and the associated scale properties for data dependent image decomposition and contrast preserving denoising. This variant was first introduced by A. Yip and F. Park in [7] and is seen as follows:

$$\min_{u\in\mathcal{M}(\Omega)} \frac{1}{2}\int_\Omega \left[u^{(0)}(\mathbf{x}) - u(\mathbf{x})\right]^2 d\mathbf{x} + \lambda\int_\Omega |\nabla u(\mathbf{x})| d\mathbf{x} \tag{1.3}$$

$$\mathcal{M}(\Omega) = \{u(\lambda) \in BV(\Omega) |\ u(\lambda) \textit{ is monotonicity preserving}\}.$$

Here, solutions $\mathbf{u}(\lambda)$ are said to be monotonicity preserving if they preserve their gray level orderings for all λ. We further remark that solutions to the minimization problem (1.1) are continuous with respect to the regularization parameter λ, see [1] and [7]. A formal definition of monotonicity preserving follows:

DEFINITION 1.1. *If $\mathbf{u}(\lambda) = \{u_{i,j}(\lambda)\}_{1\le i\le m, 1\le j\le n}$ are the solutions of the minimization problem (1.1) in the discrete setting with respect to the regularization parameter λ. We say that the solutions are monotonicity preserving if, for each i, j, p, q such that $|i-p| + |j-q| = 1$, we have*

$$u_{i,j}(\lambda_0) \ge u_{p,q}(\lambda_0) \textit{ for some } \lambda_0 \ge 0 \Longrightarrow u_{i,j}(\lambda) \ge u_{p,q}(\lambda) \textit{ for all } \lambda \ge \lambda_0. \tag{1.4}$$

In particular,

$$u_{i,j}(\lambda_0) = u_{p,q}(\lambda_0) \text{ for some } \lambda_0 \geq 0 \Longrightarrow u_{i,j}(\lambda) = u_{k,l}(\lambda) \text{ for all } \lambda \geq \lambda_0.$$

The above constraint allows for some useful properties for the task of image decomposition. It is shown in the work [9] that the geometry of image features is preserved in the sense that regions are not split apart nor altered in shape. Hence the remaining features in each subsequent scale space will not have their geometry changed. Moreover, there is the notion of critical regularization parameters, λ, first introduced in the work [7]. In this setting, each critical parameter occurs exactly when there is a change in scale. Thus, solutions to the minimization problem (1.3) at these critical parameters become the most relevant scale spaces. Furthermore, since the solutions are only calculated at each critical parameter, the method becomes data dependent in the sense that solutions are only calculated at the scales present in the given image. In addition to the above properties, there is also a causality property where one can directly utilize the information from the solution at the present critical parameter for the computation of the solution at the next critical parameter. This property allows the solutions to be efficiently computed in a hierarchical manner known as *parameter marching* (PM). Thus, one can extract all of the scale information present in a given image from fine to coarse scale. The causality property and PM method are both introduced in the work [7].

We now point out some key differences between the proposed PM method and the above mentioned methods in [22], [16], and [20]. In the multiscale decomposition [22], although the initial choice of $\lambda^{(0)}$ is based on a heuristic of the given image, there is the possibility that the subsequent parameters $2^j\lambda^{(0)}$ may not directly correspond to a scale change of the image. E.g. image scales may not necessarily be dyadic. Moreover, when solving the series of Euler-Lagrange equations in [22], the solution at a previous parameter 2^{j-1} serves only as an initial guess to solving the Euler-Lagrange equation at the next parameter, $2^j\lambda$. Thus, the information from previous solutions is not directly used in the computation of later solutions. The difference between the proposed method and the one in [16] is that in the latter, the emphasis is on the decomposition of an image into the two components of texture and cartoon. The scope of the proposed method is on the multiscale decomposition of images.

It is well known that the TV model (1.1) does not in general preserve contrast. However, in addition to the above mentioned new multiscale image decomposition, we are also proposing a new variant which is contrast preserving. This new variant, called *parameter marching L^2 projection method* (PMP), will be shown to be useful in the applications of contrast preserving image decomposition and contrast preserving image denoising. Thus, geometry and contrast can both be preserved.

In §2 we introduce the proposed monotonicity preserving TV model in the discrete setting and the *parameter marching* method. The applications of the PM method to image decomposition and feature extraction are introduced in §3. The contrast preserving variant of the PM method, the *L^2 projection method*, is introduced in §4 along with the applications to image decomposition and contrast preserving denoising. Numerical Implementation of the two algorithms and some numerical results on both real and synthetic images are presented in §5. Finally, we give the future works and directions in §6.

2. The Discrete Monotonicity Preserving TV Model. In this section we introduce the discrete monotonicity preserving total variation model that will be used

for image decomposition and introduce the 2-d Parameter Marching (PM) method to obtain solutions to this model.

Given the above definition (1.1), we introduce our proposed decomposition model in the two dimensional discrete case as the monotonicity preserving TV model (MPTV):

$$\min_{\mathbf{u}(\lambda)\in\mathcal{M}(\Omega)} \frac{1}{2}\sum_{i=1}^{m}\sum_{j=1}^{n}\left(u_{i,j}-u_{i,j}^{(0)}\right)^2+\lambda\sum_{i=1}^{m-1}\sum_{j=1}^{n-1}\left(|u_{i+1,j}-u_{i,j}|+|u_{i,j+1}-u_{i,j}|\right) \tag{2.1}$$

$$\textit{with } \mathcal{M}(\Omega)=\{u(\lambda)\in BV(\Omega)|\ u(\lambda) \textit{ is monotonicity preserving}\}.$$

Monotonicity simply states that solutions to the TV model at some $\lambda^{(0)}$ do not break regions at some $\lambda > \lambda^{(0)}$. In the 1-d discrete setting, it is shown in [7], that solutions to the TV model exhibit monotonicity preservation. In the 2-d case, this is in general, not the case. However, if we add the additional assumption that solutions are constrained to be monotonicity preserving, we may directly extend the framework set by the authors in [7] to the 2-d case.

Before computing the solutions to the decomposition model (2.1), we first introduce some important properties of the model needed in the computation.

2.1. Scale. In this paper, we will adopt the definition of scale by Chan and Strong [21] for our model:

DEFINITION 2.1. *Let F be an image feature consisting of a region R (maximal connected set) with intensity value β, then*

$$\textit{scale} = \frac{|R|}{|\partial R|}, \tag{2.2}$$

where $|R|$ denotes the area (resp. volume in 3-d) and $|\partial R|$ denotes the perimeter (resp. surface area in 3-d).

Thus, in this setting, scale is simply the area to perimeter ratio of the image features. Hence, small scale features will have small area and/or large perimeter while large scale features will have the opposite. Noise, construed as a small scale feature, has a small area but proportionally large perimeter and is consistent with the definition.

2.2. Critical Regularization Parameters. The modified TV model has the existence of critical regularization parameters λ first introduced in the work [7]. The parameters are critical in the sense that solutions to the proposed model (2.1) have a change in scale at each critical λ. Precisely, given a solution $\mathbf{u}(\lambda^{(k-1)})$ to the minimization problem (2.1) for a parameter $\lambda^{(k-1)} < \lambda^{(k)}$, the solution $\mathbf{u}(\lambda^{(k)})$ at the next critical parameter $\lambda^{(k)}$ differs from $\mathbf{u}(\lambda^{(k-1)})$ by one scale.

A simple way to understand the aforementioned critical behavior is to consider the following example observed in Fig. 5.1. Here, the initial data consists of a snowflake, a cross, and a firecracker that is exhibited in Figure 5.1(a). In context of the above definition of scale, each feature has a different scale corresponding to the area to perimeter ratio. Thus, there are 3 critical parameters, $\lambda^{(k)}, k = 1, 2, 3$, each corresponding to a scale change in the image. In Fig. 5.1(b), the first feature (snowflake) merges with the background corresponding to the first critical parameter $\lambda^{(1)}$. From the above definition of scale, this feature has a small area and large perimeter, thus, the smallest scale and merges with the background before the other features. Subsequently, in Fig. 5.1(e) the cross merges with the background and corresponds to the

next scale change associated to critical parameter $\lambda^{(2)}$. Finally, in Fig. 5.1(g), the firecracker merges with the background to signify the final and largest scale change.

This critical behavior of the regularization parameter associated to the MPTV model allows one to consider only the meaningful scale spaces where a scale change occurs. Thus, our aim is to decompose an image with respect to the present image scales defined by the data.

2.3. Causality. We must briefly mention that the causality property introduced in [7] also carries over to the proposed image decomposition model. By causality, we mean that in the MPTV model (2.1), there is a direct correlation between solutions at subsequent critical parameters. Here, each solution at a given $\lambda^{(k)}$ depends directly on the solution at the previous critical parameter $\lambda^{(k-1)}$ and not on any other solutions at parameters $\tilde{\lambda} < \lambda^{(k-1)}$. This property will be later shown to be a useful feature allowing for an efficient method of computing the solutions hierarchically from small critical regularization parameters to large critical parameters λ called *Parameter Marching* (PM).

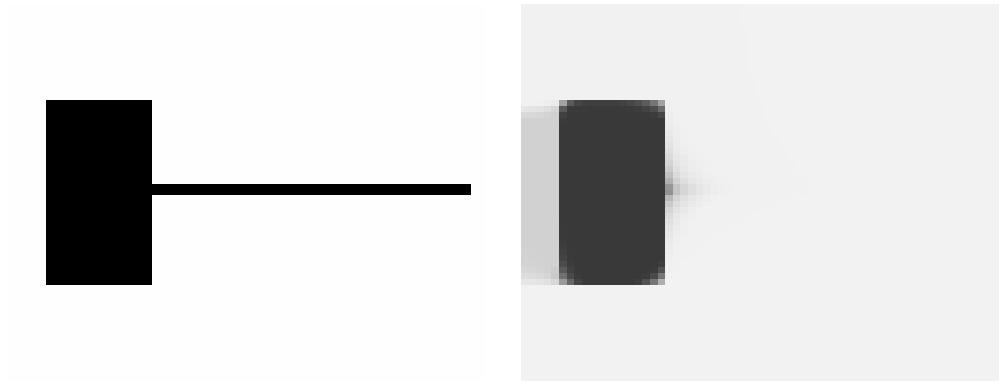
3. Geometry Preserving Total Variation. The additional constraint that minimizing functions of the proposed MPTV model be monotonicity preserving has a very interesting consequence, namely, geometry preservation.

In [19], Rudin et al proposed the total variation (TV) model for image denoising. In their model, given a noisy image $\mathbf{u}^{(0)}$ and a regularization parameter $\lambda \geq 0$, one seeks for an image $\mathbf{u}$ such that the TV functional in (1.1) is minimized over the space of all functions defined on the domain of $\mathbf{u}^{(0)}$ having bounded variation. An important feature of this model is that edges are preserved after the denoising process while traditional techniques such as linear filtering and Tikohnov regularization with L^2 or H^1 penalty would result in smeared edges.

Although the use of the TV regularization has shown to be useful in preserving edges, there are some caveats arising from the minimization of the TV-norm. Figure 3 illustrates that, using the TV denoising model, important image features may disappear or become very vague. Moreover, artifacts may also be introduced. In fact, removal of a constant region induces a decrease in total variation roughly proportional to its perimeter and an increase in the fitting term roughly proportional to its area. This gives an intuition why the long thin feature, which has a high perimeter-to-area ratio, in Figure 3(a) does not show up in Figure 3(b).

To tackle this problem, one might think of modifying the objective function in such a way that the change of the objective function value due to removal of a constant region is proportional to the area of the region rather than its perimeter-to-area ratio. However, the same goal can be accomplished with a simpler approach based on the observation that the aforementioned undesirable phenomenon of the TV model may be attributed to its inability to preserve monotonicity in the 2-D case. In other words, using a monotonicity preserving denoising method, a region having constant grey level would not be broken down into pieces having different grey levels. In this paper, we propose a denoising scheme derived from the TV model which inherits its ability to preserve edges while the monotonicity preserving property is enforced.

Interestingly, the TV model does possess the monotonicity preserving property in the 1-D case. Thus, it is natural to ask why the monotonicity preserving property in the 1-D case is lost in higher dimensional cases and to ask whether such a discrepancy can be repaired. These are the problems that we will address in this paper. In fact, we may relate this phenomenon to numerical methods for solving hyperbolic conservation laws. If we treat λ as the time variable, then any convergent numerical



(a) (b)

FIG. 3.1. *(a) Original image. (b) Solution to the TV model using $\lambda = 300$. We observe that the long thin feature has disappeared and that artifacts in grey appear on the left of the black object. This shows that the TV model cannot preserve features having constant grey levels and thus the quality of solutions is deteriorated.*

method for solving them must be total variation diminishing (TVD) since the TV-norm of the minimizer is a non-increasing function in λ, regardless of the dimension of the problem. On the other hand, it is well-known that in the 1-D case, any TVD scheme is monotonicity preserving [13, p.166] which is generally not true in higher dimensional cases. This is exactly what happens to the TV model.

We remark that in [7], Yip and Park proposed a parameter marching scheme which exploits the monotonicity preserving property and causality of λ for solving TV denoising problems in the 1-D case. Our paper generalizes the techniques in [7] to the 2-D case for the use in geometry preserving denoising and decomposition problems.

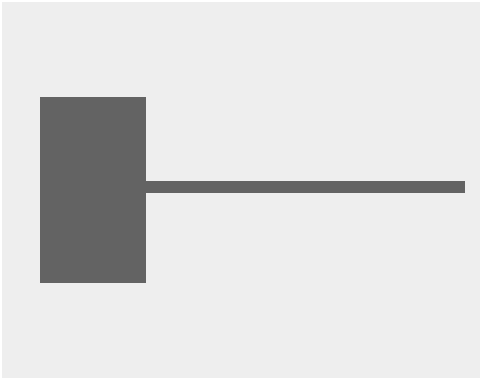
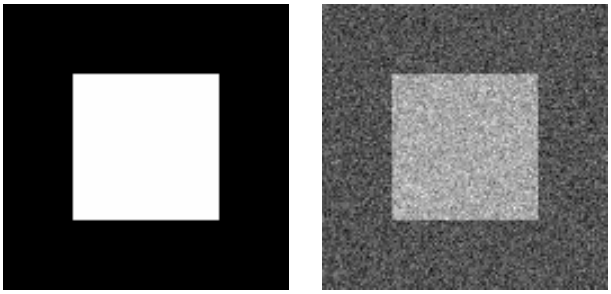


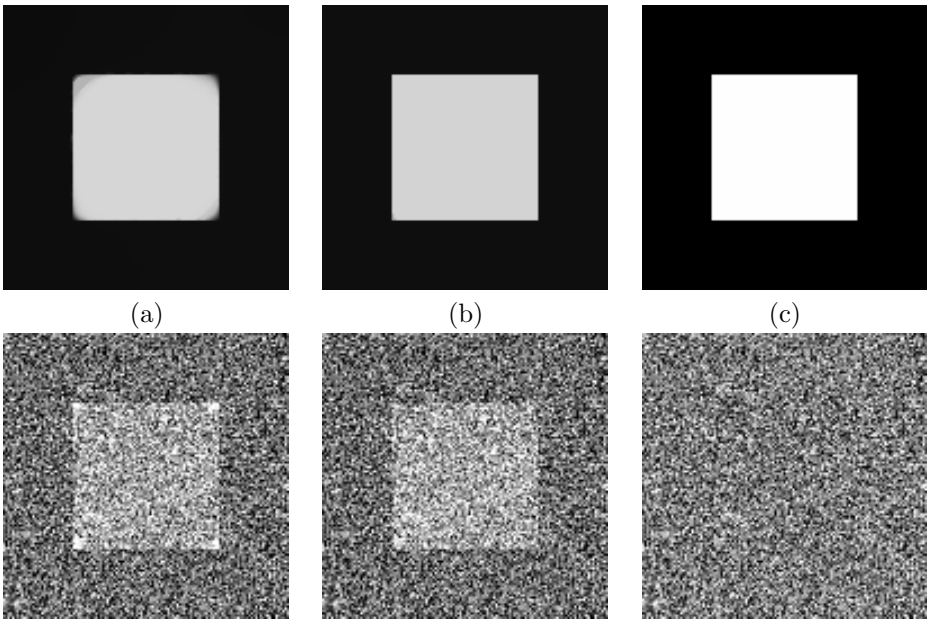
FIG. 3.2. *Denoised image using our proposed model with $\lambda = 300$.*

4. The 2-d Parameter Marching Method. In this section, we introduce the the parameter marching method for solving the minimization problem (2.1). Moreover, we also give an explicit $NLog(N)$ algorithm.



