

Spatial Kernel K-Harmonic Means Clustering for Multi-spectral Image Segmentation

Qi Li, Nikolaos Mitianoudis and Tania Stathaki

Communications and Signal Processing Group

Imperial College London

Exhibition Road, SW7 2AZ, London, UK

Email: n.mitianoudis@imperial.ac.uk

Abstract

The problem of image segmentation using intensity clustering approaches has been addressed in the literature. Grouping pixels of similar intensity to form clusters in an image has been tackled using a number of methods, such as the *K-Means* (KM) algorithm. The *K-Harmonic Means* (KHM) was proposed to overcome the sensitivity of KM to centre initialisation. In this paper, we investigate the use of a *Spatial Kernel-based KHM* (SKKHM) algorithm on the problem of *image segmentation*. Instead of the original Euclidean intensity distance, a robust Kernel-based K-Harmonic Means metric is employed to reduce the effect of outliers and noise. Spatial image information is also incorporated in the proposed clustering scheme, derived from *Markov Random Field* (MRF) modelling. An extension of the proposed algorithm to multi-spectral imaging applications is also presented. Experimental results on both single-channel and multi-channel images demonstrate the robust performance of the proposed SKKHM algorithm.

I. INTRODUCTION

Image segmentation is one of the most important tasks in image analysis. The main goal is to divide the image into parts of strong correlation with objects or areas of the real world contained in the image [15]. Segmentation can be *complete* or *partial*, if it results to a set of disjoint regions that correspond *directly* or *partially* to image objects. For *partial segmentation*, the image is divided into separate regions that are homogenous with respect to a chosen property such as *brightness*, *color*, *reflectivity* and *texture*. Higher level processing can join the partially segmented image and more consistent objects and areas, giving a more complete segmentation.

Many segmentation methods have been proposed in the literature. *Edge-based segmentation* methods perform segmentation, based on the information conveyed by the edges that exist in an image [15]. Supplementary processing steps must follow to combine edges into edge chains (borders) in order to form coherent objects in the image. *Region-based segmentation* methods perform segmentation by dividing an image into zones of maximum *homogeneity*. The criteria for homogeneity can be based on gray-level, color, texture, shape information. Several criteria can then be employed to force *region growing*, in order to form consistent objects in the image. It is very important to note that edge-based and region-based methods may produce different segmentation results, therefore, a combination of the segmentation results is often used, as the objective is to form the most accurate segmentation map.

In the case of simpler images, where objects may not interfere with each other and their intensity levels are clearly distinct from background intensities, then *intensity clustering approaches* can produce effective segmentation results [15]. These are also known as *thresholding methods* and are less computationally expensive than common segmentation approaches, however, can produce efficient results only in the above mentioned cases. The choice of optimal thresholds that can divide the distribution of pixels into desired segments is important. Information of interest may either be lost or extraneous background pixels may be classified as target information with an improper choice of threshold. In this paper, the use of clustering algorithms to address the problem of automated image segmentation is addressed.

Several clustering methods exist in the literature [8] and can often be categorised as hierarchical, graph theoretic, mode searching, mixture resolving and error-function minimisation approaches [8], [1]. The *K-Means* (KM) algorithm is one of the early centre-based clustering techniques [6], based on L_2 - *norm* minimisation, that has been applied to image segmentation. It adaptively assigns each pixel to a cluster (segment) in a closest-distance fashion. The means of these clusters are then set as the new centres. The procedure is repeated until the sum of a dissimilarity measure, such as the Euclidean distance between centres and pixels, converges to a local minimum. The probability of each data point belonging to different clusters is usually referred to as the *membership function*. In the case of KM, the membership function can only take 0 or 1 values, implying that a data point can only belong to one cluster. In this case, a *hard* membership function is implied and such detection schemes are often called “winner-takes-all” approaches.

The KM algorithm has been widely applied to clustering problems mainly due to its simplicity and efficiency. However, KM suffers from several major problems when used for unsupervised clustering [6]. Firstly, the clustering performance of the KM algorithm is greatly dependent on the *initial guess of cluster centres*. The main reason is that each data point is given equal weight during the adaptation. It

is very likely that more pixels will be attributed to dense centres, and that the algorithm will converge to a suboptimal local minimum. A wide range of clustering results can be produced due to different initialisation. Secondly, the KM algorithm employs the *standard Euclidean distance*, which increases the algorithm's sensitivity to noise and outliers, according to Huber [7]. Finally, in the original version of KM each pixel is treated as an independent data point. Intensity alone may not always be capable of representing the image's features alone, as the relative position of each pixel to its neighbours is important.

To address the above problems, a wide range of solutions have been proposed in the literature. To solve the problem of initialisation, the *K-Harmonic Means* (KHM) algorithm [21] or the *Fuzzy C-Means* (FCM) algorithm [3], [1] introduced a dynamic weight scheme that allows the same pixels to belong to different clusters simultaneously. The difference between the two approaches is that FCM performs clustering minimising the mean weighted euclidean distance of the data points and the centres, whereas the KHM minimised the mean weighted harmonic distance. To improve the robustness to noise and outliers of clustering methods, the Euclidean distance was substituted by kernel functions [19], [1], [5]. Kernel-based distance measures enable modelling of non-Euclidean structures in the data, suppress outliers that are amplified by the L_2 -norm and finally retain low computational complexity [1]. Finally, there have been several approaches to include information about the spatial neighbourhood of each pixel in the classification procedure [19], [17], [14], in order to exploit the image's spatial features and perform more accurate segmentation. A common approach [14], [1] is to penalise the clustering objective function in order to discourage undesirable results in the membership functions, such as high membership values for pixels, surrounded by low membership values and vice versa.

This paper extends the KHM algorithm with solutions to all the above-mentioned problems. This is performed by incorporating kernel-induced distance measures and neighbourhood information into the KHM algorithm. The standard Euclidean intensity distance is replaced by a Mercer kernel-based metric [5], [1] in the KHM algorithm. Markov Random Fields (MRF) [20] are used to model and incorporate the spatial homogeneous information into the kernel-based KHM. The proposed scheme will be referred to as the *Spatial Kernel-based K-Harmonic Means* (SKKHM) clustering scheme. Moreover, considering the increasing number of multi-spectral imaging applications [9], [13], [16], an extension of the SKKHM algorithm to the multivariate case is presented. In this work, we examine the case of standard color Red-Green-Blue (RGB) images, as an example of multivariate signals. The multi-spectral SKKHM algorithm delivers an enhancement to the performance of individual segmentation of each channel.

The rest of the paper is organized as follows: section II presents the original KHM algorithm. The

two proposed improvements to KHM are subsequently presented: the kernel method (section III) and the spatial modelling of neighbourhood using MRF (section IV). In section V, we extend the previous scheme in the case of multivariate signals. Finally, the performance of the proposed method is evaluated and compared with other similar methods in section VI.

II. THE K-HARMONIC MEANS ALGORITHM

A. Definition

Assume an image $I(k, l)$ of size $M \times N$. All the pixel values $x_i = I(k, l)$ will be treated independently to form the data set $X = \{x_1, \dots, x_i, \dots, x_n\}$, where $n = M \times N$ is the total number of pixels and $i = 1, \dots, n$. Let also $\underline{c} = [c_1 \dots c_j \dots c_m]^T$ be a vector containing the position of each cluster centre and m is the number of centres. The KHM algorithm [21] attempts to find an optimal clustering for the data set X , by minimising the K-Harmonic Means of Euclidean intensity distances d_{ij} between each image pixel x_i and the cluster centre c_j . The cost function using K-Harmonic Means is expressed in (1), where $d_{ij} = \|x_i - c_j\|^2$ ($\|\cdot\|$ is the L_2 -norm).

$$Q(\underline{c}) = \sum_{i=1}^n \frac{m}{\sum_{j=1}^m \frac{1}{d_{ij}}} \quad (1)$$

Since X is a scalar set, the intensity distance can be simplified to $d_{ij} = (x_i - c_j)^2$. In the case of image segmentation, we aim to classify the image pixels into different clusters and therefore we usually choose $m \geq 2$. To derive the recursive updating rule for each clustering centre, we set the partial derivative of $Q(\underline{c})$, as shown in (1), with respect to c_j equal to zero.

$$\frac{\partial Q(\underline{c})}{\partial c_j} = 0, \forall j = 1, \dots, m \Rightarrow \quad (2)$$

$$\frac{\partial Q(\underline{c})}{\partial c_j} = \sum_{i=1}^n \frac{-2m(x_i - c_j)}{(\sum_{j=1}^m \frac{1}{d_{ij}})^2 d_{ij}^2} = 0 \quad (3)$$

Re-arranging (3) gives us the next estimate for each centre c_j^+ in (4):

$$c_j^+ \leftarrow \frac{\sum_{i=1}^n \frac{x_i}{d_{ij}^2 (\sum_{j=1}^m \frac{1}{d_{ij}})^2}}{\sum_{i=1}^n \frac{1}{d_{ij}^2 (\sum_{j=1}^m \frac{1}{d_{ij}})^2}} \quad (4)$$

The full derivation of the above equations can be found in Appendix A. Consequently, the KHM algorithm [21] can be summarised as follows: starting from an arbitrary guess of the centre values \underline{c} , we repeat the following steps.

