INSTITUT DE STATISTIQUE
BIOSTATISTIQUE ET
SCIENCES ACTUARIELLES
(ISBA)

UNIVERSITÉ CATHOLIQUE DE LOUVAIN

DISCUSSION
PAPER

2012/15

SHAH: SHAPE-ADAPTIVE HAAR WAVELET TRANSFORM FOR IMAGES WITH APPLICATION TO CLASSIFICATION

TIMMERMANS , C. and P. FRYZLEWICZ

SHAH: SHape-Adaptive Haar wavelet transform for images with application to classification

Catherine Timmermans and Piotr Fryzlewicz

***Abstract*—We propose the SHAH (SHape-Adaptive Haar) transform for images, which results in an orthonormal, adaptive decomposition of the image into Haar-like components, arranged hierarchically according to decreasing importance, whose shapes reflect the features present in the image. The decomposition is as sparse as it can be for piecewise-constant images. It is performed via an iterative bottom-up algorithm with quadratic computational complexity; however, nearly-linear variants also exist. SHAH is rapidly invertible.**

We use SHAH to define the BAGIDIS semi-distance between images. It compares both the amplitudes and the locations of the SHAH components of the images and is flexible enough to account for feature misalignment. Performance of the SHAH+BAGIDIS methodology is illustrated in regression, classification and clustering problems and shown to be very encouraging.

A clear asset of the methodology is its very general scope: it can be used with any images or more generally with data that can be described as graphs or networks.

***Index Terms*—Adaptive transformations, greedy algorithms, multiscale, sparsity, dissimilarity, statistical learning.**

I. INTRODUCTION

IN this paper, we develop a new paradigm for investigating gray level images. Our main proposal is a new transformation for such images, termed the SHape-Adaptive Haar (SHAH) transform. We will often use the acronym SHAH to refer to the transform itself. The input to SHAH is an object called an Intensity Network, made up of a graph (whose nodes are the pixels and the edges define a topology accounting for neighborhood relationships between the nodes), a codebook (which encodes the location of the nodes) and a set of intensities associated with the nodes. With some abuse of terminology, our use of the term "image" below will often imply that there is an Intensity Network associated with that image.

The SHAH transform encodes the Intensity Network in a unique, invertible, and (often) sparse way. SHAH describes the image in a hierarchical (multiscale) fashion, as a ranked collection of weighted level differences between pairs of zones in the image, the most informative ones being ranked first. Thus, it provides a natural decomposition of the image as a set of features ordered according to their importance for the image description. It bypasses the classical 'dyadic' notion of wavelet scale.

C. Timmermans is with Institute of Statistics, Biostatistics and Actuarial Sciences, Universit catholique de Louvain, voie du Roman Pays 20, BE-1348 Louvain-la-Neuve, Belgium. e-mail: catherine.timmermans@uclouvain.be

Piotr Fryzlewicz is with Department of Statistics, London School of Economics, Columbia House, Houghton Street, London WC2A 2AE, UK. e-mail: p.fryzlewicz@lse.ac.uk

Two representations of the SHAH transform are described in the paper. Firstly, SHAH can be viewed as a series of recursive operations on the Intensity Network, which is how it is implemented in practice in order to minimise the computational effort. The operations start on the pixel-to-pixel level and then proceed to larger and larger regions of the image in a 'bottom-up' fashion. Secondly, SHAH is a projection of the image on a particular image-adapted multiscale orthonormal basis; hence the name 'shape-adaptive'. This interpretation of SHAH helps understand its hierarchical nature and potential for obtaining sparse representation of images. From this perspective, the transform may be seen as a particular Haar-like transform, and our work might be seen as a two-dimensional extension of the one-dimensional Unbalanced Haar transform of [1]. We emphasise again that unlike the traditional Haar wavelet transform, the Haar-like basis in our construction is itself adapted to the features of the image.

The second contribution of this work is to propose a new dissimilarity measure for images, taking as its input the SHAH transforms of both images. Our SHAH-based dissimilarity measure uses a weighted norm to compare the SHAH signatures (i.e. outputs of the SHAH transform) of the two images, and can be shown to be a semi-distance. It may be interpreted as follows: either image gets hierarchically encoded in a different basis that is best suited to it. We then compare both the bases themselves and the projections of the images onto them. The resulting semi-distance is termed BAGIDIS (Bases Giving Distances), for compatibility with [2] where a related idea was suggested for comparing one-dimensional signals and which our approach extends.

The BAGIDIS semi-distance between images compares both the intensities and the locations of features in the image. Therefore, it may be able to capture the similarity of images that are misaligned, which is often desirable in practical applications. Depending on the problem at hand, the focus of BAGIDIS can be placed on intensity differences, location differences along one axis or the other, or any combination of those at different hierarchical levels in the description of the images.

As it can be used in association with any distance-based algorithm, BAGIDIS may be used in a wide range of statistical problems, such as prediction, classification, data visualization, testing, etc. For instance, it can be used in association with multidimensional scaling [3], functional nonparametric regression and discrimination models [4] or k-NN algorithms [5].

It is worth emphasizing that the methodology we propose applies to more general data structures than merely images, although we restrict ourselves to the latter in the present work.

Both the SHAH transform and the BAGIDIS semi-distance can be applied to any data that can be encoded as an Intensity Network, i.e. a graph whose nodes are associated with a given intensity and are rooted in a normed, not necessarily two-dimensional, space.

The paper is organised as follows. Section II defines the SHAH algorithm and describes some of its properties. Section III introduces the BAGIDIS semi-distance for comparing images and discusses how it can be used in association with distance-based algorithms. Unusually, we defer the literature review to Section IV; this is done so that we can locate our work within the vast existing image processing and graph theory literature with a more detailed picture of our methodology in mind. Section V investigates some simulated and real data examples, and Section VI concludes.

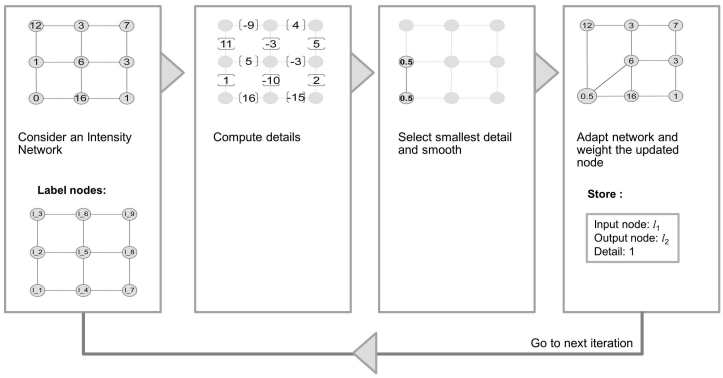

Fig. 1. A schematic illustration of the SHAH algorithm. Illustrations refer to the first iteration of the algorithm.