(a) (b)

FIG. 3.3. Observed Images: *(a) Clean image. (b) Noisy image with additive Gaussian noise, SNR=5.*



(a) (b) (c)

(d) (e) (f)

FIG. 3.4. Denoised Images: *(a) TV Model. (b) PM method. (c) PMP method. (d)–(f) noise components $v = f - u + 120$, for the TV model, PM, and PMP methods respectively. Here, u is the recovered image obtained from each method. We observe in (a), for the TV model, a geometry (e.g. corners are missing) and contrast loss. In (b), we observe the MPTV model can preserve the geometry of the image (e.g. corners are kept), however, a contrast loss is also observed. In (c), the PMP method preserves both contrast and geometry of the image and the restored image appears exactly as the original clean image. In the noise components (d)–(e), some contrast from the restored image is observed in the noise. In (f), there are no discernable image features, only noise.*

4.1. Parameter Marching Framework. Before proceeding to the PM method, we first fix some notation. For each (i, j), let $R_{i,j}^{(k)}$ be the maximal connected neighborhood of (i, j) at which $\mathbf{u}^{(k)}$ takes the same value. We note that this domain may have arbitrary shape. Let us define $a_{i,j}^{(k)}$ to be the average value of $\mathbf{u}^{(0)}$ on the region

$R_{i,j}^{(k)}$:

$$a_{i,j}^{(k)} := \frac{\sum_{(p,q)\in R_{i,j}^{(k)}} u_{p,q}^{(0)}}{|R_{i,j}^{(k)}|}. \tag{4.1}$$

We also define the indicator variables $\delta_{p+\frac{1}{2},q}(\mathbf{u}^{(k)})$ and $\delta_{p,q+\frac{1}{2}}(\mathbf{u}^{(k)})$ as:

$$\delta_{p+\frac{1}{2},q}(\mathbf{u}^{(k)}) = \begin{cases} 1 & \text{if} \quad u_{p+1,q} \geq u_{p,q} \\ -1 & \text{if} \quad u_{p+1,q} < u_{p,q} \end{cases} \quad \begin{array}{ll} \text{for} & p = 1,\ldots,n-1, \\ & q = 1,\ldots,m-1 \end{array} \tag{4.2}$$

and

$$\delta_{p,q+\frac{1}{2}}(\mathbf{u}^{(k)}) = \begin{cases} 1 & \text{if} \quad u_{p,q+1} \geq u_{p,q} \\ -1 & \text{if} \quad u_{p,q+1} < u_{p,q} \end{cases} \quad \begin{array}{ll} \text{for} & p = 1,\ldots,n-1, \\ & q = 1,\ldots,m-1. \end{array} \tag{4.3}$$

Furthermore, we define $\alpha_{i,j}^{(k)}$ as

$$\alpha_{i,j}^{(k)} := \tag{4.4}$$

$$\frac{1}{|R_{i,j}^{(k)}|} \sum_{(p,q)\in R_{i,j}^{(k)}} \left[\delta_{p+\frac{1}{2},q}(\mathbf{u}^{(k)}) - \delta_{p-\frac{1}{2},q}(\mathbf{u}^{(k)}) + \delta_{p,q+\frac{1}{2}}(\mathbf{u}^{(k)}) - \delta_{p,q-\frac{1}{2}}(\mathbf{u}^{(k)})\right]$$

$$= \frac{1}{|R_{i,j}^{(k)}|} \left[\sum_{(p,q)\in \partial_{\mathrm{T}} R_{i,j}^{(k)}} \delta_{p,q+\frac{1}{2}}(\mathbf{u}^{(k)}) + \sum_{(p,q)\in \partial_{\mathrm{R}} R_{i,j}^{(k)}} \delta_{p+\frac{1}{2},q}(\mathbf{u}^{(k)}) \right.$$

$$\left. - \sum_{(p,q)\in \partial_{\mathrm{B}} R_{i,j}^{(k)}} \delta_{p,q-\frac{1}{2}}(\mathbf{u}^{(k)}) - \sum_{(p,q)\in \partial_{\mathrm{L}} R_{i,j}^{(k)}} \delta_{p-\frac{1}{2},q}(\mathbf{u}^{(k)}) \right].$$

Here, $\partial_z R_{i,j}^{(k)}$ for $z \in \{\mathrm{T},\mathrm{R},\mathrm{B},\mathrm{L}\}$ denote the top, right, bottom, and left boundaries of $R_{i,j}^{(k)}$ respectively.

The next Lemma states that the minimization of the MPTV model (2.1) is equivalent to solving a quadratic programming problem when prior information on the neighboring points $\delta_{i+\frac{1}{2},j}(\lambda) = \mathrm{sgn}[u_{i+1,j}(\lambda) - u_{i,j}(\lambda)]$ and $\delta_{i,j+\frac{1}{2}}(\lambda) = \mathrm{sgn}[u_{i,j+1}(\lambda) - u_{i,j}(\lambda)]$ are known for all i, j.

LEMMA 4.1. *Suppose the sign of the difference between the values of each pair of neighboring points of the solution* $\mathbf{u}(\lambda)$ *to (2.1) is know, then the minimization problem (2.1) is equivalent to*

$$\begin{aligned} \min_{\mathbf{u}\in\mathbb{R}^{m\times n}} \quad & \frac{1}{2}\sum_{i=1}^{m}\sum_{j=1}^{n}\left(u_{i,j} - u_{i,j}^{(0)}\right)^2 + \lambda \sum_{i=1}^{m-1}\sum_{j=1}^{n-1} \delta_{i+\frac{1}{2},j}(\mathbf{u}(\lambda))(u_{i+1,j} - u_{i,j}) \\ & + \lambda \sum_{i=1}^{m-1}\sum_{j=1}^{n-1} \delta_{i,j+\frac{1}{2}}(\mathbf{u}(\lambda))(u_{i,j+1} - u_{i,j}) \\ \textit{subject to} \quad & \delta_{i+\frac{1}{2},j}(\mathbf{u}(\lambda))(u_{i+1,j} - u_{i,j}) \geq 0 \quad \textit{if} \quad \mathrm{sgn}[u_{i+1,j}(\lambda) - u_{i,j}(\lambda)] \neq 0 \\ & u_{i+1,j} - u_{i,j} = 0 \quad \textit{if} \quad \mathrm{sgn}[u_{i+1,j}(\lambda) - u_{i,j}(\lambda)] = 0 \\ & \delta_{i,j+\frac{1}{2}}(\mathbf{u}(\lambda))(u_{i,j+1} - u_{i,j}) \geq 0 \quad \textit{if} \quad \mathrm{sgn}[u_{i,j+1}(\lambda) - u_{i,j}(\lambda)] \neq 0 \\ & u_{i,j+1} - u_{i,j} = 0 \quad \textit{if} \quad \mathrm{sgn}[u_{i,j+1}(\lambda) - u_{i,j}(\lambda)] = 0. \end{aligned} \tag{4.5}$$

Proof. The four linear constraints simply require the solution to satisfy the constraints $\text{sgn}[u_{i+1,j}(\lambda) - u_{i,j}(\lambda)] = \delta_{i,j+\frac{1}{2}}(\lambda)$ and $\text{sgn}[u_{i,j+1}(\lambda) - u_{i,j}(\lambda)] = \delta_{i,j+\frac{1}{2}}(\lambda)$ □

The next result yields an explicit solution to the above quadratic programming problem (4.5).

LEMMA 4.2. *For each $\lambda \geq 0$, the unique solution to the quadratic programming problem (4.5), given the values of $\delta_{i+\frac{1}{2},j}(\lambda)$ and $\delta_{i,j+\frac{1}{2}}(\lambda)$, has the explicit form:*

$$u_{i,j}(\lambda) = a_{i,j}^{(k)} + \lambda \alpha_{i,j}^{(k)} = u_{i,j}^{(k)} + (\lambda - \hat{\lambda}^{(k)})\alpha_{i,j}^{(k)} \quad \forall\ i,j. \tag{4.6}$$

Proof. The result is a straightforward computation similar to the one exhibited in [7] and follows in exactly the same manner. □

The monotonicity constraint defined in (1.4) ensures that the $\delta_{i,j}$'s are invariant for small increases of λ and we state this result in the following Lemma.

LEMMA 4.3. *Given a solution $\mathbf{u}(\lambda_0)$, $\lambda_0 < \infty$, $\exists$ an $\epsilon > 0$ such that the sign of the differences between the values of neighboring points of the minimizer $\mathbf{u}(\lambda)$ of (2.1) is the same as that of $\mathbf{u}(\lambda_0)$ for all $\lambda \in [\lambda_0, \lambda_0 + \epsilon)$, precisely,*

$$\begin{aligned} \delta_{i+\frac{1}{2},j}(\lambda) = \delta_{i+\frac{1}{2},j}(\lambda_0) \text{ and } \delta_{i,j+\frac{1}{2}}(\lambda) = \delta_{i,j+\frac{1}{2}}(\lambda_0) \\ \forall\ \lambda \in [\lambda_0, \lambda_0 + \epsilon) \text{ and } \forall\ i,j. \end{aligned} \tag{4.7}$$

Proof. Case 1: $\delta_{i+\frac{1}{2},j} \neq 0$ and $\delta_{i,j+\frac{1}{2}} \neq 0$. This follows directly from the fact that solutions $\mathbf{u}(\lambda)$ of the TV model are continuous in the regularization parameter, cf. [7, 23].
Case 2: $\delta_{i+\frac{1}{2},j} = 0$ or $\delta_{i,j+\frac{1}{2}} = 0$. This follows directly from the definition of monotonicity preservation. □

The above Lemma (4.3) simply states that if the signs of the values of the difference of neighboring points, e.g. $\delta_{i+\frac{1}{2},j}(\lambda_0)$ and $\delta_{i,j+\frac{1}{2}}(\lambda_0)$ are known, then $\delta_{i+\frac{1}{2},j}(\lambda_0 + \epsilon)$ and $\delta_{i,j+\frac{1}{2}}(\lambda_0 + \epsilon)$ are also known for ϵ small enough. Thus, the shape of the solution does not change in the small interval $[\lambda_0, \lambda_0 + \epsilon)$. We now make the following remark about obtaining solutions for $\lambda \in [\lambda_0, \lambda_0 + \epsilon)$ given the solution at λ_0.

THEOREM 4.4. *If $\mathbf{u}(\lambda)$ is the solution to the discrete MPTV model at λ_0 with explicit solution formula given by:*

$$u_{i,j}(\lambda_0) = a_{i,j}(\lambda_0) + \lambda_0 \alpha_{i,j}(\lambda_0)$$

and let $\epsilon > 0$ be such that Lemma 4.3 holds, then the solution for any $\lambda \in [\lambda_0, \lambda_0 + \epsilon)$ has the following exact formulation:

$$u_{i,j}(\lambda) = a_{i,j}(\lambda_0) + \lambda \alpha_{i,j}(\lambda_0) = u_{i,j}(\lambda_0) + (\lambda - \lambda_0)\alpha_{i,j}(\lambda_0). \tag{4.8}$$

We now wish to find the update for each critical regularization parameter. In short, given the solution $\mathbf{u}(\lambda^{(k)})$ at a critical parameter $\lambda^{(k)}$, we wish to find the maximal value ϵ_M of ϵ such that the solution $u(\lambda^{(k)})$ at $\lambda^{(k)} + \epsilon_M$ has a scale change

when compared to $u(\lambda^{(k)})$. In 1-d it is shown in [7] that such a parameter is found by computing:

$$\min J(\mathbf{u}^{(k)}) = \min\left\{ \frac{u_{i+1}^{(k)} - u_i^{(k)}}{\alpha_i^{(k)} - \alpha_{i+1}^{(k)}} + \hat{\lambda}^{(k)} \mid \alpha_i^{(k)} \neq \alpha_{i+1}^{(k)},\ u_{i+1}^{(k)} \neq u_i^{(k)} \right\}. \tag{4.9}$$

In 2-d, if we define $I(\mathbf{u}^{(k)})$ as

$$I(\mathbf{u}^{(k)}) := I_{\mathrm{h}}(\mathbf{u}^{(k)}) \cup I_{\mathrm{v}}(\mathbf{u}^{(k)}) \tag{4.10}$$

with

$$I_{\mathrm{h}}(\mathbf{u}^{(k)}) := \left\{ \frac{u_{i+1,j}^{(k)} - u_{i,j}^{(k)}}{\alpha_{i,j}^{(k)} - \alpha_{i+1,j}^{(k)}} + \hat{\lambda}^{(k)} \mid \alpha_{i,j}^{(k)} \neq \alpha_{i+1,j}^{(k)},\ u_{i+1,j}^{(k)} \neq u_{i,j}^{(k)} \right\} \tag{4.11}$$

and

$$I_{\mathrm{v}}(\mathbf{u}^{(k)}) := \left\{ \frac{u_{i,j+1}^{(k)} - u_{i,j}^{(k)}}{\alpha_{i,j}^{(k)} - \alpha_{i,j+1}^{(k)}} + \hat{\lambda}^{(k)} \mid \alpha_{i,j}^{(k)} \neq \alpha_{i,j+1}^{(k)},\ u_{i,j+1}^{(k)} \neq u_{i,j}^{(k)} \right\}. \tag{4.12}$$

Then, the next critical parameter can be found by computing: $\lambda^{(k+1)} = \min I(\mathbf{u}^{(k)})$. Moreover, as in the 1-d case, $I(\mathbf{u}^{(k)}) \neq \phi$ if and only if $\mathbf{u}(\lambda^{(k)})$ is nonconstant and if $I(\mathbf{u}^{(k)}) \neq \phi$, we set $\min I(\mathbf{u}^{(k)}) = +\infty$.