- 1) Update the location of each cluster centre according to (4).

2) Re-estimate the cost function $Q(\underline{c})$ and repeat the above step until the cost function converges to a minimum.

3) Finally, classification is performed, by assigning each pixel to the closest cluster centre.

A possible discontinuity arises in the case that $d_{ij} \rightarrow 0$, when a data point is close to the centre. To tackle this problem, we replace d_{ij} with $\max(d_{ij}, \epsilon)$, where ϵ is a small constant (e.g. $\epsilon = 0.01$) [21], [6].

B. Introducing the membership and the weight function

In [6], Hamerly and Elkan introduced the *membership* function u_{ij} and the *weight* function w_i in the Harmonic KM algorithm.

The *membership* function u_{ij} measures the probability of each data point x_i being assigned to cluster j . For the original KM algorithm, each pixel can only be assigned to one closest cluster. In this case, we have that $u_{ij} = 1$, only if $j = \operatorname{argmin}_j \|x_i - c_j\|^2$. In the case of the KHM algorithm, the scheme of soft decisions implies that each pixel can belong to many clusters to some extent. Effectively, u_{ij} measures the probability of x_i being assigned to cluster j and should always satisfy the following properties:

- 1) $\sum_{j=1}^m u_{ij} = 1$
- 2) $0 \leq u_{ij} \leq 1$

According to [6], the estimate for u_{ij} at each iteration can be calculated, using the following formula:

$$u_{ij}^+ \leftarrow \frac{1/d_{ij}^2}{\sum_{j=1}^m 1/d_{ij}^2} \quad (5)$$

The *weight* function w_i quantifies the amount of influence the data point x_i has in the next estimate of cluster centres [6]. In contrast to the K-Means algorithm, which gives equal weight to all data (i.e. $w_i = 1$), the KHM algorithm uses a varying weight scheme during the estimation of cluster centres. The main idea is to attribute larger weights to pixels whose intensity is far distinct from the estimated cluster centres and smaller weights to pixels with similar intensity to cluster centres. This method ensures that data-points, that were not “well-covered” by the current solution, will be emphasised in the next centre update. Obviously, these weights should always be positive, i.e. $w_i > 0$. We can calculate the estimate for w_i at each iteration using the following formula [6]:

$$w_i^+ \leftarrow \frac{\sum_{j=1}^m 1/d_{ij}^2}{\left(\sum_{j=1}^m \frac{1}{d_{ij}}\right)^2} \quad (6)$$

Once the *membership* function u_{ij} and the *weight* function w_i were introduced, the centre update rule can be reformulated as a function of these parameters. Hence, relating (5) and (6) to (4), we can rewrite the centre function in the following form [6]:

$$c_j = \frac{\sum_{i=1}^n u_{ij} w_i x_i}{\sum_{i=1}^n u_{ij} w_i} \quad (7)$$

Thresholding is finally performed by updating the membership and weight functions, rather than the actual centre positions. The image can be then segmented by assigning each pixel x_i to cluster j with the maximum membership value $j = \operatorname{argmax}_{j=1,\dots,m}(u_{ij})$.

The introduction of the membership function u_{ij} and weight function w_i in the KHM algorithm alleviates the initialisation problem of the original KM algorithm. Another popular method for solving the initialisation problem is the *Fuzzy C-Means* (FCM) algorithm. In this case, the objective function to be optimised is:

$$Q(\underline{c}) = \sum_{i=1}^n \sum_{j=1}^m u_{ij}^r d_{ij}^2 \quad (8)$$

where u_{ij} is again the membership parameter and $r \geq 1$, which for large values makes the algorithm “more fuzzy” [6], [1]. The membership parameter u_{ij} attributes points to different clusters, in order to reduce to the effect of initialisation. However, Hamerly and Elkan demonstrated that the KHM can outperform the FCM in several experiments [6], therefore the KHM is chosen for our system.

III. THE KERNEL-BASED K-HARMONIC MEANS ALGORITHM

The KHM algorithm employs the Euclidean distance measure, as a classification criterion. However, this distance measure implicitly imposes the assumption of hyper-spherical clusters for the data, therefore, if the separation boundaries between clusters are nonlinear, the KHM will fail to identify these clusters correctly [5]. In addition, the Euclidean distance may be sensitive to noise and outliers, according to Huber [7]. To increase the robustness to these factors, another approach is to map the data into a high-dimensional nonlinear feature space and perform the clustering within this space [5].

Assume a scalar input set $X = \{x_1, \dots, x_i, \dots, x_n\}$ and a nonlinear mapping function $\kappa(\cdot)$, which maps x_i from the input space $X \subseteq \mathcal{R}^d$ to a new space $F \subseteq \mathcal{R}^D$ with higher or even infinite dimensions ($d \ll D$). The kernel function can be easily computed using Mercer kernels [11], [5], as an inner product (dot product) in the new space F , i.e. $K(x, y) = \kappa(x)^T \kappa(y)$, $\forall x, y \in X$. We introduce the nonlinear mapping $\kappa(\cdot)$ in the KHM cost function of (1), applying to all x_i and cluster centers c_j . The new objective function is given below.

$$Q(\underline{c}) = \sum_{i=1}^n \frac{m}{\sum_{j=1}^m \frac{1}{\|\kappa(x_i) - \kappa(c_j)\|^2}} \quad (9)$$

where

$$\begin{aligned} \|\kappa(x_i) - \kappa(c_j)\|^2 &= \kappa(x_i)^T \kappa(x_i) - 2\kappa(x_i)^T \kappa(c_j) + \kappa(c_j)^T \kappa(c_j) \\ &= K(x_i, x_i) - 2K(x_i, c_j) + K(c_j, c_j) \end{aligned}$$

The denominator $\|\kappa(x_i) - \kappa(c_j)\|^2$ of (9) defines a class of kernel-induced non-Euclidean distance measures. Common kernels that satisfy the Mercer conditions are the *Radial Basis Function* (RBF) and the polynomial function, as shown below [11], [5], [1]:

$$K(x, y) = \exp(-\|x - y\|^2 / \sigma^2) \quad (10)$$

$$K(x, y) = ((x^T y) + \theta)^p \quad (11)$$

where σ, p, θ are constants that control the shape of the kernel. As in [1], we will consider the Gaussian RBF in our analysis, as they tend to be more robust to noise and outliers. A Gaussian-like kernel emphasises small differences and suppresses large ones and hence possible outliers. It can effectively serve as a filter from noise and outliers [7], compared to the original Euclidean distance. The amount of suppression is controlled by the standard deviation σ of the kernel. The value of σ is usually empirically defined in the experimental section [1]. Since for the Gaussian kernel $K(x, x) = 1, \forall x \in X$, the objective function in (9) can be simplified to the following form:

$$Q(\underline{c}) = \sum_{i=1}^n \frac{2m}{\sum_{j=1}^m \frac{1}{1 - K(x_i, c_j)}} \quad (12)$$

Setting the partial derivative of $Q(\underline{c})$ with respect to c_j equal to zero, gives us the updating rules for centres (15).

$$\frac{\partial Q(\underline{c})}{\partial c_j} = 0, \forall j = 1, \dots, m \Rightarrow \quad (13)$$

$$\frac{\partial Q(\underline{c})}{\partial c_j} = \sum_{i=1}^n \frac{\frac{-4m}{\sigma^2} K(x_i, c_j)(x_i - c_j)}{\left(\sum_{j=1}^m \frac{1}{1 - K(x_i, c_j)}\right)^2 (1 - K(x_i, c_j))^2} = 0 \quad (14)$$

$$c_j^+ \leftarrow \frac{\sum_{i=1}^n \frac{K(x_i, c_j) x_i}{(1 - K(x_i, c_j))^2 \left(\sum_{j=1}^m \frac{1}{(1 - K(x_i, c_j))}\right)^2}}{\sum_{i=1}^n \frac{K(x_i, c_j)}{(1 - K(x_i, c_j))^2 \left(\sum_{j=1}^m \frac{1}{(1 - K(x_i, c_j))}\right)^2}} \quad (15)$$

The full derivation can be found in Appendix B. The above centre update function can be rewritten in terms of the membership and the weight functions. In a similar fashion to the analysis for the original KM algorithm, we can define these functions as in (16) and (17).

$$u_{ij}^+ \leftarrow \frac{K(x_i, c_j)}{(1 - K(x_i, c_j))^2 \sum_{j=1}^m \frac{K(x_i, c_j)}{(1 - K(x_i, c_j))^2}} \quad (16)$$

$$w_i^+ \leftarrow \frac{\sum_{j=1}^m \frac{K(x_i, c_j)}{(1 - K(x_i, c_j))^2}}{\left(\sum_{j=1}^m \frac{1}{1 - K(x_i, c_j)} \right)^2} \quad (17)$$

The properties, discussed in the previous section, for the new membership and weight functions should hold. Thus, the estimation of the centre position is performed, based on (7). Again, there might be cases, where the term $1 - K(x_i, c_j) \rightarrow 0$ and might cause instability. To avoid this problem, we replace the original expression with $\max(1 - K(x_i, c_j), \epsilon)$.