II. THE SHAPE-ADAPTIVE HAAR TRANSFORM FOR IMAGES

The SHape-Adaptive Haar (SHAH) transform for images is an algorithmic tool for encoding images in a sparse, data-driven, invertible, hierarchical way. The input to SHAH is an object we call an Intensity Network, constructed as follows. Consider a gray level image, stored as a real-valued matrix of dimension $N \times M$. Then, draw a *network* on this image. Each pixel is a *node* of this network, and each node is related by *edges* to its four nearest neighbours (in the left, right, top and bottom directions respectively). This graph structure mathematically encodes the idea of neighbourhood between the pixels of the image; more complex topologies are possible but we do not pursue it in this work. Associate an orientation to the edges so that each of them consists in an input node and an output node. Give a unique label to each node. Store the mapping relating those labels to the Cartesian coordinates of the pixels in a codebook. Moreover, associate a uniform weight to all the nodes of the network, as the information they store (i.e. the value of the related pixel) is *a priori* equally important in the image description. The object comprising the pixel values (the NM real values stored in an $N \times M$ matrix) and the graph structure (which consists of the NM nodes and $2NM-N-M$ edges) rooted in space through the codebook is termed an *Intensity Network*. An example of such an Intensity Network can be found in Figure 2.

Smoothing the image: The idea of the SHAH transform is to progressively smooth the image in a data-adaptive way, while retaining as much information as possible about the current image in each smoothing step. In practice, compute (weighted) differences between pairs of neighbour nodes along each edge. Those differences are called *details*. Identify the smallest detail (in absolute value) and replace the values of the corresponding linked nodes by their (weighted) average. Then, reduce those two nodes to a single node in the network, which is given a larger weight due to the increased number of pixels it encodes. Finally, update the graph structure of the network consequently, by removing the edge between the linked nodes. Since the detail being replaced is the smallest one, the loss of information is the smallest possible. This reduction process is iterated $NM-1$ times, up to the point where the image is finally reduced to a single node. Figure 1 illustrates the first iteration of this smoothing process. Figure 4 shows an example of how the graph structure might evolve along the reduction process.

Encoding the transform: At each iteration of the algorithm, store the labels of the nodes that are reduced as well as the (weighted) difference between them. Thus, each iteration returns three values: the input node label, the output node label and the selected detail, the latter being the (weighted) value at the output node minus the (weighted) value at the input node of the edge. There are $NM-1$ iterations for reducing an $N \times M$ image to a single node, associated with a unique real value for the reduced image. The complete reduction process can thus be stored in two column vectors: one of them encodes the $(NM-1)$ edges and the other encodes the $(NM-1)$ detail coefficients, which can be interpreted as intensity differences. Both of the vectors are constructed element by element, from bottom to top. In addition, the (very) top element of either vector stores, respectively, a degenerate edge linking the remaining node to itself, and the associated value of intensity. Those two vectors combined with the spatial information stored in the codebook define the SHAH transform of the image. Although more details on the SHAH transform will be provided later on in this Section, an illustration can already be found in Figure 3.

A. The SHAH algorithm

In this section, we provide the algorithmic details of the SHAH transform.

Input: an image described as an Intensity Network: The Intensity Network (IN) of an image $\mathcal{I}$ is defined as a set $\{\mathcal{D}^{(p)}, \mathcal{E}^{\mathrm{IN}}, X^{(p)}\}$, where

- $\mathcal{D}^{(p)}$ is a *codebook*. It encodes the coordinates of the p points in the image, identified by labels $l = 1 \dots p$. Those points are the locations of the p nodes of the network.
- $\mathcal{E}^{\mathrm{IN}}$ is a *graph*. It is a ranked set of E oriented edges $\epsilon_l = (j,k)$, $l = 1 \dots E$, with $j,k \in \{1,\dots,p\}, j \neq k$, identifying the linked nodes. In the case when no natural orientation exists for the edges, any choice is equally convenient but an orientation is required for the transform to be invertible.

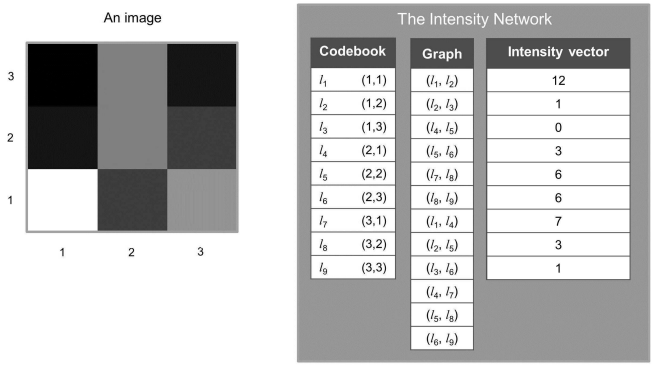

Fig. 2. Right: a typical Intensity Network. Left: the image it refers to. The codebook encodes the *location* of the pixels. The graph structure encodes the *neighbourhood relationships* between the pixels. The intensity vector encodes the *values* of the pixels.

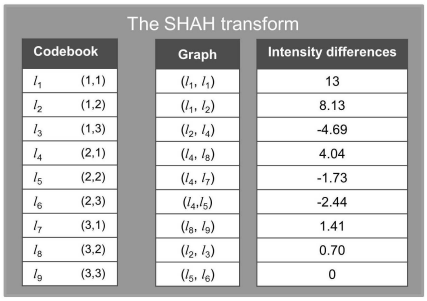

Fig. 3. SHAH of the Intensity Network from Figure 2.

- $X^{(p)}$ is a *vector of intensities*. It is a real-valued vector of length p encoding the intensities of the image $\mathcal{I}$ at the successive points defined in $\mathcal{D}^{(p)}$.

A typical example is as follows.

- The image $\mathcal{I}$ is a gray level image of $N \times M$ pixels encoded as a matrix A.
- $\mathcal{D}^{(p)} = \{(j,k)\}_{j=1\ldots N, k=1\ldots M}$ with j, k defining row and column indices in A. The points are labelled $l = 1 \ldots p$, with $p = NM$.
- $X = \{X_l\}_{l=1\ldots p}$, where $X_l = a_{jk}$ is the gray level of the pixel with coordinates (j,k) associated with the label l in A.

An example is provided in Figure 2. This typical form of the Intensity Network will be used for all examples and applications in this paper. However, more complex networks can also be considered. On the other hand, a simple one-dimensional example of an Intensity Network would be a sampled curve, with edges linking the consecutive measurement points. In this setting, our methodology would permit the analysis of a time series or, more generally, a univariate signal. The one-dimensional version of SHAH, termed Unbalanced Haar (UH) was introduced in [1] and applied to curve classification in [2].

Output: the SHAH transform of the Intensity Network: The SHAH transform of an image $\mathcal{I}$ is defined as the set $\{\mathcal{D}^{(p)}, \mathcal{E}^{\mathrm{OUT}}, d^{(p)}\}$, where

- $\mathcal{D}^{(p)}$ is the same *codebook* as in the input.
- $\mathcal{E}^{\mathrm{OUT}}$ is a graph. It is a ranked set of p oriented edges $\epsilon_l = (j,k)$, $l = 0 \ldots p-1$, with $j, k \in \{1, \ldots, p\}, j \neq k$ identifying the linked nodes. Edge ϵ_0 links to itself and $j = k$ for this edge.
- d is a vector of *intensity differences*. It is a real-valued vector of length p encoding the intensity differences associated with the edges successively defined in $\mathcal{E}^{\mathrm{OUT}}$. The value d_0 is an intensity instead of an intensity difference.