Now, suppose $\lambda^{(k+1)}$ obtained above from (4.9) is finite, then the solution to the MPTV model for $\lambda = \lambda^{(k+1)}$ can be given by:

$$u_{i,j}(\lambda^{(k+1)}) = a_{i,j}(\lambda^{(k)}) + \lambda^{(k+1)} \alpha_{i,j}(\lambda^{(k)}). \tag{4.13}$$

This result is the incremental update for a solution $\mathbf{u}(\lambda^{(k+1)})$ at $\lambda^{(k+1)}$ given the solution at $\lambda^{(k)}$, $\mathbf{u}(\lambda^{(k)})$ and follows directly by taking limits in the equation (4.8).

Thus, by utilizing the above incremental update given in equation (4.13), for a given $\lambda \geq 0$, we can construct a sequence of solutions $\{\mathbf{u}^{(k)}\}_{k=0}^{K}$ with $\hat{\lambda}^{(K)} = \lambda$ for some $0 \leq K \leq n-1$. This technique (i.e. computing the final solution by iteratively increasing λ) was first introduced in [7] and is called *parameter marching*.

4.2. Properties of the PM Method. One useful property of this method that we wish to exploit is causality. This idea is first introduced in the work [7] where the parameter λ resembles the role of the time variable in memoryless time-dependent problems (i.e. causality) in the sense that given the solution $\mathbf{u}(\lambda_0)$ at some λ_0, the solution at $\lambda > \lambda_0$ depends only on the solution at λ_0 but not on solutions at parameters less than λ_0. Thus, to obtain the solution at $\lambda > \lambda_0$, we may solve (2.1) with the observed signal $\mathbf{u}^{(0)}$ replaced by $\mathbf{u}(\lambda_0)$ and λ replaced by $\lambda - \lambda_0$. This property will be shown later to be a useful computational tool. An exact statement is formulated in the following proposition:

PROPOSITION 4.5. *(Causality) Given the solution $\mathbf{u}^{(k)}$ of (2.1) at $\hat{\lambda}^{(k)}$ and given $\hat{\lambda}^{(k+1)}$ calculated by (4.9). The solution $\mathbf{u}(\lambda)$ of (2.1) for any $\lambda \in [\hat{\lambda}^{(k)}, \hat{\lambda}^{(k+1)}]$ can be obtained by solving*

$$\min_{\mathbf{u}(\lambda) \in \mathcal{M}} \frac{1}{2} \sum_{i=1}^{m} \sum_{j=1}^{n} \left(u_{i,j} - u_{i,j}^{(k)} \right)^2 + \mu \sum_{i=1}^{m-1} \sum_{j=1}^{n-1} \left(|u_{i+1,j} - u_{i,j}| + |u_{i,j+1} - u_{i,j}| \right) \tag{4.14}$$

where $\mu := \lambda - \hat{\lambda}^{(k)}$.

We will show that each regularization parameter is related to the earlier defined scale. Thus, the solutions at each critical parameter λ become data dependent scale spaces. In a sense, the solutions at each critical parameter are the only meaningful scale spaces since each solution differs by one scale.

We would also like to take the time to remark that in [21], the authors not only make the definition of scale in the context of the TV model but further show that for radially symmetric functions the regularization parameter λ is directly proportional to this scale. If we denote by δ the change in image intensity, then we state this relation in the discrete case as the following remark:

Remark *Let* $\mathbf{u}^{(0)}$ *be an image consisting of two regions, (a) an arbitrarily shaped extremal region* R *and (b) background region* $\mathbf{u}^{(0)} \setminus R$. *If* δ *denotes the change in image intensity of this extremal region, then the regularization parameter*

$$\lambda \ \sim \ \delta \cdot scale = \delta \cdot \frac{|R|}{|\partial R|}.$$

Here $|R|$ *and* $|\partial R|$ *denote the area and perimeter of the region* R *respectively. Moreover, if we neglect the background change in the solution* $u(\lambda)$ *at the critical parameter* λ *to the MPTV problem, then*

$$\lambda = \delta \cdot scale = \delta \cdot \frac{|R|}{|\partial R|}.$$

The remark directly follows from λ update formula (4.9). From the above remark, we may deduce that each $\lambda^{(k)}$ corresponds directly to an image feature $(R^{(k)}, \beta^{(k)})$, ($R^{(k)}$ denoting a region with intensity value $\beta^{(k)}$) through the relation that $\lambda^{(k)}$ is roughly equal to the difference in image intensities (jump) times scale of the corresponding region $R^{(k)}$. Hence, as the parameter is incrementally increased, the corresponding merging of the regions proceeds from fine (small) λ to coarse scale (larger λ).

5. Applications to Image Decomposition. In this section, we adapt the above model (2.1) and the associated PM framework to be used for image decomposition.

5.1. Decomposition Methodology. Precisely, given an initial image $\mathbf{u}^0$, the parameter marching method generates the critical sequence of regularization parameters $0 = \lambda^{(0)} < \lambda^{(1)} < \ldots < \lambda^{(N)}$ and corresponding solutions (scale spaces) $\{\mathbf{u}^{(k)}\}$ for $k = 1, \ldots, N$. For each scale space $\mathbf{u}^{(k)}$ we can define the difference $\mathbf{v}^{(k)} = \mathbf{u}^{(k-1)} - \mathbf{u}^{(k)}$. Therefore, for each k, $\mathbf{u}^{(k-1)} = \mathbf{u}^{(k)} + \mathbf{v}^{(k)}$ is known as the Rudin-Osher-Fatemi decomposition when in conjunction with the TV model. The final PM decomposition becomes:

$$\mathbf{u}^{(0)} = \mathbf{u}^{(N)} + \sum_{i=1}^{N} \mathbf{v}^{(i)}. \tag{5.1}$$

$$\mathbf{u}^{(N)} = \mathbf{u}_{mean}.$$

Here, $\mathbf{u}_{mean}$ is the image having each intensity value equal to the mean value of the original image $\mathbf{u}^{(0)}$. Moreover, if we let $\tilde{\lambda}^{(i)} = \lambda^{(i)} - \lambda^{(i+1)}$ then the above equation

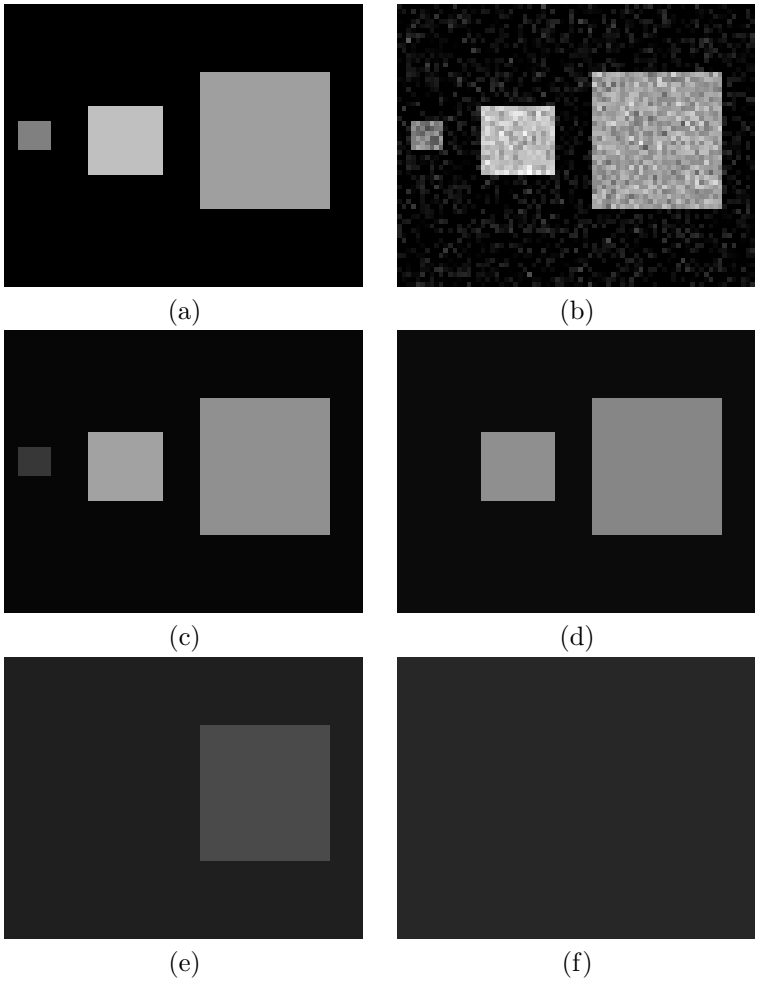


FIG. 4.1. *In (a) we have the original image consisting of 3 squares having different sizes and intensities. (b) the noisy image, SNR = 10. In (c) we have the denoised image which is the first scale space consisting of only image features. (d)–(f) the resulting scale spaces.*

becomes:

$$\mathbf{u}^{(0)} = \mathbf{u}^{(N)} + \sum_{i=0}^{N-1} \tilde{\lambda}^{(i)} \alpha^{(\mathbf{i})}. \tag{5.2}$$

Here, the $\alpha^{(\mathbf{i})}$'s become a global data defined basis in the sense that they contain information about the speed and direction of the remaining regions to be merged at the i-th level of the decomposition. The $\tilde{\lambda}^{(i)}$'s define the order of the merging. Precisely, each $\tilde{\lambda}^{(i)}$ corresponds to the merging of the i-th region in the original image $\mathbf{u}^{(0)}$ and each $\alpha^{(i)}$ contains the speed (scale) and direction of the remaining regions at the i-th step of the decomposition. Moreover, each regularization parameter depends directly on a given image feature in the sense of scale and image intensity.

5.2. Examples of PM Decomposition. In Figure 5.1 we have a decomposition of an image that consists of a cross, a snowflake, and a firecracker all with image

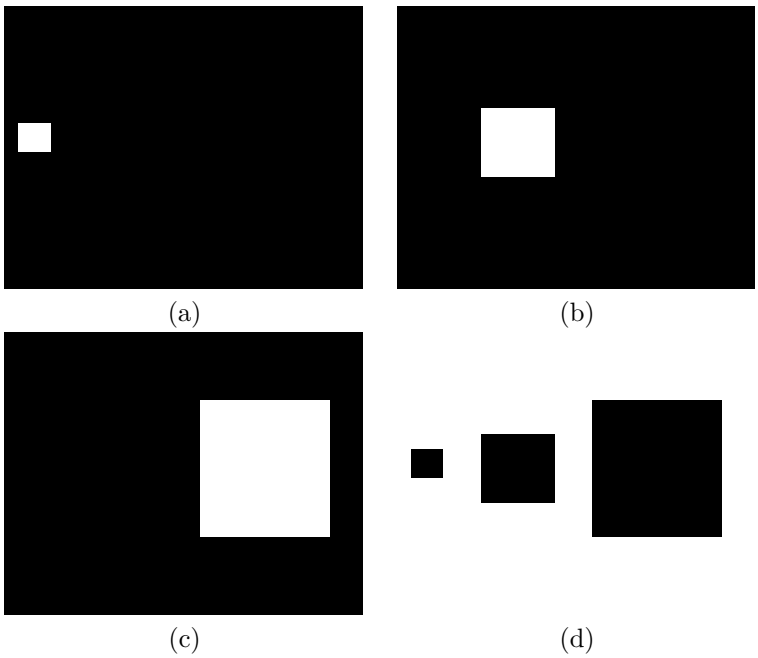


FIG. 4.2. *(a)–(d) The induced basis from the features in the denoised scale space seen in Fig. 4.1 (c).*

intensity values 0 and background value 190 . According to our methodology, the snowflake should be the first object to merge as it has a low area to perimeter ratio. The critical λ which in this case is $\lambda = 194$ is calculated from our algorithm and the snowflake is merged with the background without altering the geometry of the remaining objects in the image. The next critical parameter $\lambda = 312$ is calculated from our method to merge the cross without any incident. Finally the firecracker is merged last by $\lambda = 560$ and the remaining region is expectedly the mean of the initial data. We notice how the regions merge with the background at each step cleanly and without any visible artifacts. Moreover, we can calculate the exact λ value for the merging of each region. We contrast this to Figure 5.3 where we solve the corresponding Euler-Lagrange equation for the TV model. Here, we see that the image features become distorted, e.g. the snowflake in (b) is smeared, the cross has artifacts and the firecracker has lost some of its wick. Moreover, the λ parameters corresponding to the merging of each region are difficult to define as the regions tend to diffuse into the background as opposed to a clean merge.

5.3. TV basis and Induced Orthogonal Basis at Each Level. In this section, we further investigate the basis, $\{\alpha_{\mathbf{i}}\}$ as seen in equation (5.2) which is induced from the monotonicity preserving TV model. We will also introduce an orthogonal basis derived from the above $\alpha_{\mathbf{i}}$'s and extend this notion to the solutions at each k-th level, $\mathbf{u}^{(k)}$. Proceeding with the same decomposition as before, given an image $\mathbf{u}^{(0)}$ we can apply the PM method to find an N, a sequence of regularization parameters $\{\lambda^{(k)}\}$, and corresponding solutions $\{\mathbf{u}^{(k)}\}$ for $k = 1, \ldots, N$ such that $\mathbf{u}^{(0)}$ has the

decomposition

$$\mathbf{u}^{(0)} = \mathbf{u}^{(N)} + \sum_{i=0}^{N-1} \tilde{\lambda}^{(i)} \alpha^{(\mathbf{i})}$$

Here, for the sake of argument, we may assume that $\mathbf{u}^{(0)}$ has N distinct regions. As mentioned earlier, the $\lambda^{(i)}$'s contain information about the merging of the i-th region and $\alpha^{(i)}$'s become a global basis in the sense that they contain information about the merging speeds(e.g. how much each region will change as λ is changed) and directions(increasing or decreasing) of all the remaining regions $R_i^{(k)}$ of $\mathbf{u}^{(k)}$ at the k-th level for $i = 1, \ldots, N-k$. Furthermore, at this k-th level of the decomposition we may define

$$\mathbf{u}^{(k)} = \mathbf{u}^{(0)} - \sum_{i=0}^{k-1} \tilde{\lambda}^{(i)} \alpha^{(\mathbf{i})} = \sum_{i=k}^{N-1} \tilde{\lambda}^{(i)} \alpha^{(\mathbf{i})} + \mathbf{u}^{(N)}. \tag{5.3}$$

The decomposition (5.3), has $N-k$ remaining regions and represents the decomposition for features with scales greater than the k-th scale. Here, for the solution $\mathbf{u}^{(k)}$ we define $\nu_{\mathbf{i}}^{(\mathbf{k})}$ to be the orthogonal basis vector for the i-th region $R_i^{(k)}$ at the k-th level. The $\nu_{\mathbf{i}}^{(\mathbf{k})}$ are given by

$$\nu_{\mathbf{i}}^{(\mathbf{k})} = \delta_{R_i^{(k)}} = \begin{cases} 1 & \text{if} \quad (p,q) \in R_i^{(k)} \\ 0 & \text{if} \quad (p,q) \in \Omega \setminus R_i^{(k)} \end{cases} \qquad \text{for } i = 1, \ldots, N-k. \tag{5.4}$$

We also define $\beta_i^{(k)}$ to be the image intensity for the i-th region $R_i^{(k)}$ at the k-th level so that each feature has the representation $F_i^{(k)} = \beta_i^{(k)} \delta_{R_i^{(k)}}$. Moreover, given the merging information at this level, we may make a change of basis $\{\alpha^{(i)}\} \to \{\nu_i\}$ such that $\mathbf{u}^{(k)}$ becomes

$$\mathbf{u}^{(k)} = \sum_{i=1}^{N-k} \beta_i^{(k)} \nu_{\mathbf{i}}^{(\mathbf{k})}. \tag{5.5}$$

Furthermore, we can do this at each level from fine to coarse scale since we know the remaining regions at the k-th level of the decomposition. Clearly, the finest level is represented by

$$\mathbf{u}^{(0)} = \sum_{i=1}^{N} \beta_i^{(0)} \nu_{\mathbf{i}}^{(\mathbf{0})}. \tag{5.6}$$

We call the $\nu_{\mathbf{i}}^{(\mathbf{k})}$'s the *Induced Orthogonal Basis* at the k-th level.