IV. THE SPATIAL KERNEL-BASED KHM USING A MODIFIED MEMBERSHIP FUNCTION

The proposed KKH algorithm utilises dissimilarity measurements of independent intensity information. However, the image intensity alone can not describe the features of an image in full, as it neglects the spatial correlation between the pixels in a neighbourhood and may be inadequate in classifying the image pixels into possible clusters. Considering that adjacent pixels are likely to have similar features, extra knowledge, such as the spatial relationship between pixels, can be introduced to the classification. The inclusion of spatial information in the clustering procedure in order to improve the performance of image segmentation has been proposed before [2], [10], [1], [14]. In these approaches, the spatial information is included as a penalty term in the clustering objective function, in order to discourage clusterings that attribute largely different cluster membership values to neighbouring pixels. Unlike the methods proposed in [2], [10], [1], the idea here is to construct a stochastic process that models the neighbourhood influence, using the concept of *Markov Random Fields* (MRF) [2], [4], [20]. The proposed approach will be referred to as the *Spatial Kernel-Based KHM* (SKKHM).

The Markovian property states that the probability of the pixel x_i belonging to a cluster j (i.e. the membership function u_{ij}), partially relies on the assignment of its neighbourhood W_i . Thus, we can use the Gibbs Distribution [2] to quantify the influence F_{ij} of the spatial neighbourhood W_i to pixel x_i , using the following expression.

$$F_{ij} = \frac{1}{L} \exp \left(-\beta \sum_{k \in W_i} (1 - u_{kj}) \right) \quad (18)$$

where W_i denotes a $Q \times Q$ window neighbourhood around the pixel x_i , excluding itself. The parameter β controls the magnitude of influence (when $\beta = 0$ the influence of neighbourhood is vanishing) and L is a normalisation constant. The influence F_{ij} aims at grouping neighbouring pixels to the same cluster. If the majority of the neighbouring pixels to pixel x_i belong to cluster j , then their membership functions u_{ij} will be much above 0.5. As a result, the term $\sum_{k \in W_i} (1 - u_{kj})$ will be small and F_{ij} should be closer to $1/L$. This should force the pixel x_i to be attributed to cluster j . In contrast, if the majority of the neighbouring pixels does not belong to cluster j , then their membership functions u_{ij} should be small. Consequently, the term $\sum_{k \in W_i} (1 - u_{kj})$ will be closer to 1 and the influence F_{ij} will be small.

Following an empirical Bayesian interpretation of the membership function, we introduce an influence factor F_{ij} to (16), in a similar fashion to [17]. As the membership function can be interpreted as the probability of the pixel x_i belong to cluster c_j , this probability should also depend on the ‘‘probability’’ F_{ij} of the neighbourhood W_i belonging to cluster j . Thus, the modified membership function u_{ij}^* can be expressed by the following concept:

$$u_{ij}^* \approx u_{ij} F_{ij} \quad (19)$$

where u_{ij} refers to (16). However, as the property $0 \leq u_{ij}^* \leq 1$ should always hold, the proposed membership function should be normalised to 1. Therefore, the following update rule for the membership function is proposed:

$$u_{ij}^{*+} \leftarrow \frac{K(x_i, c_j) F_{ij}}{(1 - K(x_i, c_j))^2 \sum_{j=1}^m \frac{K(x_i, c_j)}{(1 - K(x_i, c_j))^2} F_{ij}} \quad (20)$$

Effectively, according to (18), (20), given a higher probability of neighbouring pixels belonging to cluster j , pixel x_i is more likely to be assigned to the same cluster j , due to a larger influence factor F_{ij} . On the contrary, if the probability of pixels of the neighbourhood around x_i belonging to cluster j is small, pixel x_i is less likely to be assigned to cluster j as well. Therefore, in the case that pixels are disturbed by impulsive noise, the introduction of neighbourhood information will increase the tolerance to noise and hence produce more robust classification results. The size of the neighbourhood Q can change the segmentation results and will be investigated in the experimental section.

Driven by the proposed membership function u_{ij}^* , we introduce a new cost function $Q^*(c)$ in order to perform clustering:

$$Q^*(c) = \sum_{i=1}^n \frac{2m}{\sum_{j=1}^m \frac{F_{ij}}{1 - K(x_i, c_j)}} \quad (21)$$

Optimising (21) in a similar fashion to the previous sections (the full derivation is included in Appendix

C), the new update rule for w_i^* is given below:

$$w_i^{*+} \leftarrow \frac{\sum_{j=1}^m \frac{K(x_i, c_j) F_{ij}}{(1-K(x_i, c_j))^2}}{\left(\sum_{j=1}^m \frac{F_{ij}}{1-K(x_i, c_j)} \right)^2} \quad (22)$$

In summary, the proposed SKKHM method relies on optimising the harmonic-means of Kernel-based distances, weighted by membership functions and spatial information using MRF, in order to segment an image into areas of similar content.

V. AN EXTENSION TO MULTI-SPECTRAL IMAGES

Multi-spectral image processing techniques have been recently developed and applied in various fields, including colour images, magnetic resonance images, synthetic aperture radar images or fusion of multispectral images [12], [9]. Unlike 2-D scalar signals, multi-spectral images include extra information, such as additional color channels, infrared and ultraviolet channels. A natural solution to the multi-spectral image thresholding is to segment each channel independently and combine the results. However, this methodology may not be optimal, as we seem to ignore all the correlated information between components. As a consequence, we propose to extend the proposed SKKHM algorithm to solve a T -dimensional clustering problem, where T denotes the number of available channels. We will refer to this multi-dimensional extension of the original SKKHM, as the *generalized SKKHM* algorithm.

Assume that we have T -channel multi-spectral images $I(k, l, y)$, where $y = 1, \dots, T$. We form the following data set $\mathbf{X} = \{\underline{x}_1, \dots, \underline{x}_i, \dots, \underline{x}_n\}$, where n represents the number of pixels of each single channel. Each data ‘point’ \underline{x}_i can be interpreted as a T -dimensional vector, containing the value of the same pixel at each channel, i.e. $\underline{x}_i = I(k, l, :)$. The clustering centers can now be arranged in a $m \times T$ matrix $\mathbf{C} = [\underline{c}_1, \dots, \underline{c}_j, \dots, \underline{c}_m]^T$ and m again denotes the number of cluster centers. Following the same concept, as in the one-dimensional case, a new cost function of the generalised SKKHM clustering is expressed as follows:

$$Q^*(\mathbf{C}) = \sum_{i=1}^n \frac{2m}{\sum_{j=1}^m \frac{F_{ij}}{1-K(\underline{x}_i, \underline{c}_j)}} \quad (23)$$

The Gaussian multi-dimensional RBF can be a valid candidate [5], [11]. The aim is to simplify the computation of multi-dimensional non-linear mapping $K(\underline{x}, \underline{c})$. According to [11], every linear algorithm that employs inner products can be extended to a non-linear version through kernels that follow Mercer’s conditions. The T -dimensional Gaussian RBF kernel satisfies the Mercer’s conditions and is given by:

$$K(\underline{x}_i, \underline{c}_j) = \exp\left(-\|\underline{x}_i - \underline{c}_j\|^2 / \sigma^2\right) \quad (24)$$

Following a similar derivation to section IV and Appendix C, the update rules for the generalised SKKHM are given by the expressions below:

$$u_{ij}^{*+} \leftarrow \frac{K(\underline{x}_i, \underline{c}_j) F_{ij}}{(1 - K(\underline{x}_i, \underline{c}_j))^2 \sum_{j=1}^m \frac{K(\underline{x}_i, \underline{c}_j) F_{ij}}{(1 - K(\underline{x}_i, \underline{c}_j))^2}} \quad (25)$$

$$w_i^{*+} \leftarrow \frac{\sum_{j=1}^m \frac{K(\underline{x}_i, \underline{c}_j) F_{ij}}{(1 - K(\underline{x}_i, \underline{c}_j))^2}}{\left(\sum_{j=1}^m \frac{F_{ij}}{1 - K(\underline{x}_i, \underline{c}_j)} \right)^2} \quad (26)$$

$$\underline{c}_j^{*+} \leftarrow \frac{\sum_{i=1}^n u_{ij}^* w_i^* \underline{x}_i}{\sum_{i=1}^n u_{ij}^* w_i^*} \quad (27)$$

As a summary, given n observations of T -dimensional vectors, representing a multi-spectral image, the recursive clustering process for the generalised SKKHM is described as follows:

- 1) Randomly initialise the membership w_i^* and weight u_{ij}^* functions, $i = 1 \dots n$ and $j = 1 \dots m$
- 2) Calculate each T -dimensional cluster centres based on (27), and then update the membership w_i^* and weight u_{ij}^* functions according to (25), (26), respectively.
- 3) Re-estimate the cost function $Q(\mathbf{C})$, as in (23). Repeat steps 2 and 3, until the cost function convergences.
- 4) Finally, the image can be segmented by assigning each T -dimensional pixel \underline{x}_i to cluster j with the maximum membership value $j = \operatorname{argmax}_{j=1 \dots m} (u_{ij}^*)$.