As an example, the output of the SHAH transform of the Intensity Network from Figure 2 is in Figure 3.

The algorithm: The algorithm, detailed below, is also illustrated in Figure 4.

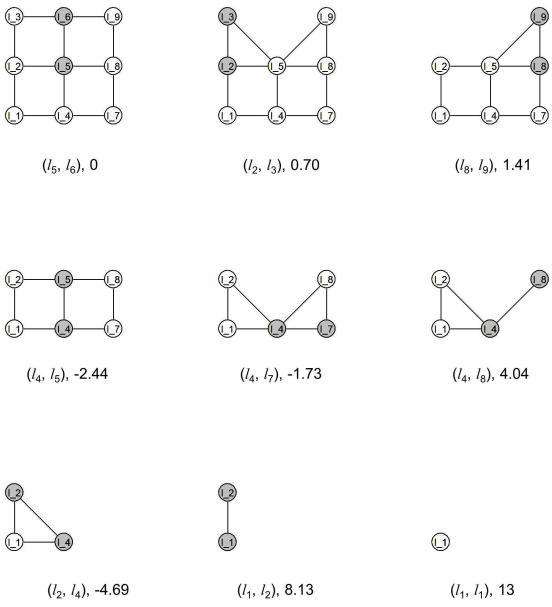

Fig. 4. A schematic illustration of SHAH applied to the image from Figure 2. The network is iteratively reduced by one node at each iteration. The nodes selected for reduction are indicated in gray. The labels of the input and output nodes as well as the detail coefficients returned at iteration k are indicated below each image.

The SHAH algorithm

Input:
INTENSITY NETWORK$= \{\mathcal{D}^{(p)}, \mathcal{E}^{\mathrm{IN}}, X\}$
Output:
SHAH$= \{\mathcal{D}^{(p)}, \mathcal{E}^{\mathrm{OUT}}, d\}$
Notation:
Index i tracks the current iteration;
$\mathcal{E}^{(i)}$ is the set of edges in the network at iteration i: $\mathcal{E}^{(i)} = \{\epsilon_l = (j,k)\}$
$X^{(i)}$ is the value of the nodes remaining in the network at iteration i.
$j^{(i)} = \{w_l\}_{l=1\ldots p-i}$ is a set of weights associated with the $p-i$ nodes remaining in the network at iteration i.
Initialization:
$i := 1$;
$\mathcal{E}^{(1)} := \mathcal{E}^{\mathrm{IN}}$
$X^{(1)} := X$
for $j = 1 \ldots p$, $w_j := 1$.
Iteration #i:
1) Compute 'details' d_l along each of the edges $\epsilon_l = (j,k)$ in $\mathcal{E}^{(i)}$:

$d_l := \frac{w_j}{\sqrt{w_j^2+w_k^2}} X_k - \frac{w_k}{\sqrt{w_j^2+w_k^2}} X_j$.

2) Select an edge ϵ_{l^*} with the minimum absolute value of detail:
$l^* := \arg\min_l |d_l|$.
In case of multiple equal minimum values $|d_l|$, select the smallest index l. Note $\epsilon_{l^*} = (j^*, k^*)$.
3) 'Smooth':
$X_{j^*} := \frac{w_{j^*} X_{j^*} + w_{k^*} X_{k^*}}{\sqrt{w_{j^*}^2+w_{k^*}^2}}$.
$X_{k^*} := X_{j^*}$.
4) Encode the partial value of SHAH:
$\mathcal{E}_{p-i}^{\mathrm{OUT}} := (j^*, k^*)$.
$d_{p-i} := d_{l^*}$.
5) Reduce the network and prepare next iteration:
Update $\mathcal{E}^i$ by replacing all indexes k^* by j^*.
Discard duplicate edges in $\mathcal{E}^i$, retaining only the first occurrence of each edge.
This defines $\mathcal{E}^{(i+1)}$.

$$w^{(i+1)} := \{ w_1, \ldots, w_{j^*-1}, \sqrt{w_{j^*}^2 + w_{k^*}^2}, w_{j^*+1}, \ldots, w_{k^*-1}, w_{k^*+1} \ldots w_{p-i+1} \}.$$

$$X^{(i+1)} := \{ X_1, \ldots, X_{j^*-1}, X_{j^*}, X_{j^*+1}, \ldots, X_{k^*-1}, X_{k^*+1} \ldots X_{p-i+1} \}.$$

$i := i + 1$.
6) Back to Step 1, until length$(X^{(i)}) = 1$.

Final step:
$\mathcal{E}_0^{\mathrm{OUT}} := (j^*, j^*)$.
$d_0 := \frac{X^{(p)}}{\sqrt{p}}$.

B. Computational complexity

In the version described above, the computational complexity of the SHAH algorithm is quadratic in the number of pixels, i.e. is of computational order p^2. This is because at each iteration i, all the edges are examined.

However, other variants of the SHAH algorithm are possible, with substantially reduced computational complexity. We outline some ideas below; they will be reported in detail elsewhere.

- *Examination of a fixed number of edges.* Substantial computational cost can be saved if only a pre-set number of edges (not exceeding a constant), are examined at each iteration i. The edges can be selected in a deterministic or random way. This would result in an algorithm of computational order p, i.e. linear in the number of pixels.
- *Two- or multi-stage algorithm.* For an image of size $N \times N$, firstly divide the image into $(N/k)^2$ non-overlapping sub-images, each of size $k \times k$. Execute the algorithm on each sub-image separately (stage 1), then execute it on the resulting $N/k \times N/k$ matrix of coefficients d_0 from each sub-image (stage 2). The computational complexity is then $(N/k)^2 k^4 + (N/k)^4$, which attains its minimum when $k = N^{1/3}$, resulting in the complexity of $N^{8/3} = p^{4/3}$. The algorithm can be executed similarly in more stages than one, bringing the computational complexity arbitrarily close to linear, if the number of stages is large enough.
- *Removal of multiple nodes at once.* In the version described above, one node is removed at each iteration. An alternative might be to remove multiple nodes, corresponding to a number of smallest detail values. For example, removing half of the remaining nodes at each iteration brings the complexity of the algorithm within a logarithmic factor of linear.

If, in addition to the output described in Section II-A, the SHAH algorithm stores the filter coefficient $w_{j^*}/\sqrt{w_{j^*}^2 + w_{k^*}^2}$ used at each iteration i, the inverse SHAH transform is performed by simply reversing the steps of the SHAH algorithm. The computational complexity of the inverse SHAH transform is then linear in the number of pixels.

C. Properties of SHAH

In this section, we briefly summarise the key mathematical properties of SHAH. The proofs are straightforward, so we omit them.