The feature extraction can best be seen in the following simple example. We consider an image $\mathbf{u}^{(0)}$ consisting of 3 rectangles and background as seen in Figure 5.4. The rectangles vary from small to large size each with image intensity 129, 193, and 160 respectively on a background of intensity 1. The PM method yields 3 parameters $\lambda^{(1)} = 190$, $\lambda^{(2)} = 600$, and $\lambda^{(3)} = 842$ and the corresponding regions to be merged at each step. This is seen in Figure 5.4. $\lambda^{(1)} = 190$ corresponds to the merging of the smallest rectangle, $\lambda^{(2)} = 600$ the merging of the medium rectangle and $\lambda^{(3)} = 842$ the merging of the last rectangle. From the merging information we

may denote each rectangle by $R_i^{(0)}$ for $i = 1, 2, 3$ where the ordering depends on the merging schedule. Here the merging proceeds from smallest to largest rectangles. Let the background be given by $R_4^{(0)}$. We may then form the decomposition

$$\mathbf{u}^{(0)} = \beta_1^{(0)}\delta_{R_1^{(0)}} + \beta_2^{(0)}\delta_{R_2^{(0)}}\beta_3^{(0)}\delta_{R_3^{(0)}} + \beta_4^{(0)}\delta_{R_4^{(0)}} \tag{5.7}$$

with $\beta_i^{(0)}$, $i = 1, 2, 3$ having intensity value 129, 193, and 160 respectively and $\beta_4^{(0)}$ with value 1. The basis vectors $\nu_i^{(0)} = \delta_{R_i^{(0)}}$ for $i = 1, 2, 3$ are exhibited in Figure 5.6. Thus, to extract the largest scale feature, we need only look at the 3rd term in the decomposition, $\beta_3^{(0)}\delta_{R_3^{(0)}}$ and for the finest scale feature, we look to the first term, $\beta_1^{(0)}\delta_{R_1^{(0)}}$. The extracted features can be seen in Figure 5.7.

6. Parameter Marching – L^2 Projection Method (PMP). In TV denoising and decomposition problems it is well known that the contrast of the image is not preserved. It is also pointed out in [15] and [9] that the geometry is not preserved as well. By restricting the class of minimizing functions to be monotonicity preserving it is shown in [9] that one can solve the geometry loss problem. However, the loss of contrast is still present in the model (4.14) and is thus inherited by the PM method. We wish to overcome this problem. Thus, we introduce a contrast preserving variant of the PM method along with its applications to contrast preserving image decomposition and contrast preserving denoising.

6.1. Parameter Marching L^2 Projection Method for Contrast Preservation. Given the decomposition of an image at the k-th level and the corresponding induced basis representation at this level

$$\mathbf{u}^{(k)} = \sum_{i=1}^{N-k} \beta_i^{(k)} \nu_{\mathbf{i}}^{(\mathbf{k})}, \tag{6.1}$$

we note that the contrast defined by the $\beta_i^{(k)}$ is obtained from the evolution of the initial data $\mathbf{u}^{(0)}$ to the solution at the k-th level $\mathbf{u}^{(k)}$ while marching the regularization parameter λ from $\lambda^{(1)}$ to $\lambda^{(k)}$. Hence there is a corresponding contrast loss of the non-merged regions in this scale space, $\mathbf{u}^{(k)}$, as opposed to the original image. This leads us to consider what the best values for the coefficients $\beta_i^{(k)}$ should be. In this setting, a natural choice would be the best fit of the original data $\mathbf{u}^{(0)}$ in the corresponding scale space defined by the basis $\{\nu_{\mathbf{i}}^{(\mathbf{k})}\}$ for $i = 1, \ldots, N-k$. Hence, unmerged regions would exactly retain their original contrast and merged regions would take the average contrast of $\mathbf{u}^{(0)}$ over these regions. Therefore we reduce the contrast loss problem at the k-th level to the minimization problem

$$\min_{\eta_j^{(k)} \in \mathbb{R}} \frac{1}{2} \|\mathbf{u}^{(0)} - \sum_{i=1}^{N-k} \eta_i^{(k)} \nu_{\mathbf{i}}{}^{(k)}\|_2^2. \tag{6.2}$$

The solutions to (6.2) are given in the following theorem

THEOREM 6.1. *For each j, j=1,. . . ,N-k, the solution to (6.2) is given by*

$$\eta_j^{(k)} = \frac{1}{|\nu_{\mathbf{j}}^{(\mathbf{k})}|} \int_{supp(\nu_{\mathbf{j}}^{(\mathbf{k})})} \mathbf{u}^{(0)} d\mathbf{x} \tag{6.3}$$

Here, supp$(\nu_{\mathbf{j}}^{(\mathbf{k})})$ *= support of* $\nu_{\mathbf{j}}^{(\mathbf{k})} = \{(p,q)|\ \nu_{\mathbf{j}}(p,q) = 1\}$ *and* $|\nu_{\mathbf{j}}^{(\mathbf{k})}| = |supp(\nu_{\mathbf{j}}^{(\mathbf{k})})|$ *where* $|\cdot|$ *denotes the counting norm.*

Proof. This follows from the fact that any minimizer of (6.2) must satisfy

$$\frac{\partial}{\partial \eta_j}\frac{1}{2}\|\mathbf{u}^{(0)} - \sum_{i=1}^{N-k}\eta_i\nu_{\mathbf{i}}\|_2^2 = 0$$

where we suppress the k-th level notation on $\eta_i^{(k)}$ and $\nu_{\mathbf{i}}^{(\mathbf{k})}$ respectively. Hence,

$$\int_\Omega (\mathbf{u}^{(0)} - \sum_{i=1}^{N-k}\eta_i\nu_{\mathbf{i}})\nu_{\mathbf{j}}d\mathbf{x} = 0.$$

Using inner product notation implies

$$\langle \mathbf{u}^{(0)}, \nu_{\mathbf{j}}\rangle - \sum_{i=1}^{N-k}\eta_i\langle\nu_{\mathbf{i}},\nu_{\mathbf{j}}\rangle = 0.$$

Since $\langle\nu_{\mathbf{i}},\nu_{\mathbf{j}}\rangle = 0$ for $i \neq j$ we arrive at

$$\eta_j = \frac{\langle \mathbf{u}^{(0)},\nu_{\mathbf{j}}\rangle}{\langle\nu_{\mathbf{j}},\nu_{\mathbf{j}}\rangle} = \frac{1}{\|\nu_{\mathbf{j}}\|_2^2}\int_\Omega \mathbf{u}^{(0)}\nu_{\mathbf{j}}d\mathbf{x} = \frac{1}{|\nu_{\mathbf{j}}|}\int_{supp(\nu_{\mathbf{j}})}\mathbf{u}^{(0)}d\mathbf{x}.$$

□

Thus, in this new setting the contrast preserving decomposition of a given image $\mathbf{u}^{(0)}$ at the k-th level becomes

$$\mathbf{u}^{(k)} = \sum_{i=1}^{N-k}\eta_i^{(k)}\nu_{\mathbf{i}}{}^{(k)} \tag{6.4}$$

with the $\eta_i^{(k)}$ defined from (6.3). In particular, image features that have not yet merged retain their exact contrast as originally in $\mathbf{u}^{(0)}$. Moreover, piecewise constant images can be recovered exactly for all λ parameters up to the critical one. These theoretical results will be shown at the conclusion of this section as we wish to emphasize the applications. In short, this method combines the same beneficial attributes of the original PM method in the sense that the geometry and dynamics are the same but with better contrast preservation. Moreover, since the average value of each region is stored at each step of the algorithm, the contrast preserving method retains the $N\log N$ complexity of the original PM method. Hence, to preserve contrast, the idea flow is:

PM-method $\longrightarrow$ TV-basis $\longrightarrow$ Induced basis $\longrightarrow$ L^2 projection of $\mathbf{u}^{(0)}$ onto the induced basis.

We call this new method the *PM–L^2 Projection Method* (PM–L^2).

In practice, the new PMP method does not differ much from the PM method since the average values of each region are stored and updated throughout the merging process. To calculate the projection, one need just omit the update step in the PM algorithm.

6.2. Application to Contrast Preserving Denoising and Decomposition. Let the forward noise model is given by $\mathbf{u}_{obs} = \mathbf{u}_{clean} + \aleph$ with $\mathbf{u}_{obs}$, $\mathbf{u}_{clean}$, and $\aleph$ the observed image, the clean image, and the additive noise respectively then we can choose a $\lambda^{(k)}$ such that the noise $\aleph$ is removed. Then the k-th level decomposition $\mathbf{u}^{(k)}$ at $\lambda^{(k)}$ is the approximation to the clean image $\mathbf{u}_{clean}$ obtained from evolving λ and $\mathbf{u}^{(0)}$ via the PM–L^2 method. Then, the resulting solution at the k-th level, $\mathbf{u}^{(k)}$, has the decomposition

$$\mathbf{u}^{(k)} = \sum_{i=1}^{N-k} \eta_i^{(k)} \nu_{\mathbf{i}}^{(\mathbf{k})} \tag{6.5}$$

where the first k regions are construed as noise and the $\eta_i^{(k)}$'s as in (6.3). Hence the remaining regions should be close to those of the clean image $\mathbf{u}_{clean}$. Moreover, since the $\nu_{\mathbf{i}}{}^{(k)}$'s are derived from the L^2 projection, the contrast of the restored image $\mathbf{u}^{(k)}$ should be close to that of the clean image. Precisely, if $\mathbf{u}^{(k)}$ denotes the decomposition at the k-th level as in (6.5) with regions $\nu_{\mathbf{i}}^{(\mathbf{k})}$ and $\mathbf{u}_{clean}$ the clean image with regions $\nu_{\mathbf{i}}^{*}$ then we denote the contrast of $\mathbf{u}^{(k)}$ and $\mathbf{u}_{clean}$ by $\eta_i^{(k)}$ and η_i^{*} respectively. If we further assume that $\mathbf{u}^{(k)}$ and $\mathbf{u}_{clean}$ have the same number of regions and that the regions $\{\nu_{\mathbf{i}}^{(\mathbf{k})}\}$ and $\{\nu_{\mathbf{i}}^{*}\}$ coincide, then we can bound the difference in intensities

$$\begin{aligned}
&|\eta_i^{(k)} - \eta_i^{*}| \\
&= |\frac{1}{|\nu_{\mathbf{i}}^{(\mathbf{k})}|}\int_{supp(\nu_{\mathbf{i}}^{(\mathbf{k})})} \mathbf{u}^{(0)} d\mathbf{x} - \frac{1}{|\nu_{\mathbf{i}}^{*}|}\int_{supp(\nu_{\mathbf{i}}^{*})} \mathbf{u}_{clean} d\mathbf{x}| \\
&= |\frac{1}{|\nu_{\mathbf{i}}^{(\mathbf{k})}|}\int_{supp(\nu_{\mathbf{i}}^{(\mathbf{k})})} \mathbf{u}_{clean} d\mathbf{x} + \frac{1}{|\nu_{\mathbf{i}}^{(\mathbf{k})}|}\int_{supp(\nu_{\mathbf{i}}^{(\mathbf{k})})} \aleph d\mathbf{x} - \frac{1}{|\nu_{\mathbf{i}}^{*}|}\int_{supp(\nu_{\mathbf{i}}^{*})} \mathbf{u}_{clean} d\mathbf{x}| \\
&= |\frac{1}{|\nu_{\mathbf{i}}^{(\mathbf{k})}|}\int_{supp(\nu_{\mathbf{i}}^{(\mathbf{k})})} \aleph d\mathbf{x}|.
\end{aligned}$$

Here, the last integral is just the average value of the noise $\aleph$ over the support of $\nu_{\mathbf{i}}^{(\mathbf{k})}$. Hence, for denoising applications, if the orthogonal basis of the reconstructed image $\mathbf{u}^{(k)}$ from the PM method coincides with the basis of the clean image $\mathbf{u}_{clean}$ then we can preserve the contrast up to the average value of the noise over each image feature. Therefore, we have just introduced a contrast preserving TV denoising method. Of course in practice, the remaining basis elements of the denoised image may not coincide exactly with that of the original image and depends strongly on the present noise level. However, the closer the remaining basis elements to that of the original image, the better the contrast is preserved. Numerically, for piecewise constant images, even under moderate noise levels, a good approximation to the original basis is restored and the contrast is much better preserved over the standard TV denoising method. In practice, one does not need to explicitly compute the change of basis since the PM algorithm already stores the location of regions and the average intensity value of the original image over these regions.

6.3. Basic L^2 Projection Results.

6.3.1. Mean value preserved by PM–L^2 method. It is proven in [7] that the mean value is preserved for the PM method. We wish to show that the same property is true for the PM–L^2 method. By mean value preservation we mean $\int_\Omega \mathbf{u}^{(0)} d\mathbf{x} = \int_\Omega \mathbf{u}^{(k)} d\mathbf{x}$ for each k with $\mathbf{u}^{(k)}$ as in (6.4). We state this as the following proposition

PROPOSITION 6.2. *Given an initial image* $\mathbf{u}^{(0)}$*, for each* $\mathbf{u}^{(k)}$ *obtained from the PM–*L^2 *Projection method,*

$$\int_\Omega \mathbf{u}^{(0)} d\mathbf{x} = \int_\Omega \mathbf{u}^{(k)} d\mathbf{x} \tag{6.6}$$

Proof. To ease notation, we may assume that for each k there are $N-k$ distinct regions of $\mathbf{u}^{(k)}$. For the case of multiple identical regions, the proof follows in the same manner. Then, at each level k we may form the decomposition

$$\mathbf{u}^{(k)} = \sum_{i=1}^{N-k} \eta_i^{(k)} \nu_{\mathbf{i}}^{(\mathbf{k})} \tag{6.7}$$

with the $\nu_{\mathbf{i}}^{(\mathbf{k})}$ the induced orthogonal basis as in (5.4) and the $\eta_i^{(k)}$ the contrast from the L^2 projection method as in (6.3). Thus,

$$\begin{aligned}
\int_\Omega \mathbf{u}^{(k)} d\mathbf{x} &= \int_\Omega \sum_{i=1}^{N-k} \eta_i^{(k)} \nu_{\mathbf{i}}^{(\mathbf{k})} d\mathbf{x} \\
&= \sum_{i=1}^{N-k} \eta_i^{(k)} \int_\Omega \nu_{\mathbf{i}}^{(\mathbf{k})} d\mathbf{x} \\
&= \sum_{i=1}^{N-k} \frac{1}{|\nu_{\mathbf{i}}^{(\mathbf{k})}|} \int_{supp(\nu_{\mathbf{i}}^{(\mathbf{k})})} \mathbf{u}^{(0)} d\mathbf{x} |\nu_{\mathbf{i}}^{(\mathbf{k})}| \\
&= \sum_{i=1}^{N-k} \int_{supp(\nu_{\mathbf{i}}^{(\mathbf{k})})} \mathbf{u}^{(0)} d\mathbf{x} \\
&= \int_\Omega \mathbf{u}^{(0)} d\mathbf{x}.
\end{aligned}$$

Here, $|\cdot|$ and $supp(\nu_{\mathbf{i}}^{(\mathbf{k})})$ are the same as in Theorem 6.1. The Last line follows from the fact that for all k, $\bigcup_{i=1}^{N-k} supp(\nu_{\mathbf{i}}^{(\mathbf{k})}) = \Omega$. □

7. Future Works.

7.1. L^1 **fitting term algorithm for Image Decomposition:.** In the work by Chan and Esedoglu it is shown that in the continuous setting the L^1 fitting term is useful for image decomposition as features tend to preserve their contrast until a critical parameter λ is reached. Unfortunately there is no efficient way to compute these critical λ values. Our approach is to apply the same methodology of the PM method to create an algorithm that may be associated to the L^1 fitting case.