VI. EXPERIMENTAL STUDY AND ANALYSIS

It is not our intention to compare the proposed SKKHM approach with the numerous image segmentation methods that exist in the literature. However, we will include some experiments to quantify the performance of the SKKHM algorithm in comparison to other common clustering segmentation approaches that employ kernel methods and spatial information.

A. Evaluation of SKKHM alone

In this section, we will examine the performance of each additional step in the gradual development of the SKKHM. In addition, the effect of several parameters on the algorithm's performance will be investigated.

1) *General Overview:* In the first experiment, the standard “coins” image of the MATLAB v.6.5 Image Processing Toolbox is used. The cases of no additive noise, additive Gaussian noise of zero mean and variance 0.001 and “Salt & Pepper” noise of density 0.1 are explored. All input images are scaled to the range $[0, 1]$. Figure 1 depicts the segmentation results of the standard K-Means (KM), the K-Harmonic Means (KHM), the Kernel K-Harmonic Means (KKHM) and the Spatial Kernel K-Harmonic Means (SKKHM) algorithms for these three cases. The values of $\sigma = 2$, $\beta = 1.2$, $L = 1$ and $Q = 3$ were used in the experiments. We will also assume that the number of clusters m is known. In this case, we set $m = 2$. To quantify the segmentation results, a “ground-truth” segmentation map was created using hand-labeling and was compared with the segmentation maps created by the algorithms. The ratio of correctly clustered pixels over the total number of image pixels is also depicted in Figure 1 and the term *Correct Clustering ratio* will be used for the rest of the paper.

The KM shows its limitations for effective segmentation especially in the presence of noise. In addition, the segmentation output is greatly dependent on the initialisation. The results depicted in Figure 1 are an average example of typical results using the KM. This problem of initialisation is solved in all algorithms that employ the KHM. The KHM seems to perform favourably in the case of no additive noise, however, shows its limitations in the case of noise. The introduction of a kernel-based measure seems to impose no improvement in the case of no additive noise. However, the segmentation results are improved in the case of Gaussian noise and not very much improved in the case of Salt & pepper noise. Finally, the proposed SKKHM seems to produce the most accurate segmentation map of all previous versions. The two types of additive noise are almost completely filtered and the SKKHM gives similar results with the noiseless image.

2) *Investigation of several parameters:* The importance of several parameters of the SKKHM algorithm in its performance is examined in this section. The standard “coins” image is employed again, together with the two noisy versions that were used in the previous example.

a) *Choice of σ in the Radial Basis Function kernel:* The choice of the parameter σ in the Radial Basis Function (RBF) kernel (Eq. (10)) is examined here. As mentioned earlier, σ controls the shape of the kernel and the suppression to outliers. We calculate the Correct Clustering ratio for each of the three test images for a range of values $0.01 \leq \sigma \leq 1$ using the Kernel-based KHM algorithm. The results are depicted in Figure 2. The graphs indicate that any choice of $\sigma > 0.3$ will guarantee optimal performance of the algorithm. Therefore, in the following experiments, we set $\sigma = 2$.

b) *Choice of β in the Markov Random Field function:* Here, we investigate the effect of the parameter β in the definition of the Markov Random Field (MRF) coefficient (Eq. 18). The parameter β controls

the effects of the MRF in the clustering cost function of the SKKHM algorithm. We used again the same three input images, set the rest of the SKKHM parameter's to $\sigma = 2$, $Q = 3$ and measured the algorithm's performance for $0.1 \leq \beta \leq 2.5$. The results are shown in Figure 3. The choice of β has almost negligible influence in the algorithm's performance in the noiseless case, although we get the best score around a value $\beta = 1.2$. The choice of β seems to be more important in the case of noisy input, especially in the case of "Salt & Pepper" noise. We can see in figure 3 that an optimum in performance is reached for β around 1.2, which is the typical value that will be used in the following experiments.

c) Choice of neighbourhood size Q : The effect of the neighbourhood size Q in the estimation of the MRF component F_{ij} (Eq. 18) is investigated here. In a similar fashion, the performance of the SKKHM is investigated for $Q = 3, 5, \dots, 15$ for the three input images ($\sigma = 2$, $\beta = 1.2$). The results are illustrated in Figure 4. In all cases, it seems that for $Q > 3$, the algorithm seems to give worse performance than for $Q = 3$. Therefore, the optimal neighbourhood size is $Q = 3$, which is going to be used in the following experiments.

B. Comparison of SKKHM to other Spatial Clustering approaches

Once the basic properties of the SKKHM are explored, in this section, the proposed algorithm is compared with similar type of algorithms, based on the Fuzzy C-Means cost function. More specifically, we implemented the two variants of a Spatial Kernel-based Fuzzy C-Means (SKFCM) approach that was introduced in [1]. In these approaches, the Fuzzy C-Means cost function is used, instead of the Harmonic K-Means that is used in our algorithm. The problem of noise and outliers is addressed using the Radial Basis Function kernel. The spatial information is introduced as an extra constraint in the FCM cost function. The two variants, proposed in [1], are two different practical implementations of the imposed spatial constraints. The main difference between this approach and the proposed approach using Markov Random Fields is that, in the SKFCM case, the spatial information is extracted from the kernel-based distance between the pixels in the local neighbourhood, whereas in the SKKHM case the spatial information is extracted only from the membership function u_{ij} .

In figure 5, the "coins" image is used again from the previous experiment to compare the performance of SKKHM to the two variants SKFCM₁ and SKFCM₂. Two types of noise are used to contaminate the image, as previously, with the same amount of noise. The parameter values used for SKKHM were $\sigma = 2$, $\beta = 1.2$, $L = 1$ and $Q = 3$ and for the SKFCM, we used $\sigma = 2$, $\alpha = 1.5$ and $Q = 3$, as proposed in [1]. The results are depicted in figure 5, along with the corresponding Correct Clustering Ratio using a hand-labeled segmentation map. In general, we can see that the three algorithms perform similarly in

the case of no noise or Gaussian noise. However, the visual evaluation shows that the SKKHM offers a more connected and coherent segmentation map than the SKFCMs. The major difference lies in the case of “Salt & Pepper” noise, where the SKKHM features similar performance as previously but the SKFCMs produce rather noisy segmentation results. This can be due to the different method of extracting spatial information. It seems that extracting spatial information from the membership function and forcing neighbouring pixels to have similar membership values is immune even to strong impulsive noise.

To verify the conclusions from the previous experiment, we repeated the same experiment with the standard “rice” image from the MATLAB Image Processing Toolbox. The results are depicted in figure 6. A similar trend can be observed here. The three schemes perform similarly in the case of no noise and Gaussian noise in terms of Correct Clustering Ratio. The SKKHM still yields improved visual clustering results compared to SKFCMs. Again, in the case of “Salt & Pepper” the SKKHM produces much improved visual and numerical performance, compared to SKFCM₁ and SKFCM₂.

To quantify the effect of noise level to the algorithms’ performance, we measured the three algorithm’s performance in the case of various levels of Gaussian and “Salt & Pepper” noise. The experiments were performed on the “rice” image and the Correct Clustering Ratio was measured in each case. The variance of Gaussian Noise was set to a range of values from 0 to 0.01 and the density of the “Salt & Pepper” noise to a range from 0 to 0.1. In figure 7 (left), the performance of the three algorithms in the case of Gaussian Noise (GN) is depicted and in figure 7 (right) the case of “Salt & Pepper” (SPN) noise. In the GN case, we can spot a similar decrease in performance with the increasing noise level, however, the rate of change is similar for all approaches. In figure 7 (right), we can observe that the performance of the two SKFCM algorithms drops significantly compared to the SKKHM in the case of large SPN.

Finally, in order to test the performance of the three algorithm in the case of a multi-dimensional setup and more than 2 clusters, we performed segmentation of the image in figure 8. The image was acquired by a Sony DSC-P10 camera using three objects of different colour. The objective is to use the RGB color intensity information, in order to segment the image in four different regions (clusters) $m = 4$: the background, the mug, the vase and the salt box. We used the generalised SKKHM, as described previously, and the generalised version of SKFCM₁ and SKFCM₂. We also performed the same experiment, in the case of “Salt & Pepper” noise of density 0.05. We used the same values for the parameters for all algorithms as before. The full set of results is depicted in figure 8. In this case, the SKKHM seemed to produce much more coherent segmentation maps in both cases. The two SKFCM algorithms performed reasonably well in the noiseless case, however, they made serious segmentation mistakes in the noisy case. Two objects were attributed to the same cluster, which implies that impulsive

noise can mislead the clustering operator rather seriously.