1) *SHAH as a data-driven orthonormal decomposition of the image.* At iteration i of the SHAH algorithm, each d_{p-i} can be represented as the inner product of the original image X and an image Ψ_{p-i}, where
   - Ψ_{p-i} is selected in a data-driven way at each iteration i,
   - Ψ_{p-i} has mean zero, except when $i = p$,
   - Ψ_{p-i} is orthonormal to all previously selected Ψ_k, $k > p - i$.

   Therefore, $\{\Psi_k\}_{k=0}^{p-1}$ is an orthonormal basis and
   $$X = \sum_{k=0}^{p-1} d_k \Psi_k. \quad (1)$$
   Further, due to the Parseval identity, the total energy (i.e. the sum of squares) of X equals $\sum_{k=0}^{p-1} d_k^2$. An example of the basis Ψ_k is provided in Figure 5. The orthonormality of Ψ_k is a simple consequence of the orthonormality of the 'detail' and 'smooth' filters used at each iteration of the algorithm. SHAH is, obviously, an invertible transform.
2) *Hierarchical nature and Haar-like character of the basis* Ψ_k. Let supp(Ψ_k) denote the support of Ψ_k, i.e. the domain on which it is non-zero.
   - For each $k = 1, \ldots, p-1$, supp(Ψ_k) consists of two contiguous adjacent zones such that Ψ_k is constant positive on one and constant negative on the other. Ψ_0 is positive and constant on the entire domain.
   - The structure of the basis Ψ_k is hierarchical is the sense that if the supports of Ψ_l and Ψ_k overlap and $l < k$, then supp(Ψ_k) must be contained either within the zone where Ψ_l is positive or the zone where it is negative.

   These properties are reminiscent of the Haar wavelet basis. However, here, the key difference is that the supports of Ψ_k are determined by the data and can have arbitrary contiguous shapes, as is apparent from the example in Figure 5. This is because the basis images Ψ_k are chosen adaptively from the data at each iteration of the algorithm.
3) *Sparsity of representation and energy concentration.*
   - For each $k = 1, \ldots, p-1$, if supp(Ψ_k) is contained within a region where X is constant, then the

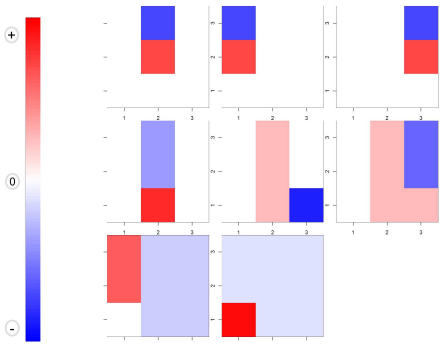

Fig. 5. Basis functions $\{\Psi_k\}_{k=p-1\ldots1}$ for the image of Figure 2, obtained in the order of their construction, from rank $k = p-1$ (*top left*) to rank $k = 1$ (*bottom right*). The remaining basis function Ψ_0 is constant. The basis functions are orthonormal and, except for Ψ_0, reflect level changes between contiguous zones in the image.

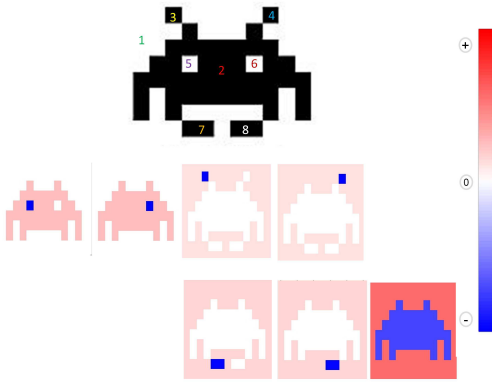

Fig. 6. The "space invader" image (top row), being a black and white image with 8 zones of contiguous identical values, numbered from 1 to 8. 2nd and 3rd rows: basis functions Ψ_k associated with the non-zero detail values d_k, from rank $k = 7$ (*second row, left*) to $k = 1$ (*third row, right*).

corresponding $d_k = 0$. This is a consequence of the mean-zero property of Ψ_k.

- Consequently, by the construction of the SHAH algorithm, for a piecewise-constant image X, the final elements of the vector $(d_0, d_1, \ldots, d_{p-1})$ will be zero. The only non-zero elements, besides possibly d_0, will be $d_1, \ldots d_{Z-1}$, where Z is the number of zones of contiguous identical values in X, the notion of contiguity being defined by the linkage structure of the network. Therefore, SHAH encodes the edges of such an image in the sparsest possible way. See Figure 6 for an illustration.
- For non-piecewise-constant (e.g. noisy) images, the SHAH algorithm is an attempt to achieve the same effect, i.e. to concentrate as much energy of the image X in as few initial coefficients $d_1, d_2, \ldots$ as possible, and therefore to represent its significant features sparsely. An illustration of this property will be provided in Section V.

We also mention that SHAH can be seen as a process of iterative dimension reduction of the image, with minimum loss of information at each step. Alternatively, it can be interpreted as an agglomerative-type algorithm where pixels, each initially forming a separate zone, get progressively grouped into contiguous zones according to a specific criterion.

We end this section by defining a particular arrangement of the information contained in the output of SHAH, which we term the *signature* of the image. We define is as a matrix of dimension $p \times 5$, whose kth row, $k = 0 \ldots p-1$, is related to rank k in the basis expansion (1). Its first two elements contain the Cartesian coordinates (x_1, y_1) of the input node at rank k, next two – the Cartesian coordinates (x_2, y_2) of the output node at rank k. The final element is d_k. An example is given in Figure 7. This notation will be useful to us in the next sections.

	Input (x_1,y_1)		Output (x_2,y_2)		d	
Construction	l_1	(1,1)	l_1	(1,1)	13	Rank
	l_1	(1,1)	l_2	(1,2)	8.13	
	l_2	(1,2)	l_4	(2,1)	-4.69	
	l_4	(2,1)	l_8	(3,2)	4.04	
	l_4	(2,1)	l_7	(3,1)	-1.73	
	l_4	(2,1)	l_5	(2,2)	-2.44	
	l_8	(3,2)	l_9	(3,3)	1.41	
	l_2	(1,2)	l_3	(1,3)	0.70	
	l_5	(2,2)	l_6	(2,3)	0	

Fig. 7. The signature of the image from Figure 2.

III. The Bagidis semi-distance between images

The Bagidis semi-distance for curves was introduced in [2] as a data-driven and highly adaptive way for comparing curves with possibly misaligned sharp local patterns. Its good classification and discrimination performance was reported in [2], [6], [7]. In this section, we extend Bagidis to images.

Bagidis for curves takes as its input the Unbalanced Haar [1] decompositions of both curves. Similarly, Bagidis for images will operate on the SHAH transformations of both images. Bagidis does not require preliminary knowledge of what the images may represent. This sets it apart from many methods based on *landmarks*, which are designed for images containing particular features.