Acknowledgements. The authors would like to thank Andy M. Yip and Selim Esedoglu for their valuable comments and discussions and are also thankful to Ivy Yip for generously sharing her expertise in data structures.

REFERENCES

[1] R. ACAR AND C. R. VOGEL, *Analysis of bounded variation penalty methods for ill-posed problems*, Inverse Problems, 10 (1994), pp. 1217–1229.

[2] L. Alvarez, F. Guichard, P. Lions, and J. Morel, *Axioms and fundamental equations of image processing*, Arch. Rational Mech. Anal., 123 (1993), pp. 199–257.
[3] Antonin Chambolle, *An algorithm for total variation minimization and applications*, Ceremade technical report 0240, University of Paris IX, Dauphine, France, 2002.
[4] R. H. Chan, T. F. Chan, and W. L. Wan, *Multigrid for differential-convolution problems arising from image processing*, Proc. of the Workshop on Sci. Comput., G. Golub, S. H. Lui, F. Luk and R. Plemmons, eds., Springer-Verlag, 1997, pp. 58–72.
[5] T. F. Chan, G. H. Golub, and P. Mulet, *A nonlinear primal-dual method for total variation-based image restoration*, SIAM J. Sci. Comp., 20:6 (1999), pp. 1964–1977.
[6] T. F. Chan, S. Osher, and J. Shen, *The digital TV filter and nonlinear denoising*, IEEE Trans. Image. Processing, 10 (2001), pp. 231–242.
[7] A. M. Yip and F. Park, *Solution dynamics, causality, and critical behavior of the regularization parameter in total variation denoising problems*, UCLA CAM Report 03-59, Department of Mathematics, University of California, Los Angeles, CA, 2003.
[8] T. F. Chan, H. M. Zhou, and R. H. Chan, *Continuation method for total variation denoising problems*, Proceedings to the SPIE Symposium on Advanced Signal Processing: Algorithms, Architectures, and Implementations, San Diego, CA, F. Luk, ed., vol. 2563, 1995, pp. 314–325.
[9] T. F. Chan, S. Esedoglu, A. Yip, and F. Park , *A Monotonicity Preserving Total Variation Denoising Model*, in preparation.
[10] G. H. Golub and C. F. Van Loan, *Matrix Computations*, Third ed., The Johns Hopkins University Press, Baltimore, MD, 1996.
[11] K. Lange, *Optimization Theory*, Lecture notes, Department of Biomathematics, University of California, Los Angeles, CA, 2002.
[12] J. J. Koenderink, *The Structure of Images*, Biological Cybernetics, 50 (1984), pp. 363-370.
[13] R. J. Randall, *Numerical methods for conservation laws*, Second ed., Birkhauser Verlag, 1992.
[14] M. Lysaker, S. Osher, and X. Tai, *Noise removal using smoothed normals and surface fitting*, UCLA CAM Report 03-03, Department of Mathematics, University of California, Los Angeles, CA, 2003.
[15] Y. Meyer, *Oscillating Patterns in Image Processing and Nonlinear Evolution Equations: The Fifteenth Dean Jacqqueline B. Lewis Memorial Lectures*, The American Mathematical Society, Providence, RI, 2001.
[16] S. Osher, A. Sole, and L. Vese, *Image decomposition and restoration using total variation minimization and the H^{-1} norm*, UCLA CAM Report 02-57, Department of Mathematics, University of California, Los Angeles, CA, 2002.
[17] P. Perona and J. Malik, *Scale-space and edge detection using anisotropic diffusion*, IEEE Transactions on Pattern Analysis and Machine Intelligence, PAMI, 12 (1990), pp. 629–639.
[18] E. Radmoser, O. Scherzer, and J. Weickert, *Scale-space properties of regularization methods*, Scale-space Theories in Computer Vision: Second International Conference, Corfu, Greece, 1999, Mads Nielsen et al, eds., Lecture Notes in Computer Science, vol. 1682, 1999, pp. 211–220, Springer.
[19] L. Rudin, S. Osher, and E. Fatemi, *Nonlinear total variation based noise removal algorithms*, Physica D, 60 (1992), pp. 259–268.
[20] G. Strang, *Wavelets and Dilation Equations: A Brief Introduction*, Siam Review, 31:4 (1989), pp. 614–627.
[21] D. M. Strong and T. F. Chan, *Edge-preserving and scale-dependent properties of total variation regularization*, UCLA CAM Report 00-38, Department of Mathematics, University of California, Los Angeles, CA, 2000.
[22] E. Tadmor, S. Nezzar, and L. Vese, *A multiscale image representation using hierarchical (BV, L^2) decompositions*, UCLA CAM Report 03-32, Department of Mathematics, University of California, Los Angeles, CA, 2003.
[23] C. R. Vogel and M. E. Oman, *Iteration methods for total variation denoising*, SIAM J. Sci. Comp., 17 (1996), pp. 227–238.
[24] C. R. Vogel, *A multigrid method for total variation-based image denoising*, K. Bowers and J. Lund, eds., Computation and Control IV, Birkhauser, 1995.
[25] I. Pollak, A.S. Willsky, and Y. Huang, *Nonlinear Evolution Equations as Fast and Exact Solvers of Estimation Problems*, To appear in IEEE Transactions on Signal Processing, (2004).
[26] A. P. Witkin *Scale-Space Filtering*, Proc. 8th Int. Joint Conf. on Artificial Intelligence, (Karlsruhe, Germany), pp. 1019-1022, 1983.

An N Log N Parameter Marching Algorithm

Input: $\mathbf{u^{(0)}}$, $\lambda > 0$
Output: $\mathbf{u}(\lambda)$
Initialization:
m $\leftarrow$ number of horizontal nodes, n $\leftarrow$ number of vertical nodes
$\lambda^{(0)} \leftarrow 0$, $j \leftarrow 0$, $k \leftarrow 0$
For $i = 1, \ldots, m$
  For $j = 1, \ldots, n$ **do**
   $R_{i,j} \leftarrow \{i, j\}$
   $\alpha_{i,j}^{(0)} \leftarrow {\delta^{(0)}}_{i+\frac{1}{2},j} - {\delta^{(0)}}_{i-\frac{1}{2},j} + {\delta^{(0)}}_{i,j+\frac{1}{2}} - {\delta^{(0)}}_{i,j-\frac{1}{2}}$
  EndFor
EndFor
$N \leftarrow n \times m$ /*number of flat regions*/
For $i = 1, \ldots, m-1$
  For $j = 1, \ldots, n-1$ **do**
    If $u_{i+1,j}^{(0)} = u_{i,j}^{(0)}$
      $t_{i+\frac{1}{2},j} \leftarrow 0$
    ElseIf $u_{i,j+1}^{(0)} = u_{i,j}^{(0)}$
      $t_{i,j+\frac{1}{2}} \leftarrow 0$
    ElseIf $\alpha_{i+1,j}^{(0)} = \alpha_{i,j}^{(0)}$
      $t_{i+\frac{1}{2},j} \leftarrow \infty$
    ElseIf $\alpha_{i,j+1}^{(0)} = \alpha_{i,j}^{(0)}$
      $t_{i,j+\frac{1}{2}} \leftarrow \infty$
    Elseif
      $t_{i+\frac{1}{2},j} \leftarrow \frac{(u_{i+1,j}^{(0)} - u_{i,j}^{(0)})}{(\alpha_{i,j}^{(0)} - \alpha_{i+1,j}^{(0)})}$
    Else
      $t_{i,j+\frac{1}{2}} \leftarrow \frac{(u_{i,j+1}^{(0)} - u_{i,j}^{(0)})}{(\alpha_{i,j}^{(0)} - \alpha_{i,j+1}^{(0)})}$
    EndIf
  EndFor
EndFor
Iteration Step:
While $\lambda^{(k)} < \lambda$ **do:**
  If $N = 1$
    Return A constant vector with entries equal to the mean of $\mathbf{u^{(0)}}$.
  EndIf
  $\{i^*, j^*\} \leftarrow arg\, min_{1 \le i \le m-1,\, 1 \le j \le n-1} \left\{ t_{i+\frac{1}{2},j},\, t_{i,j+\frac{1}{2}} \right\}$
  $T \leftarrow min_{1 \le i \le m-1,\, 1 \le j \le n-1} \left\{ t_{i+\frac{1}{2},j},\, t_{i,j+\frac{1}{2}} \right\}$
  If $T < \lambda$
    If $t_{i^*+\frac{1}{2},j^*} < t_{i^*,j^*+\frac{1}{2}}$
      Merge R_{i^*,j^*} and R_{i^*+1,j^*}, $N \leftarrow N-1$, $m \leftarrow m-1$, re-index all $R_{i,j}$
      Update α_{i^*-1,j^*} and α_{i^*+1,j^*}, delete α_{i^*,j^*}, re-index all $\alpha_{i,j}$
      Update $t_{i^*-\frac{1}{2},j^*}$ and $t_{i^*+\frac{3}{2},j^*}$, delete $t_{i^*+\frac{1}{2},j^*}$, re-index all $t_{i+\frac{1}{2},j}$
    Else
      Merge R_{i^*,j^*} and R_{i^*,j^*+1}, $N \leftarrow N-1$, $n \leftarrow n-1$, re-index all $R_{i,j}$
      Update α_{i^*,j^*-1} and α_{i^*,j^*+1}, delete α_{i^*,j^*}, re-index all $\alpha_{i,j}$
      Update $t_{i^*,j^*-\frac{1}{2}}$ and $t_{i^*,j^*+\frac{3}{2}}$, delete $t_{i^*,j^*+\frac{1}{2}}$, re-index all $t_{i,j+\frac{1}{2}}$
    EndIf
  EndIf
  $k \leftarrow k+1$
  $\lambda^{(k)} \leftarrow \min\{T, \lambda\}$
EndWhile
  Compute $\mathbf{u}^{(k)}$ by $\mathbf{u}^{(k)} = a_{i,j}^{(k)} + \lambda^{(k)} \alpha_{i,j}^{(k)}$
Return $\mathbf{u^{(k)}}$

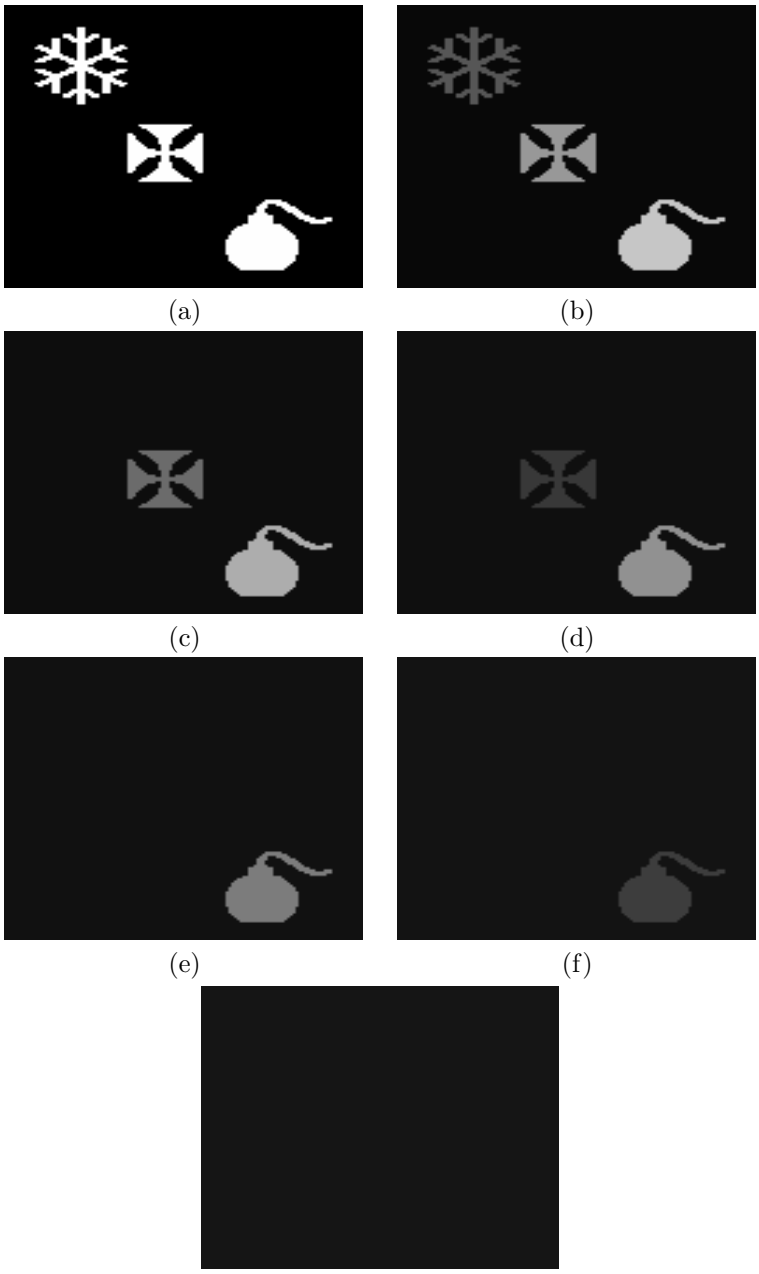


FIG. 5.1. *Full decomposition of an image by the proposed PM method. Given an initial image (a), we have in (b)–(e) decompositions with* $\lambda = 130$, $\lambda^{(1)} = 194$, $\lambda = 270$, $\lambda^{(2)} = 312$, $\lambda = 500$, *and* $\lambda^{(3)} = 560$ *respectively. Note how the regions merge with the background in each stage cleanly without any visible artifacts nor geometric changes to the remaining regions. We also note that in (b) and (c) the snowflake region is the first to merge as its area to perimeter ratio is the smallest in this group. The cross and the firecracker subsequently follow. The values* $\lambda^{(1)}$, $\lambda^{(2)}$, *and* $\lambda^{(3)}$ *are critical values respectively while the values* $\lambda = 160$, $\lambda = 250$, *and* $\lambda = 450$ *respectively are intermediate* λ *values.*