VII. CONCLUSION

In this paper, we have presented a Spatial Kernel-based K-Harmonic Means clustering technique, incorporating spatial information from the images. Inheriting the immunity of K-Harmonic means clustering to initialisation, the new algorithm is less prone to over-segmentation, outliers and noise, compared to other traditional clustering methods: K-Means (KM), the standard K-Harmonic Means (KHM) and the standard Fuzzy C-Means. A Gaussian kernel induces less sensitivity to outliers and noise. Incorporating neighbouring spatial information in the clustering cost function can yield more accurate membership and weighting functions. The proposed scheme yields encouraging results in the area of image segmentation, where spatial information is important. Comparison with similar Kernel-based Fuzzy C-Means algorithms with spatial constraints showed that the proposed approach can produce more coherent segmentation maps and in addition is less prone to impulsive noise, such as ‘‘Salt & Pepper’’ noise.

ACKNOWLEDGMENTS

This work is supported by the Data Information Fusion project of the Defence Technology Centre, UK. The authors would like to thank the anonymous reviewers for their insightful comments and suggestions that helped to improve the quality of the paper.

APPENDIX

A. Derivation of the K-Harmonic Means algorithm

The derivative $\partial Q(\underline{c})/\partial c_j$ is calculated as follows:

$$\frac{\partial Q(\underline{c})}{\partial c_j} = \sum_{i=1}^n \frac{\partial}{\partial c_j} \frac{m}{\sum_{j=1}^m (x_i - c_j)^{-2}} = \sum_{i=1}^n \frac{-2m(x_i - c_j)^{-3}}{\left(\sum_{j=1}^m (x_i - c_j)^{-2}\right)^2} = \sum_{i=1}^n \frac{-2m(x_i - c_j)}{\left(\sum_{j=1}^m \frac{1}{d_{ij}}\right)^2 d_{ij}^2} \quad (28)$$

The update for the centre function c_j^+ is calculated as follows:

$$\frac{\partial Q(\underline{c})}{\partial c_j} = 0 \Rightarrow \sum_{i=1}^n \frac{-2m(x_i - c_j)}{\left(\sum_{j=1}^m \frac{1}{d_{ij}}\right)^2 d_{ij}^2} = 0 \Rightarrow \sum_{i=1}^n \frac{x_i}{\left(\sum_{j=1}^m \frac{1}{d_{ij}}\right)^2 d_{ij}^2} = c_j \sum_{i=1}^n \frac{1}{\left(\sum_{j=1}^m \frac{1}{d_{ij}}\right)^2 d_{ij}^2} \Rightarrow \quad (29)$$

$$c_j^+ \leftarrow \frac{\sum_{i=1}^n \frac{x_i}{d_{ij}^2 \left(\sum_{j=1}^m \frac{1}{d_{ij}}\right)^2}}{\sum_{i=1}^n \frac{1}{d_{ij}^2 \left(\sum_{j=1}^m \frac{1}{d_{ij}}\right)^2}} \quad (30)$$

B. Derivation of the Kernel-based K-Harmonic Means algorithm.

The derivative $\partial Q(\underline{c})/\partial c_j$ is calculated as follows:

$$\frac{\partial Q(\underline{c})}{\partial c_j} = \sum_{i=1}^n \frac{\partial}{\partial c_j} \frac{2m}{\sum_{j=1}^m (1 - K(x_i, c_j))^{-1}} \quad (31)$$

$$= \sum_{i=1}^n \frac{-2m}{\left(\sum_{j=1}^m (1 - K(x_i, c_j))^{-1}\right)^2} (1 - K(x_i, c_j))^{-2} K(x_i, c_j) \frac{2(x_i - c_j)}{\sigma^2} \quad (32)$$

$$= \sum_{i=1}^n \frac{\frac{-4m}{\sigma^2} K(x_i, c_j)(x_i - c_j)}{\left(\sum_{j=1}^m (1 - K(x_i, c_j))^{-1}\right)^2 (1 - K(x_i, c_j))^2} \quad (33)$$

$$(34)$$

The update for the centre function c_j^+ is calculated as follows:

$$\frac{\partial Q(\underline{c})}{\partial c_j} = 0 \Rightarrow \quad (35)$$

$$\sum_{i=1}^n \frac{K(x_i, c_j)x_i}{\left(\sum_{j=1}^m (1 - K(x_i, c_j))^{-1}\right)^2 (1 - K(x_i, c_j))^2} = c_j \sum_{i=1}^n \frac{K(x_i, c_j)}{\left(\sum_{j=1}^m (1 - K(x_i, c_j))^{-1}\right)^2 (1 - K(x_i, c_j))^2} \Rightarrow \quad (36)$$

$$c_j^+ \leftarrow \frac{\sum_{i=1}^n \frac{K(x_i, c_j)x_i}{(1 - K(x_i, c_j))^2 \left(\sum_{j=1}^m \frac{1}{(1 - K(x_i, c_j))}\right)^2}}{\sum_{i=1}^n \frac{K(x_i, c_j)}{(1 - K(x_i, c_j))^2 \left(\sum_{j=1}^m \frac{1}{(1 - K(x_i, c_j))}\right)^2}} \quad (37)$$

C. Derivation of the Spatial Kernel-based K-Harmonic Means algorithm.

The derivative $\partial Q^*(\underline{c})/\partial c_j$ for the SKKHM is calculated as follows:

$$\frac{\partial Q^*(\underline{c})}{\partial c_j} = \sum_{i=1}^n \frac{\partial}{\partial c_j} \frac{2m}{\sum_{j=1}^m F_{ij}(1 - K(x_i, c_j))^{-1}} \quad (38)$$

$$= \sum_{i=1}^n \frac{-2mF_{ij}}{\left(\sum_{j=1}^m F_{ij}(1 - K(x_i, c_j))^{-1}\right)^2} (1 - K(x_i, c_j))^{-2} K(x_i, c_j) \frac{2(x_i - c_j)}{\sigma^2} \quad (39)$$

$$= \sum_{i=1}^n \frac{\frac{-4m}{\sigma^2} F_{ij}K(x_i, c_j)(x_i - c_j)}{\left(\sum_{j=1}^m F_{ij}(1 - K(x_i, c_j))^{-1}\right)^2 (1 - K(x_i, c_j))^2} \quad (40)$$

$$(41)$$

The update for the centre function c_j^+ of the SKKHM is calculated as follows:

$$\frac{\partial Q^*(\underline{c})}{\partial c_j} = 0 \Rightarrow \quad (42)$$

$$\sum_{i=1}^n \frac{F_{ij}K(x_i, c_j)x_i}{\left(\sum_{j=1}^m F_{ij}(1 - K(x_i, c_j))^{-1}\right)^2 (1 - K(x_i, c_j))^2} = c_j \sum_{i=1}^n \frac{F_{ij}K(x_i, c_j)}{\left(\sum_{j=1}^m F_{ij}(1 - K(x_i, c_j))^{-1}\right)^2 (1 - K(x_i, c_j))^2} \Rightarrow \quad (43)$$

$$c_j^+ \leftarrow \frac{\sum_{i=1}^n \frac{F_{ij}K(x_i, c_j)x_i}{(1 - K(x_i, c_j))^2 \left(\sum_{j=1}^m \frac{F_{ij}}{1 - K(x_i, c_j)}\right)^2}}{\sum_{i=1}^n \frac{F_{ij}K(x_i, c_j)}{(1 - K(x_i, c_j))^2 \left(\sum_{j=1}^m \frac{F_{ij}}{1 - K(x_i, c_j)}\right)^2}} \quad (44)$$

Following a similar methodology as in the Harmonic K-Means, one can derive the expressions for the membership and the weight function, as in (20) and (22).