A. Definition

The Bagidis semi-distance provides a measure of the dissimilarity of pairs of images observed on the same grid of measurements. The compared images are first described through their Intensity Networks, with an identical codebook and an identical graph component. The SHAH transforms of both images are then computed and represented as their signatures $\boldsymbol{s}_k^{(i)} = (x_{1k}^{(i)}, y_{1k}^{(i)}, x_{2k}^{(i)}, y_{2k}^{(i)}, d_k^{(i)})$ for $i = 1, 2$. Bagidis compares images $\mathcal{I}^{(1)}$ and $\mathcal{I}^{(2)}$ hierarchically in the following way.

$$
\begin{aligned}
& d^{\text{BAGIDIS}}(\mathcal{I}^{(1)},\mathcal{I}^{(2)}) \\
= & \sum_{k=0}^{p-1}\Big(\alpha_k(x_{1k}^{(1)}-x_{1k}^{(2)})^2+\beta_k(y_{1k}^{(1)}-y_{1k}^{(2)})^2 \\
+ & \gamma_k(x_{2k}^{(1)}-x_{2k}^{(2)})^2+\delta_k(y_{2k}^{(1)}-y_{2k}^{(2)})^2 \\
+ & \epsilon_k(d_k^{(1)}-d_k^{(2)})^2\Big)^{1/2} \\
= & \sum_{k=0}^{p-1}\left\|\boldsymbol{s}_k^{(1)}-\boldsymbol{s}_k^{(2)}\right\|_{2,\boldsymbol{p}_k} \qquad (2)
\end{aligned}
$$

where $\boldsymbol{p}_k=(\alpha_k,\beta_k,\gamma_k,\delta_k,\epsilon_k)$ is a vector of parameters in $\mathbb{R}_+^5$. Notation $\|\cdot\|_{2,\boldsymbol{p}_k}$ indicates that the contribution of rank k to the distance is a semi-2-norm, weighted by the parameters $\boldsymbol{p}_k$. The semi-norm property arises since the parameters $\alpha_k,\beta_k,\gamma_k,\delta_k$ and ϵ_k can be 0 so that $\left\|\boldsymbol{s}_k^{(1)}-\boldsymbol{s}_k^{(2)}\right\|_{2,\boldsymbol{p}_k}=0$ does not necessarily imply $\boldsymbol{s}_k^{(1)}=\boldsymbol{s}_k^{(2)}$. However, (2) satisfies the triangle inequality and the property that $d^{\text{BAGIDIS}}(\mathcal{I},\mathcal{I})=0$.

Such rich parameterization tends to be unmanageable in practice. The following more constrained form might be more appropriate:

$$
\begin{aligned}
& d^{\text{BAGIDIS}}(\mathcal{I}^{(1)},\mathcal{I}^{(2)}) \\
= & \sum_{k=0}^{p-1} w_k\Big(\lambda_x\big((x_{1k}^{(1)}-x_{1k}^{(2)})^2 \\
+ & (x_{2k}^{(1)}-x_{2k}^{(2)})^2\big)+\lambda_y\big((y_{1k}^{(1)}-y_{1k}^{(2)})^2 \\
+ & (y_{2k}^{(1)}-y_{2k}^{(2)})^2+\lambda_D(d_k^{(1)}-d_k^{(2)})^2\Big)^{1/2} \qquad (3)
\end{aligned}
$$

where w_k is a non-negative weight function and λ_x, λ_y, λ_D are scaling parameters satisfying $0\le\lambda_x,\lambda_y,\lambda_D\le 1$ and $\lambda_x+\lambda_y+\lambda_D=1$. The parameters λ_x, λ_y, λ_D are responsible for variations along the x axis, along the y axis, and changes in the intensity, respectively. (3) is a direct extension of BAGIDIS for curves from [2].

B. Discussion of the BAGIDIS semi-distance

BAGIDIS compares the signatures of the two images, which can be viewed as "paths" in the $\mathbb{R}^5$ space. The hierarchy induced by SHAH makes the paths comparable from one image to another. One might expect the initial parts of the paths to reflect the main (most important) features of the images, and their final parts to reflect patterns of decreasing importance and ultimately noise.

The choice of the parameters $\boldsymbol{p}_k$ is key. As is often the case in this type of problems, the selection rule should be flexible enough for BAGIDIS to be able to extract relevant information about the discriminative patterns in the images, but on the other hand constrained enough to prevent overfit. Later, we discuss the choice of $\boldsymbol{p}_k$ in some practical settings. In addition, $\boldsymbol{p}_k$ may be used to incorporate prior knowledge of the differences between the images. Finally, $\boldsymbol{p}_k$ can also be used as an exploratory tool for diagnosing what makes the images different.

Thanks to the SHAH transform underlying the BAGIDIS semi-distance, the latter fully takes into account the notion of neighbourhood between the pixels, which permits the identification of important regions/features in the image and ultimately their comparison in a simple way. A key property is that BAGIDIS compares both the locations and intensities of the patterns. Finally, we emphasise the fact that BAGIDIS does not require any preliminary smoothing or registration of the images.

C. Practical use of BAGIDIS

BAGIDIS can fit within any distance-based method for dataset visualization, unsupervised discrimination or prediction in a supervised setting. Multidimensional scaling [3], Ward's hierarchical algorithm [8] and nonparametric functional regression [4] are examples of algorithms that can be used with BAGIDIS in those different areas, respectively. Furthermore, the descriptive statistics and tests, specifically designed for BAGIDIS for curves [2], can be extended to images. Key to all these uses is a suitable selection of the parameters to be used in formula (3), which is typically achieved through an optimization strategy, and done differently depending on whether the problem falls into the 'supervised' or 'unsupervised' learning category. We now discuss the two cases in detail.

Supervised learning: Leave-one-out cross-validation within a training set, optimising either e.g. the mean square error (MSE) of prediction (in regression problems) or the number of misclassifications (in classification problems), can be used to select the parameters of BAGIDIS in supervised learning. Theoretical efficacy of such a scheme has been proved in the case of functional nonparametric kernel regression [4] in [6]. To save computational time, the examples of Section V use a CV scheme where the weights w_k in (3) are chosen to be a step function taking the value of 1 up to a certain rank k (to be chosen via CV) and 0 thereafter. A computationally cheaper alternative, also illustrated in Section V, might be to only select those components of the signature that significantly relate to the response, e.g. through a significant correlation with the response in the case of regression problems.

Unsupervised learning: In unsupervised learning, we use the following procedure to select the parameters of BAGIDIS in Section V. Consider the signatures $(x_{1k}^{(i)},y_{1k}^{(i)},x_{2k}^{(i)},y_{2k}^{(i)},d_k^{(i)})_{k=0}^{p-1}$ of each of the $i=1,\dots,N$ images. We view them as $5p$ vectors of length N, which define the $5p$ variables, each of which may or may not be included in the computation of the distance, depending on the parameterization. We use vector quantization (in this case the K-MEANS algorithm [9]) to define groups of variables displaying similar behaviour. A suitable number of groups is chosen in a heuristic way using an elbow criterion. In each group one variable is randomly chosen as a representative and is set to be *active* (i.e. it is associated with parameter 1), while the other ones are set to be *inactive* (i.e. associated with 0) in the semi-distance. The hope is that variables with a significant discriminative role will have a sufficiently different behaviour to form a group by themselves. Similarly,

it is hoped that the variables that do not carry discriminative information will be sufficiently grouped so that they do not have much weight in the semi-distance. This procedure drastically reduces the dimensionality of the semi-distance. It may be tested several times with different random choices of the representatives in each group, in order to assess the stability of the result.