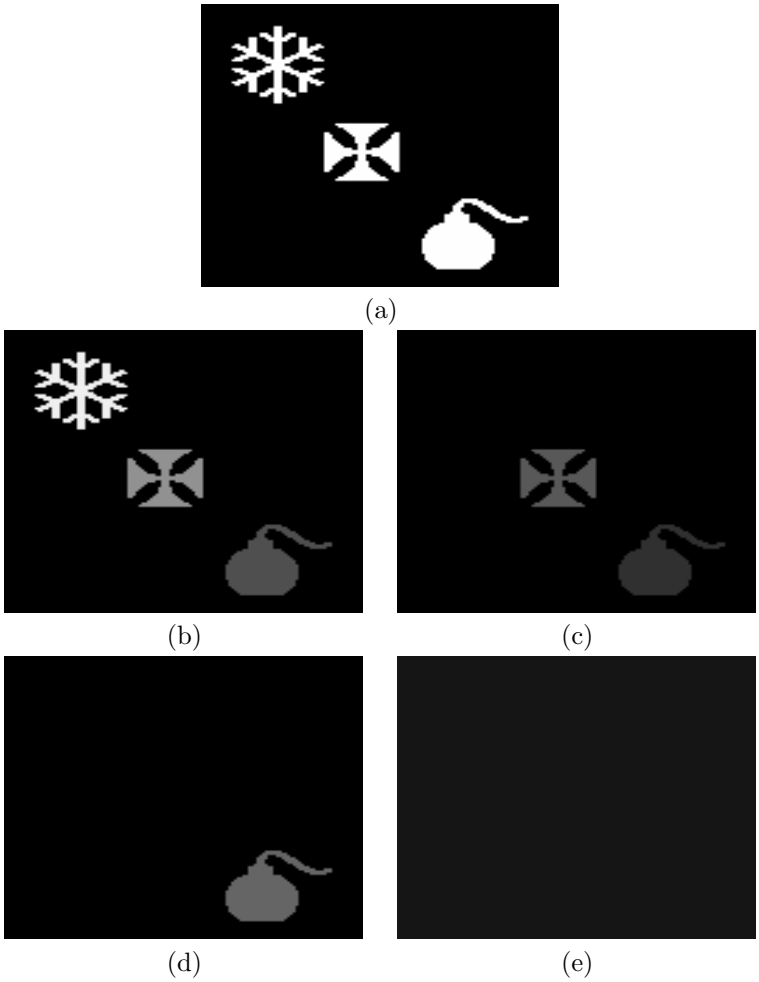


FIG. 5.2. *Decomposition into* 4 *summands, (a) = (b) + (c) + (d) + (e) by using the PM method. In (a) we have the initial data while in (b)–(d) we exhibit the corresponding residuals*+180 *at* $\lambda^{(1)}$, $\lambda^{(2)}$ *and* $\lambda^{(3)}$ *respectively. Due to the global contrast loss, the residuals contain the desired removed image feature along with some contrast from the other features. However, the amount of contrast removed remains constant with respect to each feature and each feature being removed at each stage has clearly has a higher contrast than the other features. For example, in (b) the snowflake is the first feature to be removed, hence, it has a higher contrast than the cross and the firecracker. We also note that there are no geometric changes to any of the regions.*

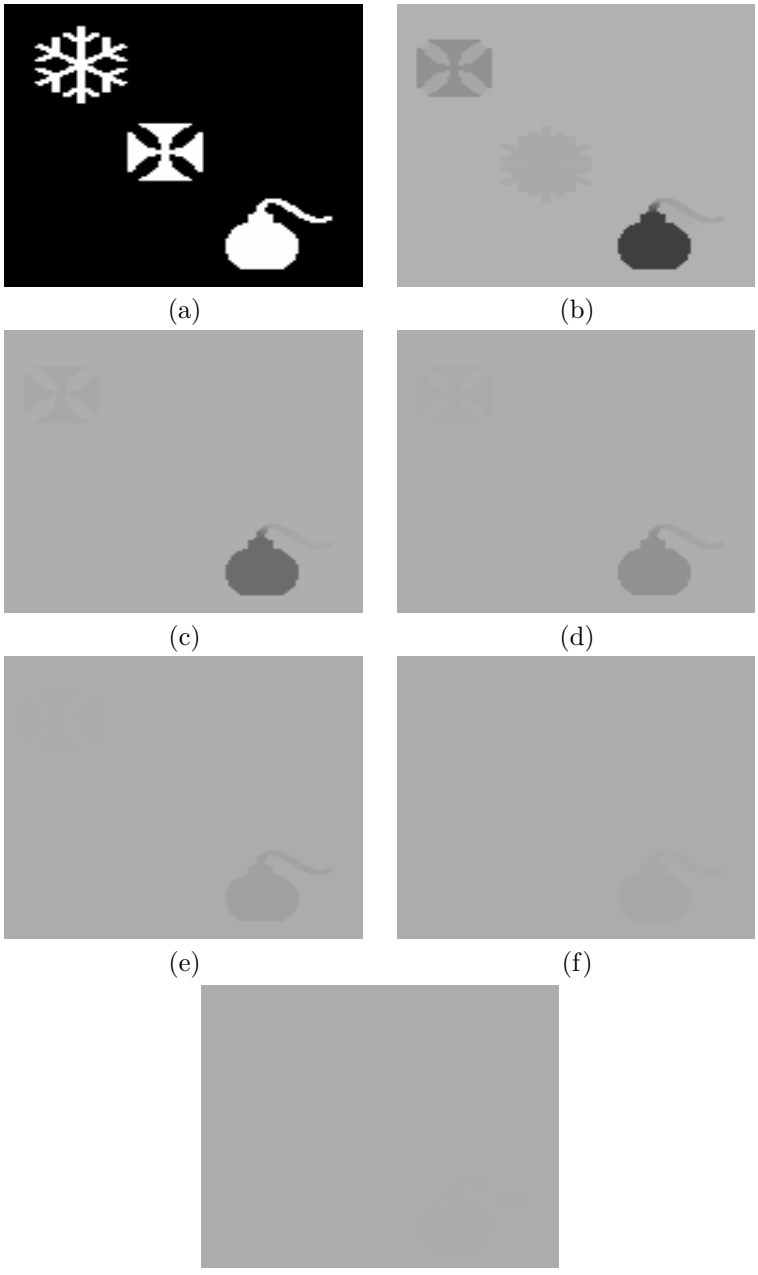


FIG. 5.3. *Decompositions of an image by solving the Euler-Lagrange equation for the TV model for various λ's. Given an initial image (a), we have in (b)–(g) decompositions with $\lambda = 300$, $\lambda^{(1)} = 460$, $\lambda = 650$, $\lambda^{(2)} = 900$, $\lambda = 1400$, and $\lambda^{(3)} = 1800$ respectively. Note how the region merging is not as clean as with the PM method in Figure 5.1. The features tend to diffuse with the background making the critical λ values, which correspond to the merging, difficult to define. Particularly in (b) we observe this behavior as the snowflake's shape is smeared during the merging process. Moreover, in (b) as well, the cross has artifacts appearing and the firecracker has lost some of its wick. The values $\lambda = 300$, $\lambda = 650$, and $\lambda = 1400$ respectively are intermediate λ values.*

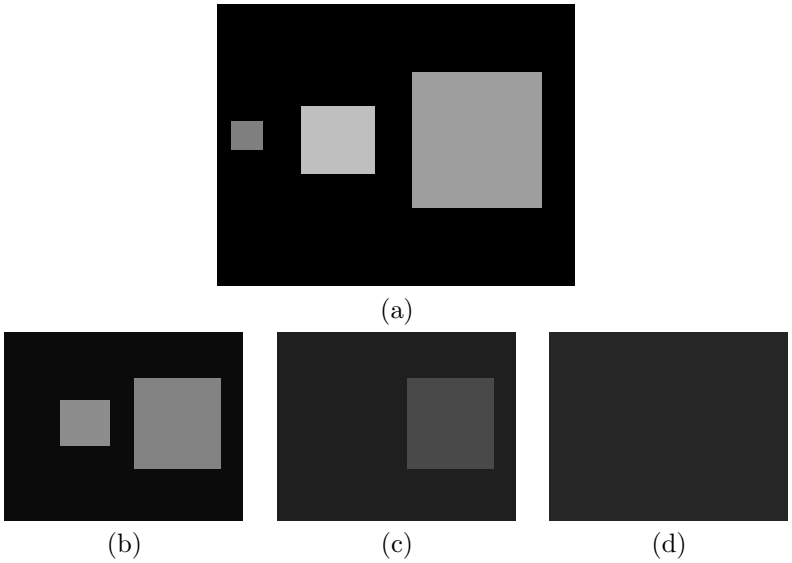


FIG. 5.4. *In (a) we have the original image* $\mathbf{u}^{(0)}$ *consisting of 3 rectangles on a dark background. In (b)–(d), the merging of regions from the PM method from fine scale (small rectangle) to large scale (largest rectangle). Here, the critical* λ*'s calculated from the method are given by* $\lambda^{(1)} = 190$*,* $\lambda^{(2)} = 600$*, and* $\lambda^{(3)} = 842$*.*

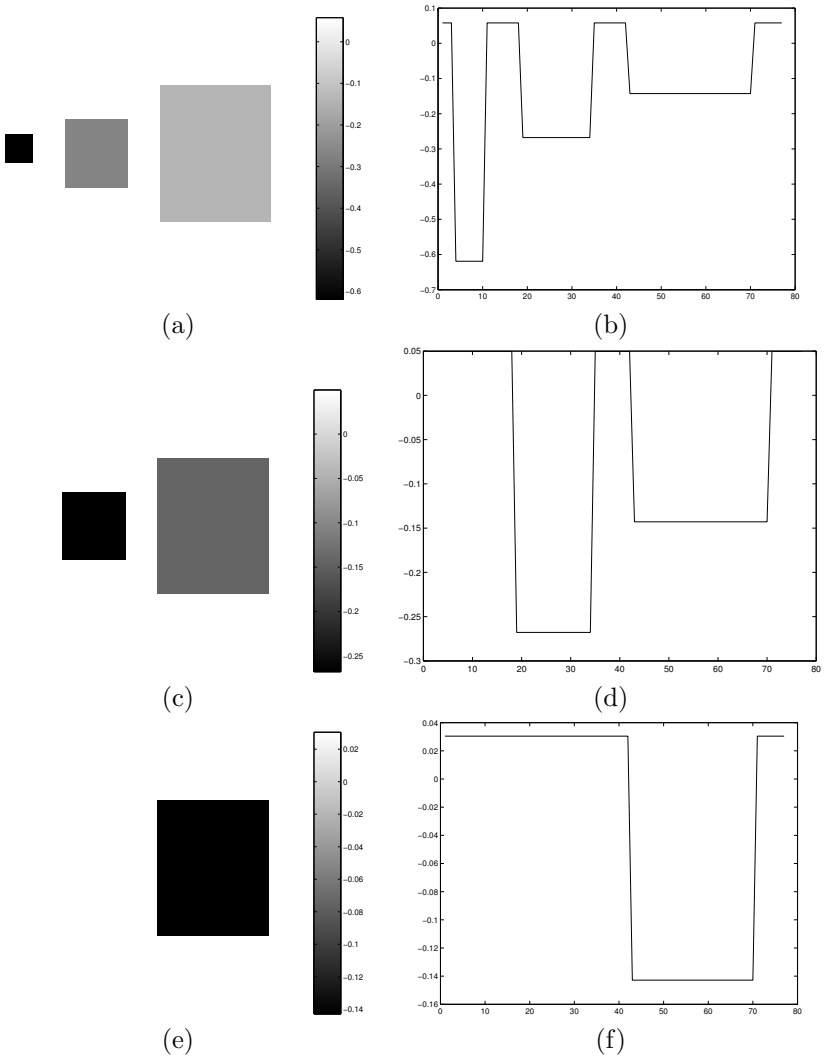


FIG. 5.5. *Global basis at each k-th level, $k = 1, 2, 3$, defined by the monotonicity constrained TV norm. These are global in the sense that at the k-th level, each basis $\alpha^{(\mathbf{k})}$ has information about the speed, and direction of each non merged region $R_i^{(k)}, i = 1, \ldots, 3-k$ at that level. The critical value $\lambda^{(k)}$ determines the features to be merged at the k-th level. (a), (c) and (e) are the 2-d $\alpha^{(\mathbf{1})}$, $\alpha^{(\mathbf{2})}$, and $\alpha^{(\mathbf{3})}$ basis vectors respectively. (b), (d), and (f) represent the cross sections of $-\alpha^{(\mathbf{1})}$, $-\alpha^{(\mathbf{2})}$, and $-\alpha^{(\mathbf{3})}$ respectively. Here, we see that each non background region has negative speed (value) so they will drop to the background in subsequent steps. Moreover, the smallest region which has the smallest area to perimeter ratio (small scale) drops the fastest hence has the highest peak in the cross section image (b).*

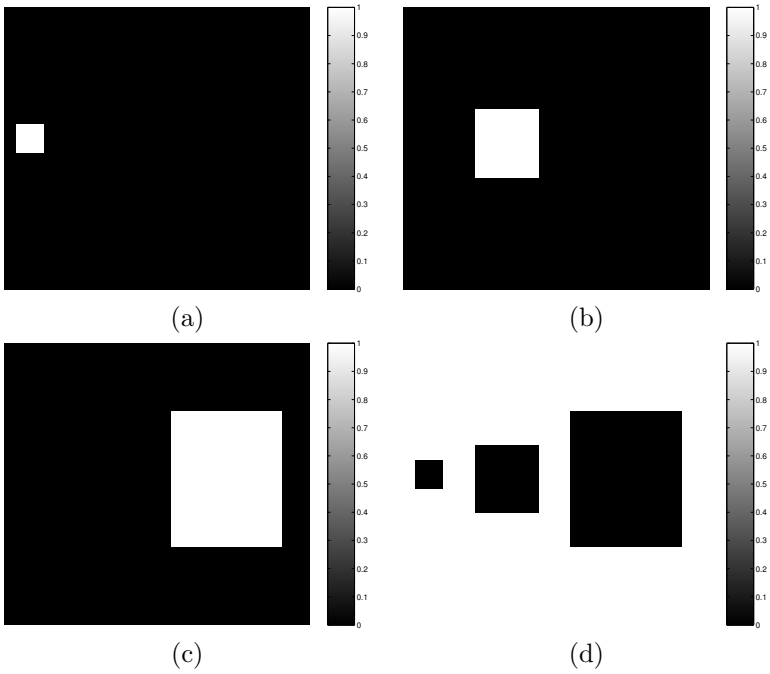


FIG. 5.6. *The induced basis vectors, $\nu_i^{(0)}, i = 1, 2, 3$, defined from the decomposition at the finest level $\mathbf{u}^{(0)}$. In one sweep of the PM algorithm, we can generate the ordering of merging defined by the $\lambda^{(k)}$'s and the corresponding i-th regions, $R_i^{(k)}$ to be merged at the k-th level. Hence at the k-th level, we may define the induced basis vectors $\nu_i^{(k)}, i = 1, \ldots, N-k$ ordered corresponding to their subsequent merging at later levels.*