REFERENCES

- [1] Chen S.C. and Zhang D.Q., "Robust Image Segmentation using FCM with spatial constraints based on new kernel-induced distance measure", IEEE Transactions on Systems, Man and Cybernetics, Part B - Cybernetics, Vol. 34, No. 4, pp. 1907-1916, Aug. 2004.
- [2] Derin H. and Elliot H., "Modelling and segmentation of noisy and textured images using Gibbs random field", IEEE Transactions on Patter Analysis and Machine Intelligence, vol. 9, pp. 39-55, 1987.
- [3] Dunn J.C., "A Fuzzy Relative of the ISODATA Process and Its Use in Detecting Compact Well-Separated Clusters", Journal of Cybernetics 3: 32-57.
- [4] Geman S. and Geman D., "Stochastic Relaxation, Gibbs Distributions, and the Bayesian Restoration of Images", IEEE Transactions on Patter Analysis and Machine Intelligence, vol. 6, pp. 721-41, 1984.
- [5] Girolami M., "Mercer kernel-based clustering in feature space", IEEE Transactions on Neural Networks, Vol. 13, No. 3, May 2002.
- [6] Hamerly G. and Elkan C., "Alternatives to the K-means algorithm that find better clusterings", Proceeding of CIKM, 2002.
- [7] Huber P.J., "Robust statistics", Wiley, New York, 1981.
- [8] Jain A.K., Murty M.N. and Flynn P.J., "Data clustering: a review", ACM Computing Surveys, 31(3):264(323), 1999.
- [9] Kerfoot I.B., Bresler Y., "Theoretical Analysis of Multispectral Image Segmentation Criteria", IEEE Transactions on Image Processing, vol. 8, No. 6, June 1999.
- [10] Luo J., Chen C.W., and Parker K.J., "On the application of Gibbs random field in image processing: from segmentation to enhancement", Journal of Electronics Imaging, vol. 4, pp. 187-198, 1995.
- [11] Muller K.R., Mika S., Ratsch G., Tsuda K. and Scholkopf B., "An Introduction to Kernel-Based Learning Algorithms", IEEE Trans. Neural Networks, vol. 12, pp. 181-202, March 2001/
- [12] Nattkemper T.W., "Multivariate Image Analysis in Biomedicine - A Methodological Review", Journal of Biomedical Informatics, 2004.
- [13] Pappas T.N., "An Adaptive Clustering Algorithm for Image Segmentation", IEEE Transactions on Signal Processing, vol. 40, pp. 901-914, 1992.
- [14] Pham D.L., "Fuzzy clustering with spatial constraints", IEEE Proc. Int. Conf. Image Processing, New York, Aug. 2002, pp. II-65-II-68.
- [15] Sonka M., Hlavac V., Boyle R., "Image processing, Analysis and Machine Vision", Second Edition, Brooks/Cole Publishing Company, 1999.

- [16] Tang K., Astola J. and Neuvo Y., "Nonlinear Multivariate Image Filtering Techniques", IEEE Transactions on Image Processing, vol. 4, no. 6, 1995.
- [17] Wiemker R., "An Iterative Spectral-Spatial Bayesian Labeling Approach For Unsupervised Robust Change Detection On Remotely Sensed Multispectral Imagery", in G. Sommer , K. Daniilidis and J. Pauli (eds.): Proceedings of the 7th International Conference on Computer Analysis of Images and Patterns, Springer LNCS 1296, ISBN 3-540-63460-6, pp. 263-270, CAIP'97, Kiel, September, 1997.
- [18] Zhang D. and Chen S., "Kernel-based fuzzy and possibilistic C-Means clustering", Proceedings of the International Conference on Artificial Neural Networks (ICANN03), Istanbul, Turkey, 2003, pp. 122-125.
- [19] Zhang D. and Chen S., "A Novel Kernelized Fuzzy C-means Algorithm with Application in Medical Image Segmentation", Artificial Intelligence in Medicine, no. 32, vol. 1, pp. 37-50, 2004.
- [20] Zhang J., "The mean field theory in EM procedures for Markov Random Fields", IEEE Transactions on Signal Processing, vol. 40, no. 10, pp. 2570-2583, 1992.
- [21] Zhang B., Hsu M. and Dayal U., "K-Harmonic Means: A Spatial clustering algorithm with boosting", Source Lecture Notes In Computer Science; Proceedings of the First International Workshop on Temporal, Spatial, and Spatio-Temporal Data Mining-Revised Papers table of contents, vol. 2007, pp. 31-45, 2000.

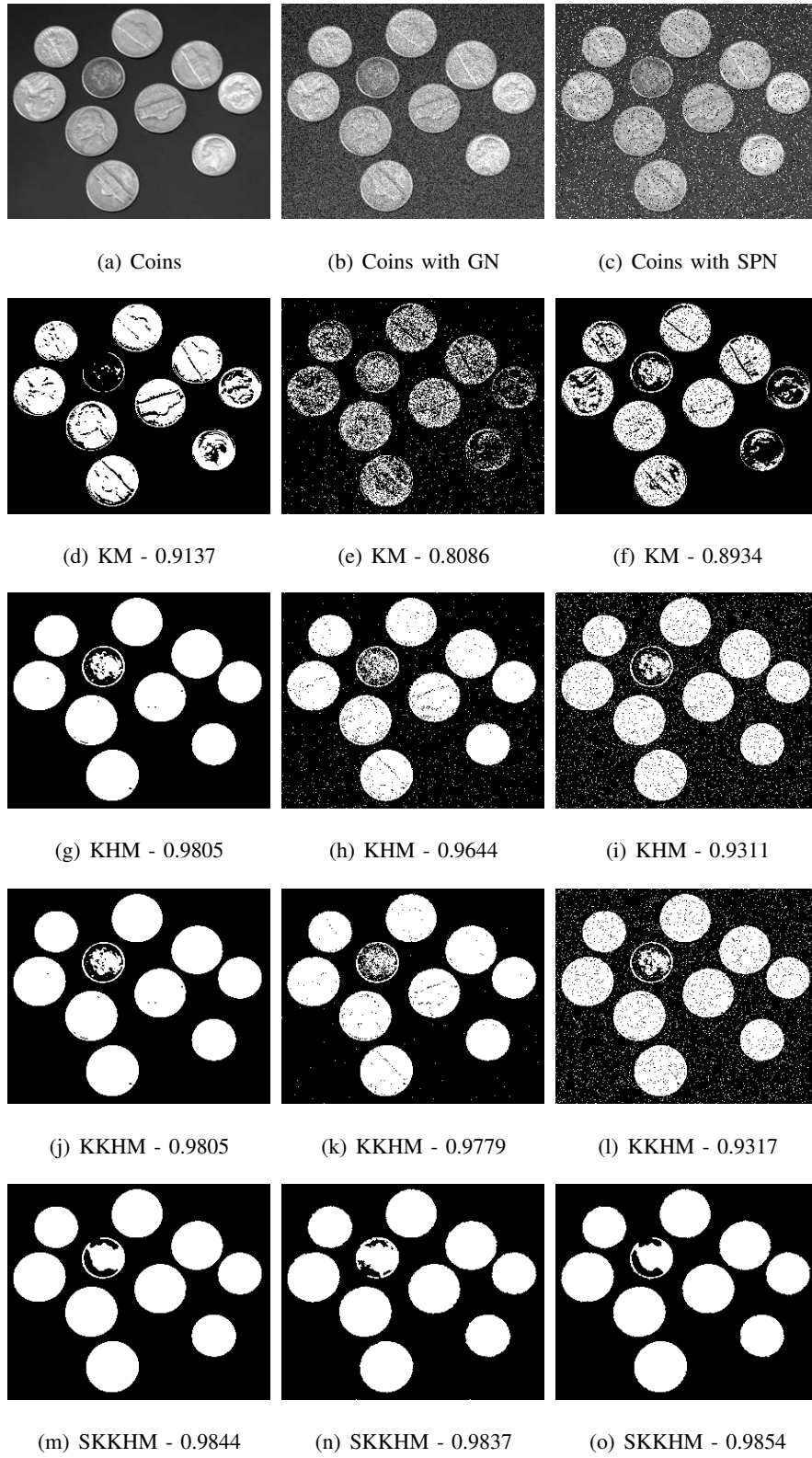


Fig. 1. Comparison of segmentation results of the K-Means (KM), K-Harmonic Means (KHM), Kernel K-Harmonic Means (KKHM) and Spatial Kernel K-Harmonic Means (SKKHM) in the case of no additive, Gaussian (GN) and “Salt & Pepper” noise (SPN).

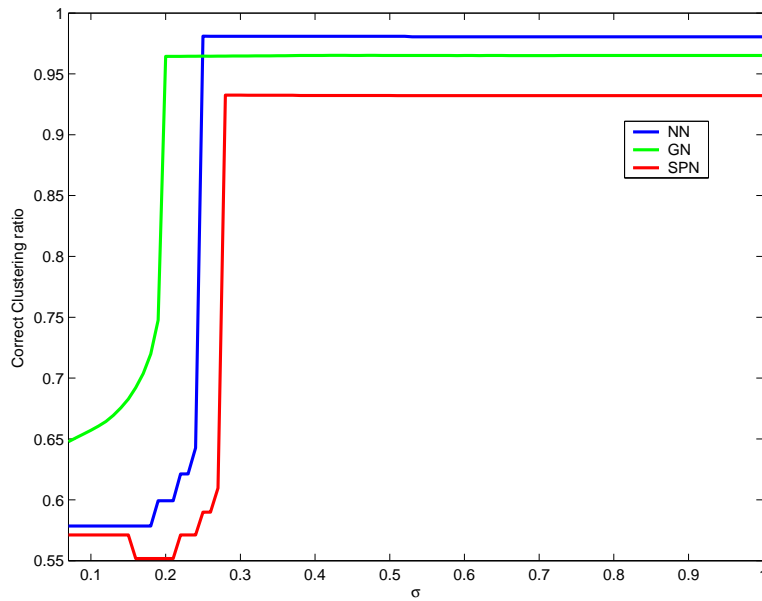


Fig. 2. Performance of KKHM for different values of σ in the case of no additive noise (NN), Gaussian noise (GN) and “salt & Pepper” noise (SPN).