IV. Literature review

The SHAH transform falls into the category of "multiscale representation of images". (Nonadaptively selected) wavelet bases are a canonical example of a tool used to achieve such representations, and early surveys of their use in image processing can be found in [10] or [11]. Wavelets, although widely used and relatively well understood, suffer from inefficacies in capturing non-horizontal or -vertical features in an image; *curvelets* [12] attempt to remedy this by using a more flexible family of building blocks, which are also not selected adaptively.

In contrast to wavelets or curvelets, the building blocks Ψ_k of the SHAH transform are selected adaptively from the data. A recent review of adaptive image representations can be found in [13]. The principle of adaptivity (although not the particular construction used in SHAH) is shared by a number of "-let" transforms, including *bandlets* of [14] (see also [15] for a review of related techniques), *wedgelets* [16], [17], *tetrolets* [18] and the *Easy Path Wavelet Transform* [19]. *Grouplets* [20] preserve the classical notion of scale and grid subdivision present in the Haar or lifting transforms (see below for references to lifting), but equip the standard Haar transform with an "association field" that groups together points that are not necessarily neighbours. This leads, in a context different from that in SHAH, to similar Haar-like filtering operations with not necessarily equal weights as those in SHAH. We emphasise that in contrast to grouplets, SHAH completely circumvents the classical notion of wavelet scale. Other approaches to image processing (in this case, denoising) which can be viewed as adaptive but do not use the notion of decomposition or hierarchy are, for example, adaptive weight smoothing [21] and penalized regression on a graph [22]. A recent review of image denoising techniques can be found in [23].

Regarding the prediction of an external response from an image, one may distinguish four broad families of methods, mostly originating from image classification and pattern recognition literature. Firstly, methods which compare images through *landmarks*, i.e. specific features present in each image in a dataset, such as e.g. facial features (eyes, nose, mouth) in face recognition problems [24]. The SHAH+BAGIDIS methodology does not fall into this class as it does not require any prior registration of features. Moreover, it does not require a preliminary knowledge of what the image represents. Secondly, methods that compare images through suitably chosen vectors of features, such as colour histograms, measures of image coarseness, contrast, etc., computed either for the entire image [25] or localised [26], and then used in a distance measure. The key difference with SHAH is that in this class of methods, the dimensionality reduction process is non-adaptive. Thirdly, methods that compute distances between the images directly, without relying on a preliminary dimension reduction. These include the Euclidean (L_2) Manhattan (L_1) and Hausdorff [27] distances. Finally, methods that initially use dimension reduction via e.g. principal components [4], which we feel SHAH+BAGIDIS bears the most similarity to, although we emphasise again that one unique feature of SHAH is that the decomposition it provides is into a set of particularly simple functions, which may make it more interpretable than these techniques. Moreover, as discussed in [2] and [6] in the framework of curve comparison, computing a distance between principal components is a powerful technique as long as the significant patterns to be compared are perfectly aligned across a dataset. This limitation was the initial motivation for defining the BAGIDIS methodology for curves [2]. Another interesting robust technique is IMED, [28]; however, unlike BAGIDIS, it cannot be tuned to extract specific patterns in the image. The problem of misalignement of patterns in images is often tackled by registering the images [29] before computing a distance between them. However, this implies that spatial misalignment cannot be used as discriminative signal (unlike in BAGIDIS).

We end this section with a brief review of existing wavelet-like methods for graphs outside of the image context. [30] and [31] define wavelets on graphs by studying eigenvalues of the graph Laplacian; the latter takes the form of a matrix encoding the connectivity of each node and edge. [32] uses the powers of a diffusion operator as the scaling tool leading to multiscale analysis. Several variants of their ideas [33], [34] lead to different wavelet constructions. [35] uses the n-hop distance (the minimal number of edges one has to travel to go from the central node to another) to define wavelets on the graph. [36] uses the lifting algorithm akin to that of [37] to construct wavelets on graphs using a bottom-up approach where wavelets between the nearest nodes get constructed first. Some authors also have defined wavelet transforms specifically designed for the dendrogram: [38] uses Haar bases, while [39] generalizes to unbalanced Haar. [40] iteratively reduces the graph by replacing two (groups of) nodes by a single one, but unlike in SHAH, the graph structure is not used in the reduction process. The latter method is closely related to the idea behind *treelets* [41], defined for unordered data.

V. Applications

A. The multizone noisy image

This example illustrates the ability of SHAH to segment a noisy multizone image. We consider the image in Figure 8, containing 6 different zones whose intensities vary between 0 and 50. We contaminate it with zero-mean Gaussian noise, with standard deviations σ set to $1, 5, 8$. For each of those noisy images, the SHAH transform is performed. If there were no noise, the only basis matrices carrying the signal would be $\Psi_0, \Psi_1, \ldots, \Psi_5$. Figures 9, 10 and 11 show the three noisy images, with successive levels of noise, and the associated basis matrices Ψ_1, $\Psi_2 \ldots \Psi_5$ (the basis matrix Ψ_0 is always constant, by construction). Perfect extraction is achieved for

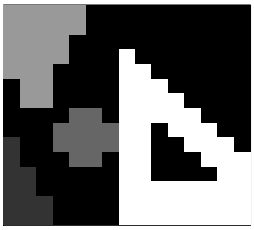

Fig. 8. Multizone image without noise. The intensities of the pixels are recorded on a black to white scale from 0 to 50.

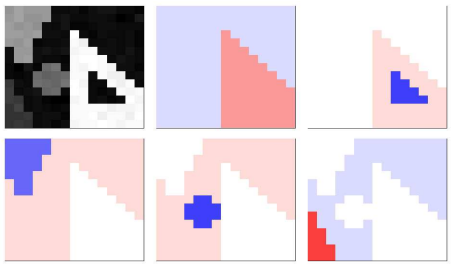

Fig. 9. *From top left to bottom right, by row:* Multizone image with $N(0,1)$ noise; its basis matrices Ψ_1 to Ψ_5. The intensities of the basis matrices are on a blue to red scale with white representing 0.

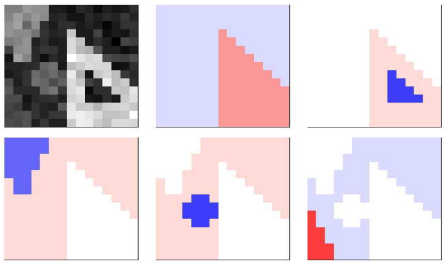

Fig. 10. *From top left to bottom right, by row:* Multizone image with $N(0,5)$ noise; its basis matrices Ψ_1 to Ψ_5. The intensities of the basis matrices are on a blue to red scale with white representing 0.