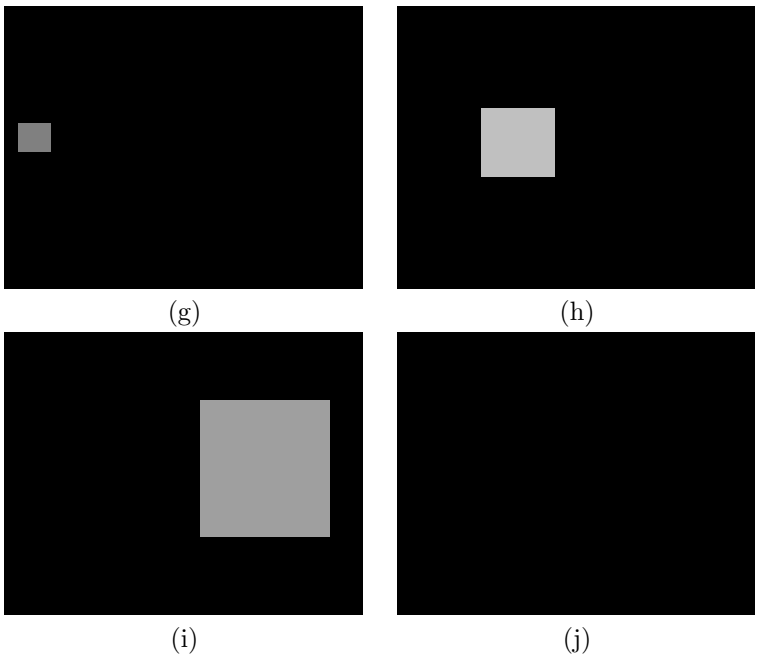


FIG. 5.7. *The image features extracted from the given image $\mathbf{u}^{(0)}$ from fine to coarse scale.*

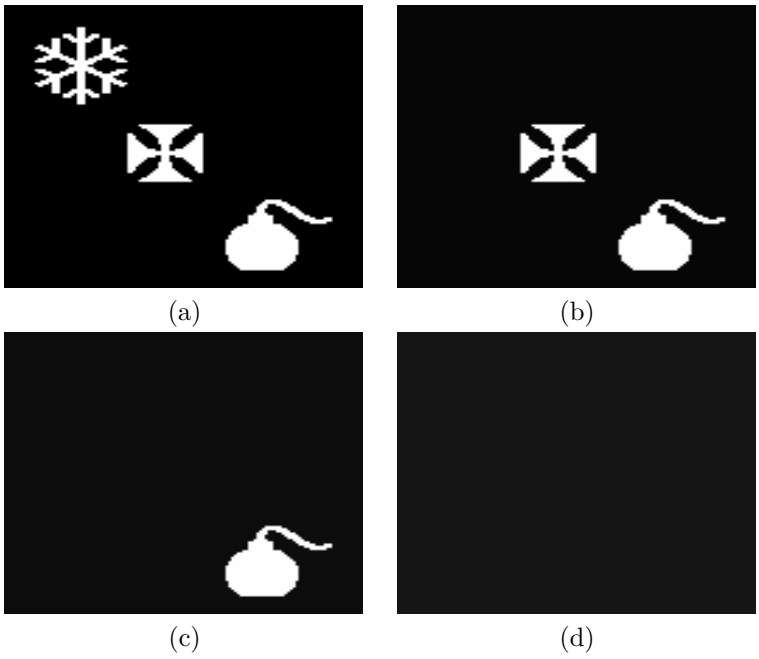


FIG. 6.1. *Full decomposition of an image by the PM–L^2 Projection method. Given an initial image (a), we have in (b)–(d) decompositions with $\lambda^{(1)} = 194$, $\lambda^{(2)} = 312$, and $\lambda^{(3)} = 560$ respectively. The regions merge with the background in each stage cleanly without any visible artifacts nor geometric changes to the remaining regions. The merging proceeds in the same fashion as with the PM method viewed in Figure 5.1, however, the contrast of each remaining feature is exactly preserved. The values $\lambda^{(1)}$, $\lambda^{(2)}$, and $\lambda^{(3)}$ are the critical values respectively.*

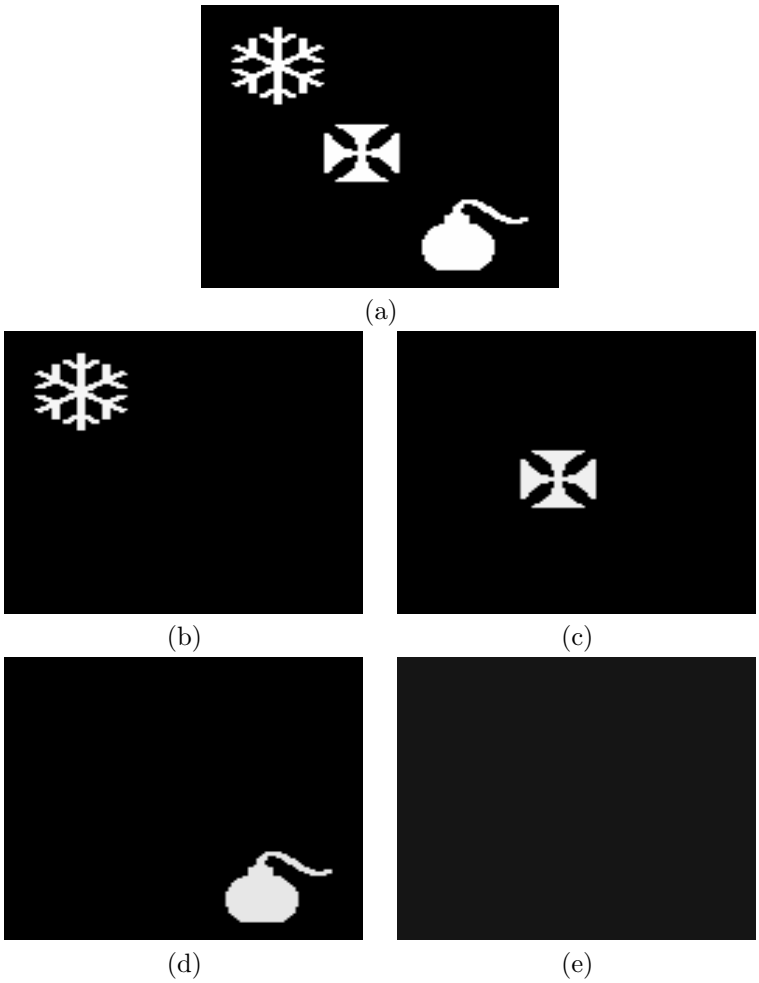
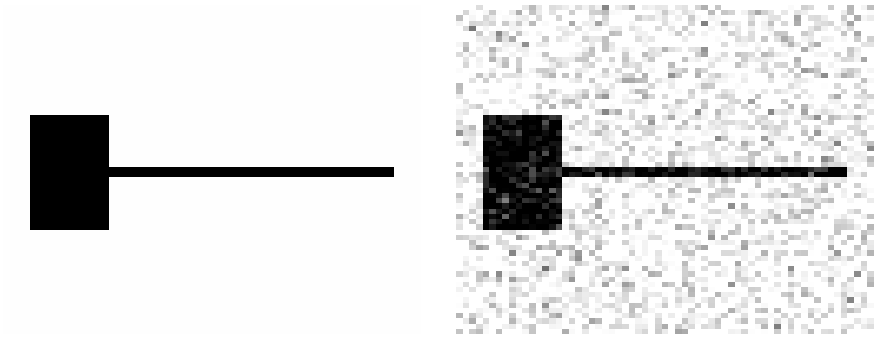
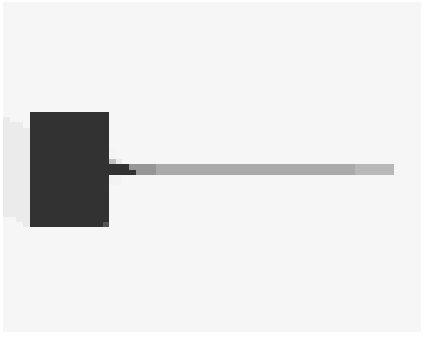


FIG. 6.2. *Decomposition into* 4 *summands, (a) = (b) + (c) + (d) + (e) by the PM–L^2 method. In (b)–(d) we have the residuals*+185 *at* $\lambda^{(1)}$, $\lambda^{(2)}$ *and* $\lambda^{(3)}$ *respectively. Note that in each summand, the features are more cleanly extracted as compared to those resulting from the PM method observed in Figure 5.2. This is due to the exact contrast preservation of each image feature in the decompositions and allows for a "cleaner" decomposition since the features are better isolated.*

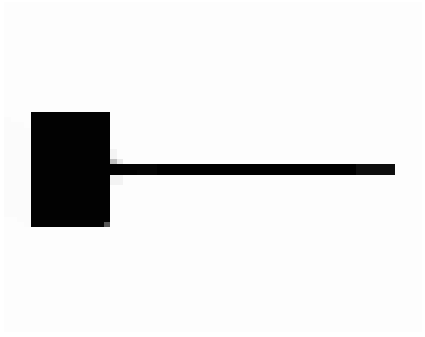


(a)

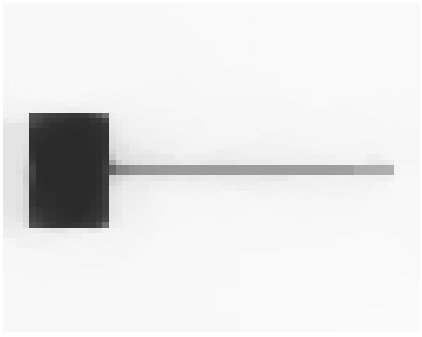
(b)



(c)



(d)



(e)

FIG. 6.3. *In (a) we have a clean* 63×63 *synthetic hammer image. The noisy image is shown in (b) with signal to noise ratio,* $SNR = 5$*. In (c) we have a denoised image via the PM method and in (d) the denoised image via PM-*L^2 *Projection method. We have in (e) the denoised image by solving the Euler-Lagrange equations from the original TV model. Clearly, the contrast is better preserved in (d) over the images in (c) and (e). In (c)–(e)* $\lambda = 170$*.*

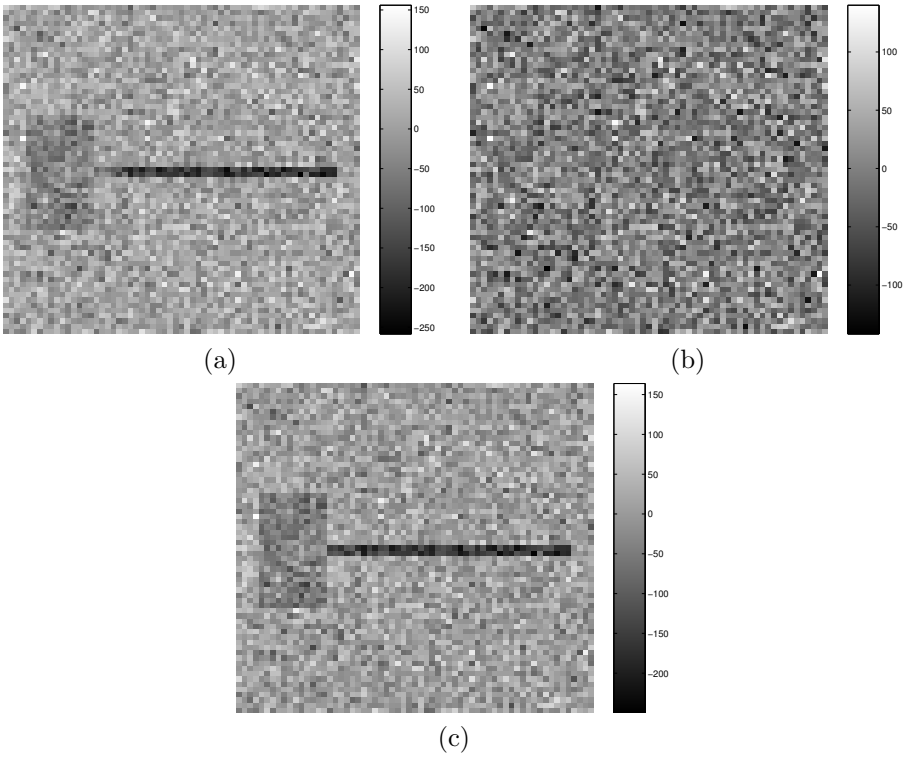


FIG. 6.4. *Here, we have the residuals defined as* $\mathbf{u}_{obs} - \mathbf{u}_{restored}$. *In (a) and (c) the contrast of the image shows up in the residual while in (b) it appears as only noise hence the contrast of the restored image is preserved.*