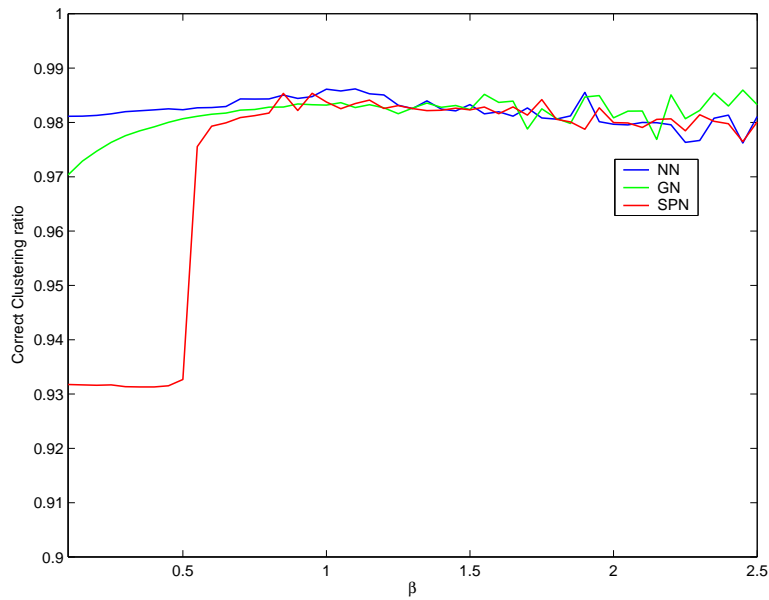


Fig. 3. Performance of SKKHM for different values of β ($Q = 3, \sigma = 2$) in the case of no additive noise (NN), Gaussian noise (GN) and “salt & Pepper” noise (SPN).

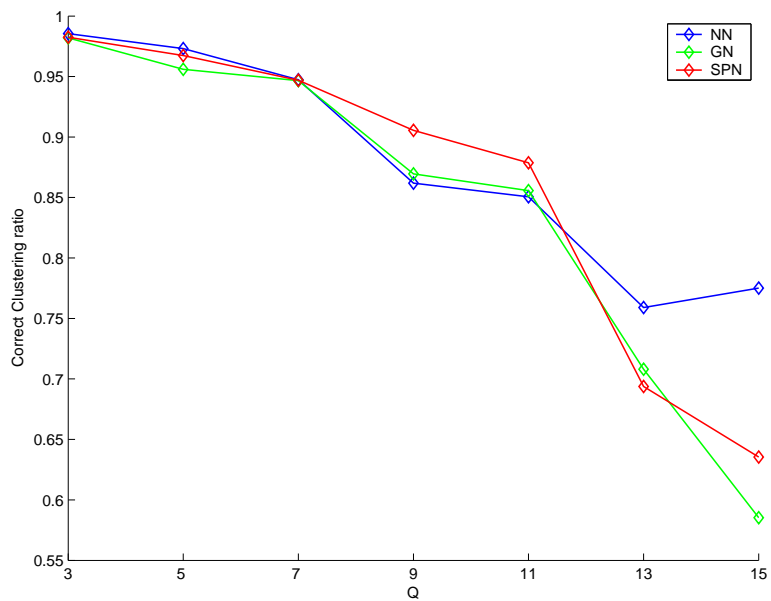


Fig. 4. Performance of SKKHM for different values of Q ($\beta = 1.2, \sigma = 2$) in the case of no additive noise (NN), Gaussian noise (GN) and “salt & Pepper” noise (SPN).

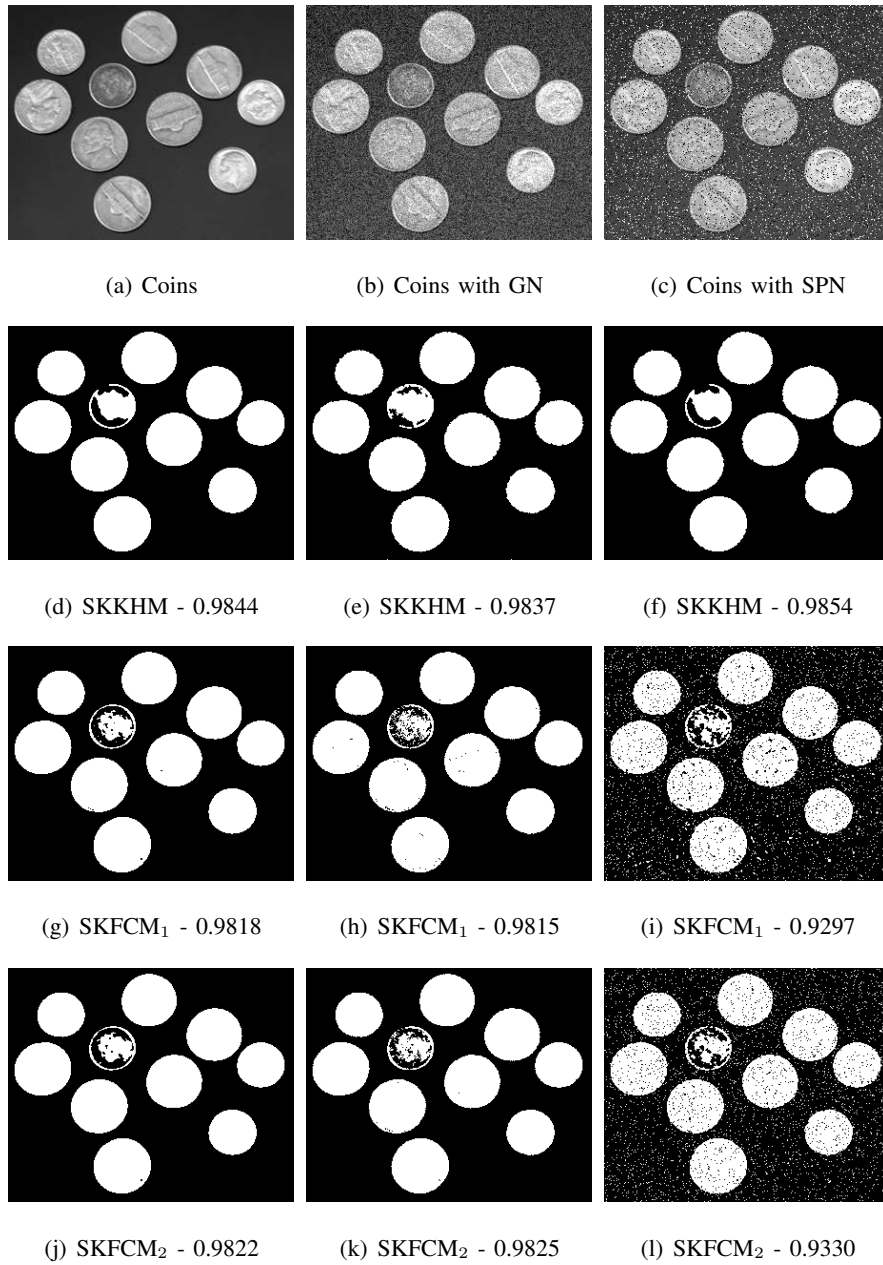


Fig. 5. Comparison of segmentation results of the Spatial Kernel K-Harmonic Means (SKKHM) with the two Spatial Kernel-based Fuzzy C-Means schemes [1] using the “coins” image in the case of no additive, Gaussian (GN) and “Salt & Pepper” noise (SPN).