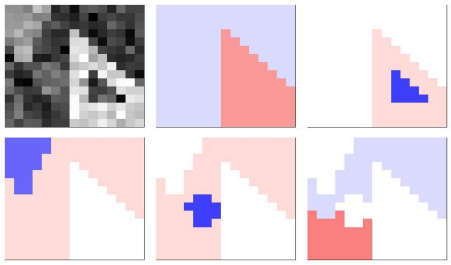

Fig. 11. *From top left to bottom right, by row:* Multizone image with $N(0,8)$ noise; its basis matrices Ψ_1 to Ψ_5. The intensities of the basis matrices are on a blue to red scale with white representing 0.

$\sigma = 1$ and $\sigma = 5$. When $\sigma = 8$, the algorithm still detects the significant zones, although the shapes of the two least significant ones (the cross in the middle and the triangle in the bottom left) are not reproduced exactly. However, it is remarkable that these zones have been identified at all, given that even their visual identification in the noisy image is challenging.

B. The moving pixel

We now illustrate the ability of BAGIDIS (versus competitors) to capture the relevant variations of a signal regarding the location or the intensity of a simple pattern in a noisy setting.

Three test scenarios are considered. They all involve 5×5 images whose noiseless pixels have values zero except at one randomly selected location (the "moving pixel") whose intensity varies according to the scenario. In each case, a dataset of 90 images (70 training + 20 test) is generated and Gaussian noise with standard deviation σ is added. Each image is related to a scalar value Y, determined according to the scenario. The goal is to predict Y.

- **Scenario 1: Capturing the location of a moving pixel.** The noiseless intensity of the moving pixel is 10; Y is the row index of the moving pixel. Maximum σ is 4.
- **Scenario 2: Capturing the location of a moving pixel with varying intensity.** The noiseless intensity of the moving pixel is chosen randomly from the set $\{5, 10, 15, 20, 25\}$ with equal probabilities; Y is the row index of the moving pixel. Maximum σ is 10.
- **Scenario 3: Capturing both the location and the intensity of a moving pixel.** The noiseless intensity of the moving pixel and the maximum value of σ are as in Scenario 2; Y is the sum of the intensity and the row index of the moving pixel.

In each scenario, we estimate a nonparametric functional regression model [4] on the training set, using BAGIDIS and its competitors (below). The mean square prediction error over the test images is recorded, and averaged over 10 random test/training splits.

The parameterization of the BAGIDIS semi-distance is chosen by considering the correlations ρ between each component of the signature (including the absolute value of the details) at each rank, and the response Y. A test of significance is applied on the correlations, with $H_0 : \rho = 0$ and $H_1 : \rho \neq 0$ with significance level $\alpha = 0.1$.

A PCA-based semi-distance, with $k = 1$, 3 or 6 components, is obtained as follows:

$$d_k^{PCA}(\mathcal{I}^{(1)}, \mathcal{I}^{(2)}) = \sqrt{\sum_{q=1}^{k} \left(\sum_{x=1}^{N} \sum_{y=1}^{M} (\mathcal{I}^{(1)}(x,y) - \mathcal{I}^{(2)}(x,y)) \hat{v}_q(x,y) \right)^2},$$

where $\hat{v}_q$, $q = 1 \dots k$ is the q^{th} estimated eigenfunction and the pair (x, y), $x = 1 \dots N$, $y = 1 \dots M$, identifies the location of a pixel. The IMED distance is used in its non-parameterized form (eq. (6) in [28]). No parameterization is needed for the Euclidean or the Hausdorff distance. The bandwidth of the regression estimator is chosen by cross-validation in each case.

The results are illustrated in Figure 12 (as a function of σ). The Euclidean distance and the PCA-based semi-distance perform poorly, which is unsurprising as the images are misaligned. IMED is quite competitive in Scenario 1 for low noise levels, due to its robustness to misalignment. However, it fails as soon as the intensity of the moving pixel varies

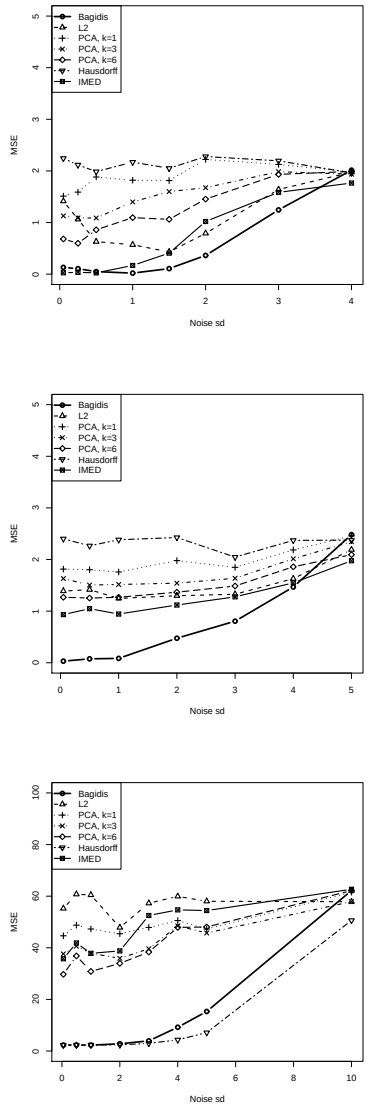

Fig. 12. The moving pixel. Mean-square prediction error averaged over 10 repetitions, as a function of σ, in Scenarios 1, 2, 3 (respectively from top to bottom), for the competing methods.

(Scenarios 2, 3) as it cannot decouple the information about intensity and location (which BAGIDIS is able to achieve). The Hausdorff distance is the most successful in Scenario 3, with BAGIDIS not far behind. However, the Hausdorff distance does poorly in Scenarios 1 and 2 as it is not able to distinguish between two images with a moving pixel of the same intensity but at different locations. Its good performance in Scenario 3 is due to the fact that the part of the response that concerns the location (with values in $\{1, 2, 3, 4, 5\}$) is dominated by the part of the response related to intensity (with values > 5). In summary, BAGIDIS is either the best or close to best in all experimental settings.

C. Handwritten digits

In this Section, we use SHAH to analyse (both in a supervised and in an unsupervised way) images of handwritten digits zero and one from the publicly available MNIST database (http://yann.lecun.com/exdb/mnist/). The database contains 70000 black-and-white 28x28 images of digits. We use Brendan O'Connor's code from https://gist.github.com/39760 to read the data into R. We restrict ourselves to a small subset of images of zeros and ones from the database.

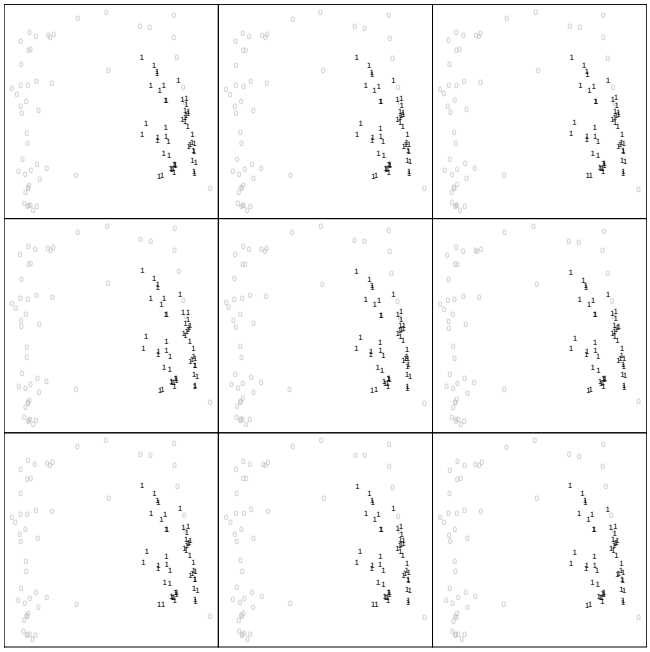

Fig. 13. Unsupervised learning for the handwritten digits recognition problems. 9 multidimensional scaling representations of a dataset of 100 images, with different random choices of the activated coefficients in the groups determined by the K-MEANS algorithm.