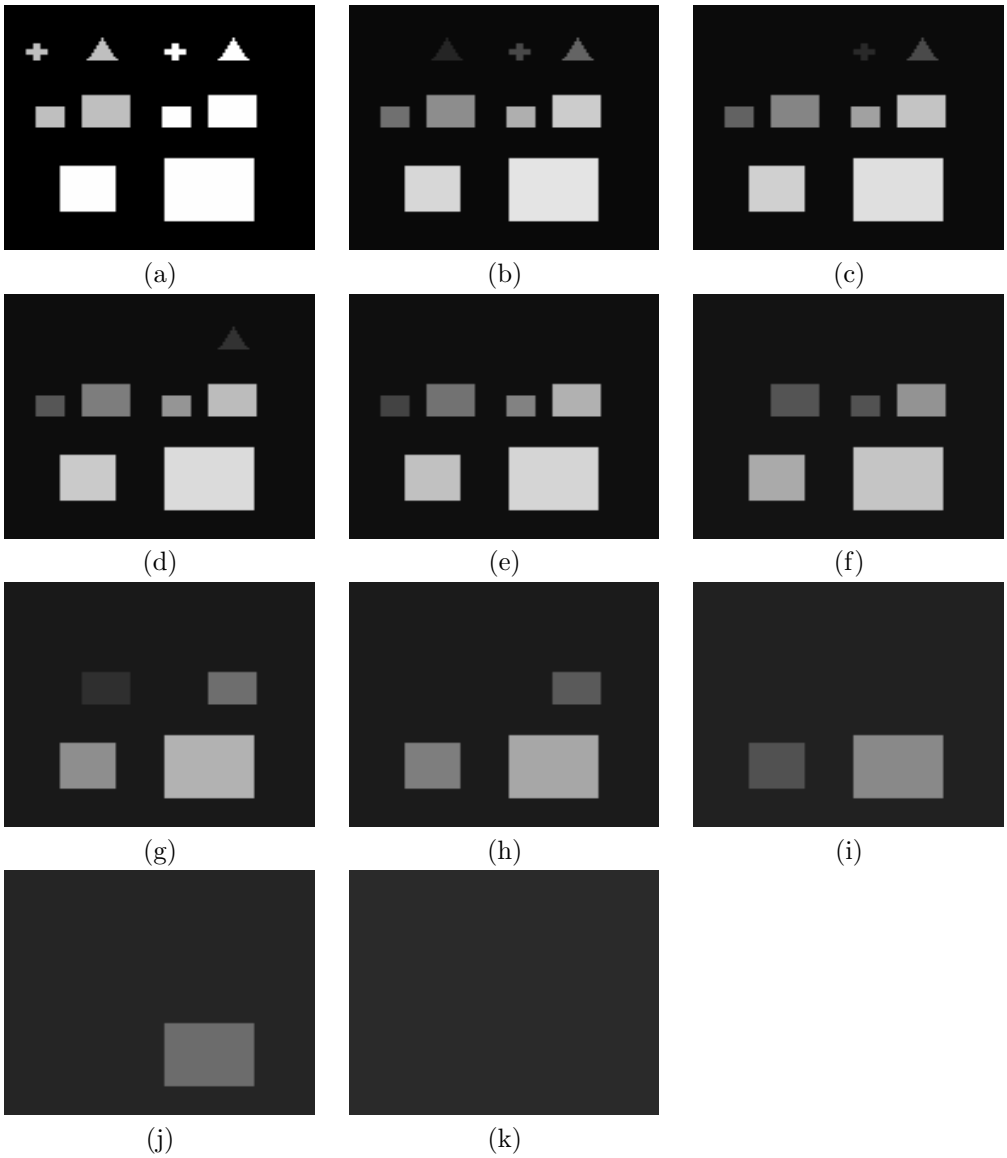


FIG. 6.5. *Full decomposition of an image by the PM algorithm. (a) Given initial image. (b)–(k) Decompositions with* $\lambda^{(1)} = 227$, $\lambda^{(2)} = 266$, $\lambda^{(3)} = 302$, $\lambda^{(4)} = 354$, $\lambda^{(5)} = 492$, $\lambda^{(6)} = 658$, $\lambda^{(7)} = 751$, $\lambda^{(8)} = 1014$, $\lambda^{(9)} = 1272$, *and* $\lambda^{(10)} = 1843$ *respectively. Note how the regions merge with the background in each stage cleanly without any visible artifacts nor geometric changes to the remaining regions. We also note that in (a)–(g) the small scale features merge first followed by the large scale features in (h)–(k)*

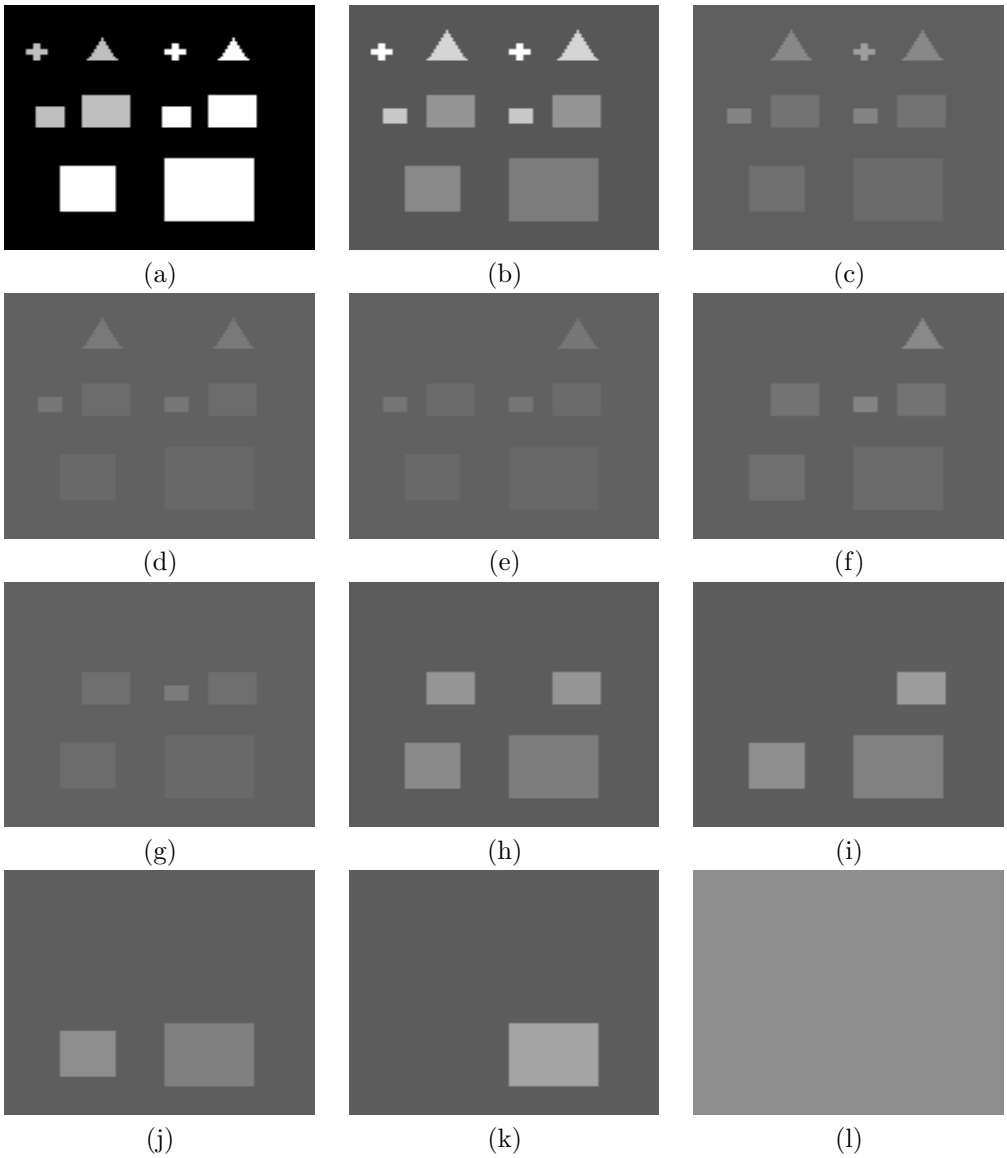


FIG. 6.6. *Decomposition into* 11 *summands, (a) = (b) + (c) + (d) + (e) + (f) + (g) + (h) + (i) + (j) + (k) + (l) by using the PM method. In (b)–(k) we have the residuals*+100 *of the PM method. The constant image of the mean value of (a)*+100 *is exhibited in (l).*

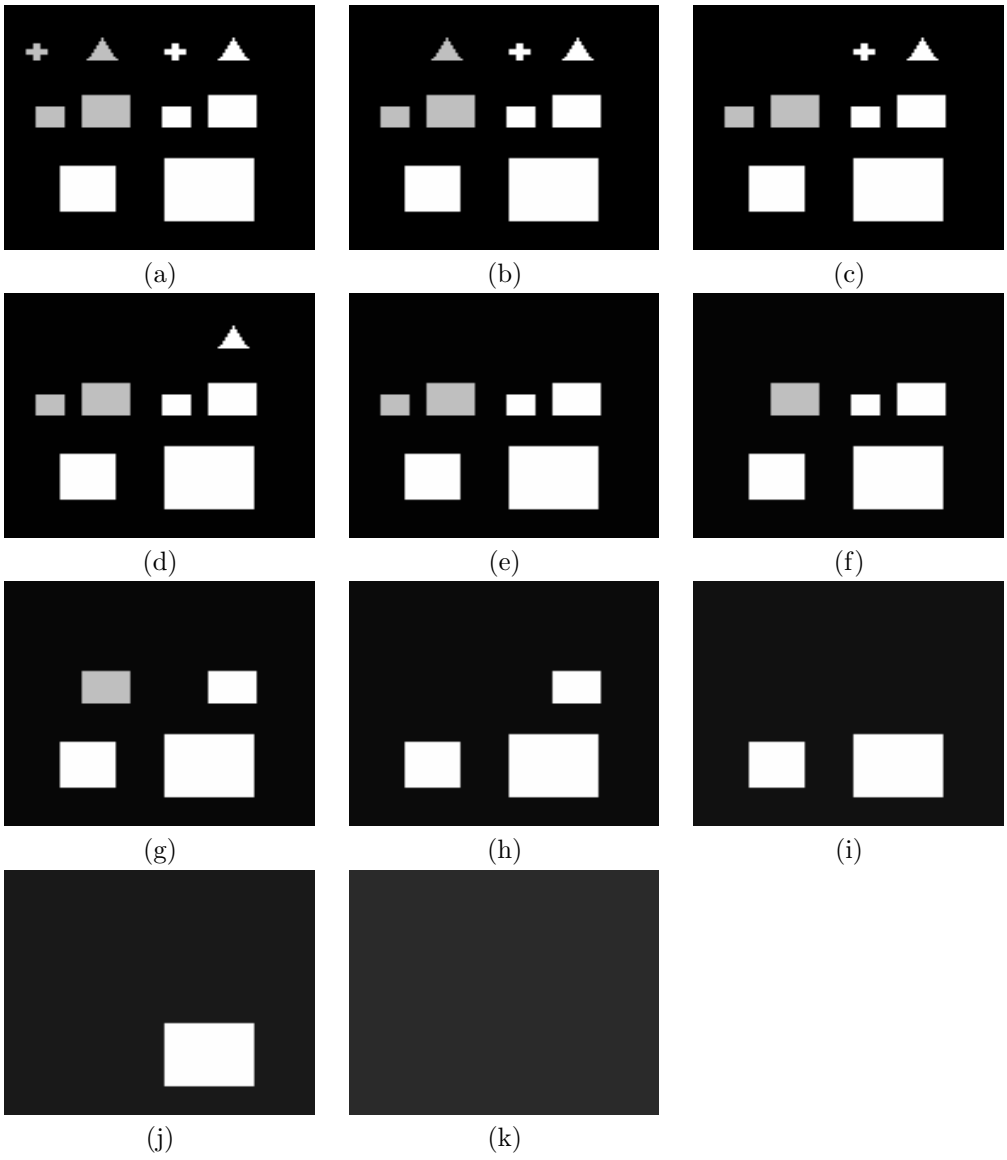


FIG. 6.7. *Full decomposition of an image by the PM–L^2 projection algorithm. (a) Given initial image. (b)–(k) Decompositions with* $\lambda^{(1)} = 227$, $\lambda^{(2)} = 266$, $\lambda^{(3)} = 302$, $\lambda^{(4)} = 354$, $\lambda^{(5)} = 492$, $\lambda^{(6)} = 658$, $\lambda^{(7)} = 751$, $\lambda^{(8)} = 1014$, $\lambda^{(9)} = 1272$, *and* $\lambda^{(10)} = 1843$ *respectively. Note how the regions merge with the background in each stage cleanly without any visible artifacts nor geometric changes to the remaining regions. The contrast is clearly preserved over the PM decomposition while regions are merged according to the same geometric criteria as in the PM case. We also note that in (a)–(g) the small scale features merge first followed by the large scale features in (h)–(k)*

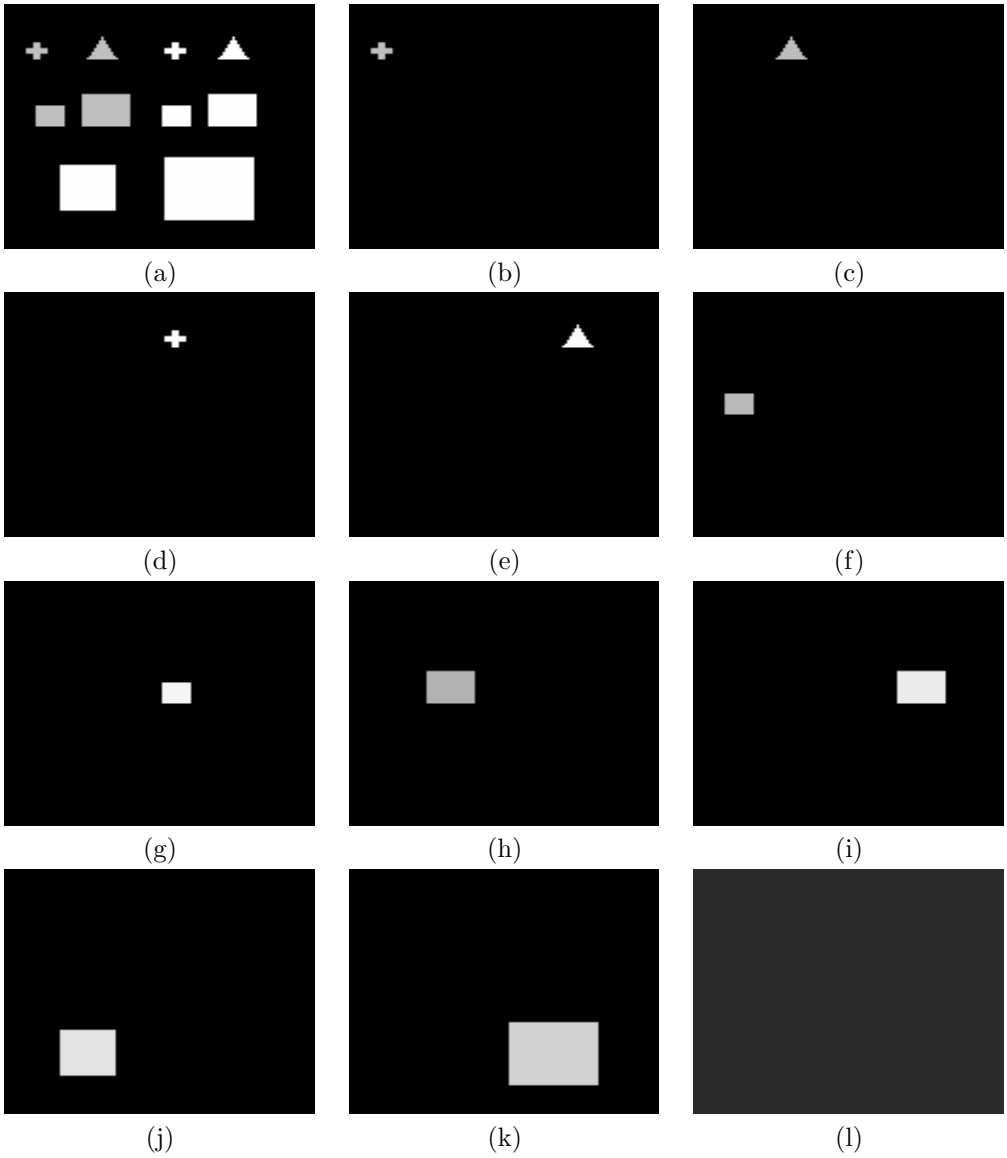


FIG. 6.8. *Decomposition into* 11 *summands, (a) = (b) + (c) + (d) + (e) + (f) + (g) + (h) + (i) + (j) + (k) via the PM–L^2 method. In (b)–(j) we have the residuals of the PM–L^2 projection method. Note that the image features in each summand are cleanly removed. This is due to the preservation of contrast with this method.*

(a) (b)

FIG. 6.9.

(a) (b)

(c)

FIG. 6.10.