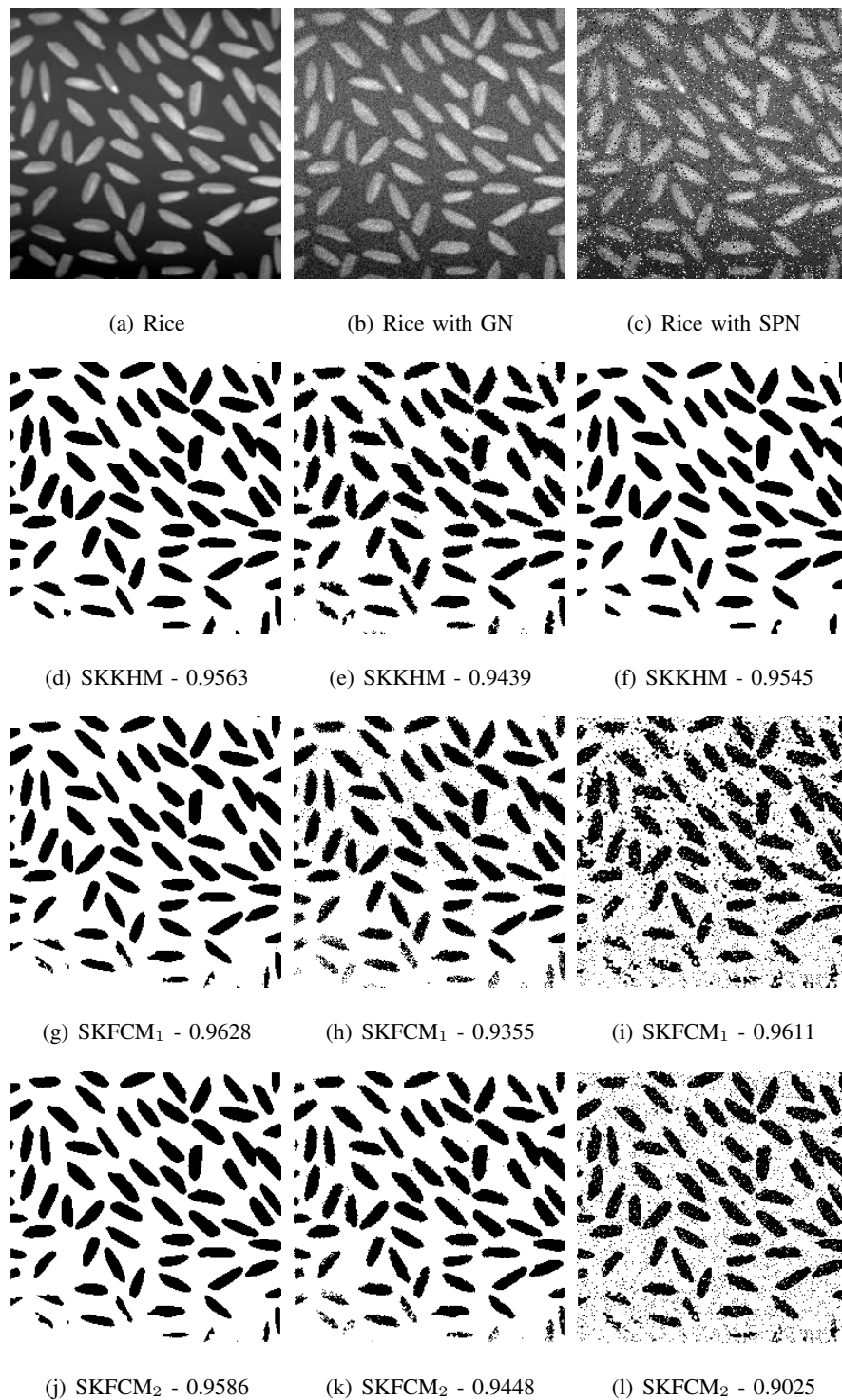


Fig. 6. Comparison of segmentation results of the Spatial Kernel K-Harmonic Means (SKKHM) with the two Spatial Kernel-based Fuzzy C-Means schemes [1] using the “rice” image in the case of no additive, Gaussian (GN) and “Salt & Pepper” noise (SPN).

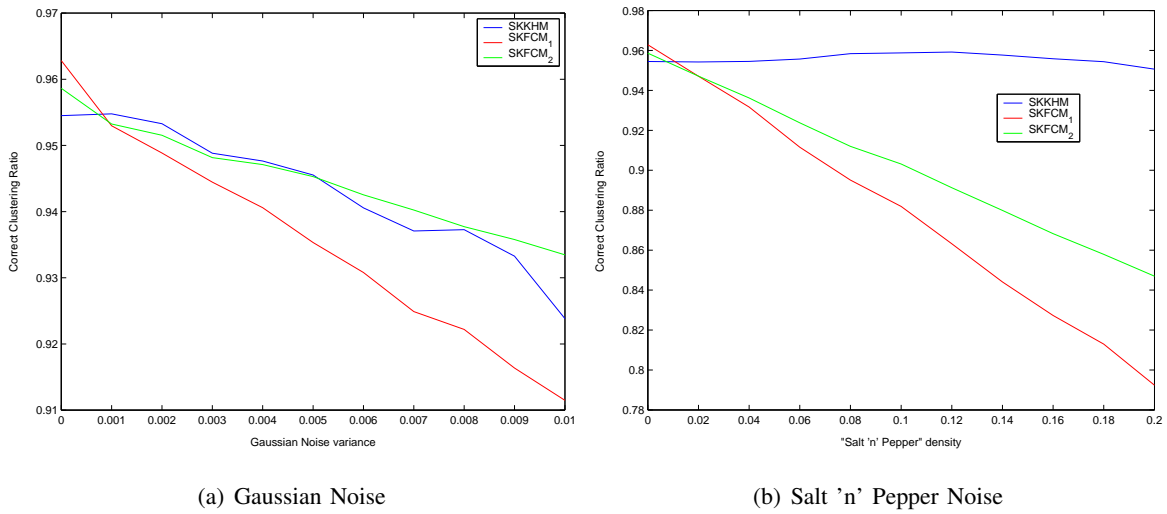


Fig. 7. Comparison of SKKHM with SKFCM₁ and SKFCM₂ using the “Rice” image with different levels of Gaussian (left) and “Salt & Pepper” noise (right).

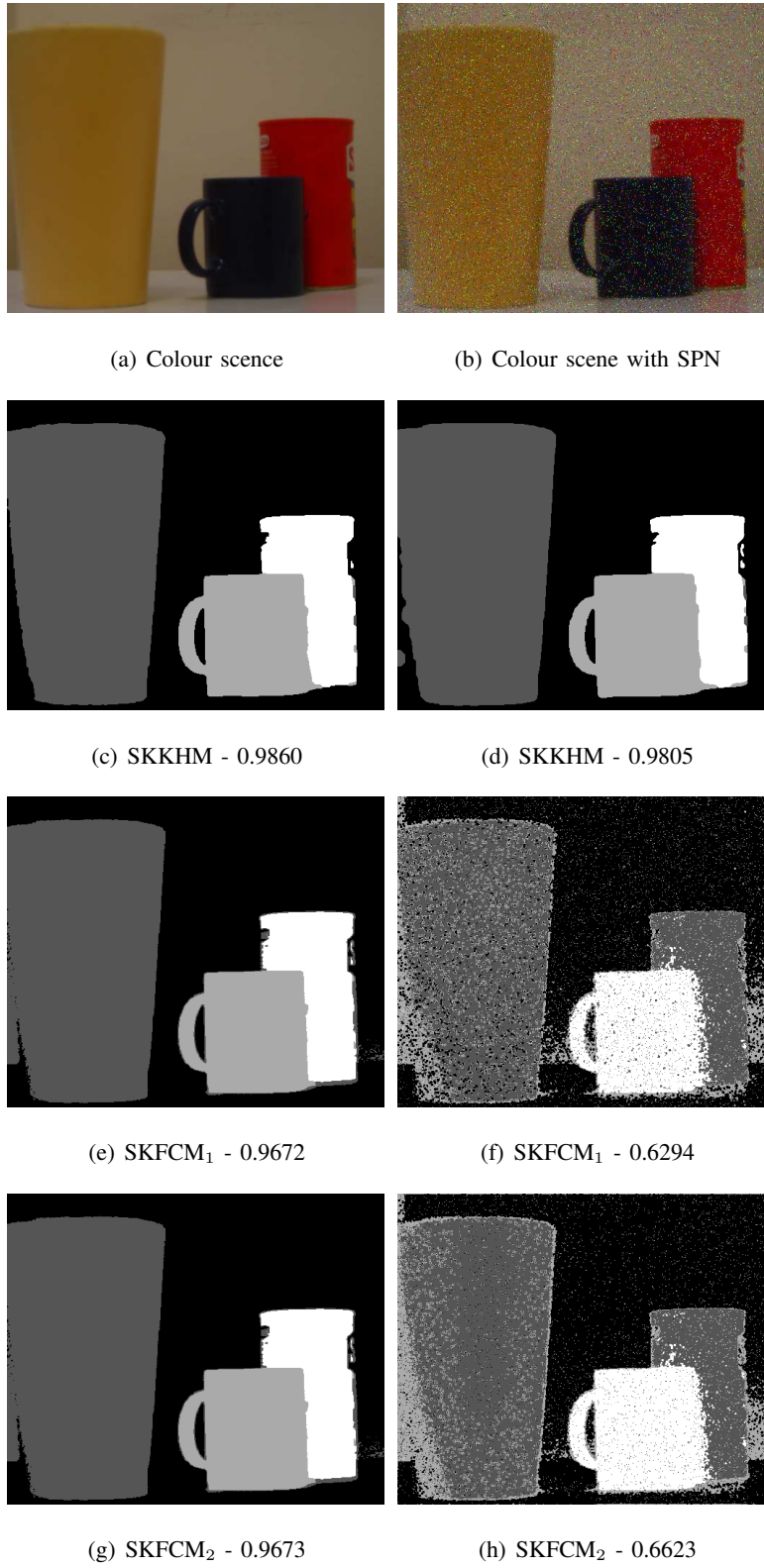


Fig. 8. Comparison of segmentation results of the Multi-dimensional Spatial Kernel K-Harmonic Means (SKKHM) with the two Multidimensional Spatial Kernel-based Fuzzy C-Means schemes [1] for a real RGB image in the case of no additive and “Salt & Pepper” noise (SPN).