Unsupervised learning: We consider 50 images of zeros and 50 images of ones. We compute the set of pairwise BAGIDIS distances between all images, with a parameterization chosen according to a K-MEANS algorithm as described in Section III-C. We then use multidimensional scaling to produce a 2D projection, in which the points are labelled 0 or 1 according to which digit they represent. The procedure is applied to two different datasets and repeated 9 times for each, with different random choices in the K-MEANS algorithm (see Section III-C for details). Figure 13 shows the results for the first dataset; those for the other one are very similar. The results are stable with respect to the random choices in the K-MEANS algorithm. All 1s and most 0s form separate well-identified groups, with a few 0s appearing to be misclassified; despite this, we find this result very encouraging for an unsupervised analysis with an unsupervised choice of the parameterization.

Supervised learning: A nonparametric functional discrimination model [4], with a k-NN algorithm, is trained on a dataset of 70 images of ones and 70 images of zeros. It is then tested on an independent set of 15 ones and 15 zeros. The number k of neighbours is selected using a local cross-validation (routine `funopadi.knn.lcv` from the companion website of [4], adapted for BAGIDIS.) We also perform cross-validation to choose $(\lambda_x, \lambda_y, \lambda_D)$ in formula (3); they are chosen from the set $\{0; \frac{1}{3}; \frac{2}{3}; 1\}$ under the constraint that they sum to 1. Each combination of these values is tested with

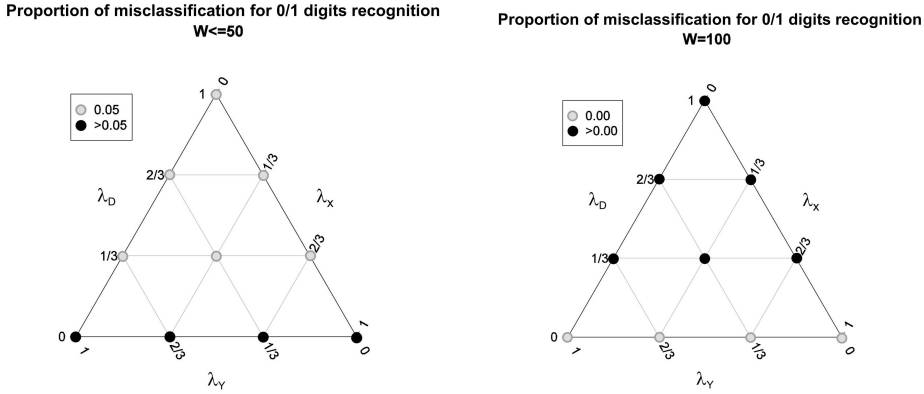

Fig. 14. Handwritten digits recognition: Investigation of the effects of the parameters and restricted cross-validation procedure. Left panel: results when $W \in \{2; 5; 10; 20; 50\}$; right: $W = 100$.

several weight functions w_k, the latter being step functions with values 1 up to a rank $k = W$ and 0 thereafter; $W = 2, 5, 10, 20, 50$ and 100 are tested. This experimental design is repeated twice with different training sets and test sets. The proportion of misclassification on the test set is recorded for each configuration. Results are presented in Figure 14. The results imply that the first ranks of the signature clearly carry some discriminative information in the detail component. However, one needs to look further down in the hierarchy (between ranks 50 and 100) to achieve perfect discrimination. This extra information lies in the components (x_1, y_1) and (x_2, y_2) of the signature, indicating that some information about the geometry of the image is needed. Given this, our recommendation for the choice of the parameterization would be $(W = 100, \lambda_D = 0, \lambda_x = 0.5, \lambda_y = 0.5)$.

Cross-validation is, unfortunately, rather time-consuming. We tested the following faster strategy. We considered the formulation (2) of the semi-distance. Each component of the signatures of the images in the training set was considered separately, rank by rank. A t-test was performed for the difference of the means of the values taken by that component in either group (zero or one) of the images. The p-values were recorded, and only those components with p-values smaller than 10^{-10} were assigned the weight of 1 in the parameterization, with all other weights being set to 0. The thus-trained model was used for discriminating the images in the test set and the number of misclassifications was recorded. The procedure was repeated 8 times with different training sets and test sets. We obtained the average misclassification rate of 0.02.

Overall, we find these results encouraging, especially given the small size of the training set and the fact that no prior information about the images or the shapes of the discriminant patterns are used in at any stage of the procedure.

VI. Conclusion

In this article, we have proposed the SHAH (SHape-Adaptive Haar) transform for images, which results in an orthonormal, adaptive decomposition of the image into Haar-like components, arranged hierarchically according to decreasing importance, whose shapes reflect the features present in the image. The decomposition is extremely sparse for piecewise-constant images. It is performed via an iterative 'generous' (as opposed to 'greedy') bottom-up algorithm with quadratic computational complexity; however, nearly-linear variants also exist. SHAH is rapidly invertible.

Secondly, we have used SHAH to define the BAGIDIS semi-distance between images. It compares both the amplitudes and the locations of the SHAH components of the images and is flexible enough to account for feature misalignment. Performance of the SHAH+BAGIDIS methodology was illustrated in regression, classification and clustering problems and shown to be very encouraging.

A clear asset of the methodology is its very general scope: it can be used with any images or more generally with any data that can be described as graphs or networks.

References

[1] P. Fryzlewicz, "Unbalanced Haar technique for non parametric function estimation," *Journal of the American Statistical Association*, vol. 102, no. 480, pp. 1318–1327, 2007.

[2] C. Timmermans and R. von Sachs, "Bagidis, a new method for statistical analysis of differences between curves with sharp local patterns," *under revision*, 2010. [Online]. Available: http://www.stat.ucl.ac.be/ISpub/dp/2010/DP1030.pdf

[3] M. A. Cox and T. F. Cox, "Multidimensional scaling," in *Handbook of Data Visualization*, ser. Springer Handbooks of Computational Statistics, C.-H. Chen, W. K. Hardle, and A. Unwin, Eds. Springer Berlin Heidelberg, 2008, ch. 3, pp. 315–347.

[4] F. Ferraty and P. Vieu, *Nonparametric Functional Data Analysis: Theory and Practice*, ser. Springer Series in Statistics. Secaucus, NJ, USA: Springer-Verlag New York, Inc., 2006.

[5] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning*, ser. Springer Series in Statistics. New York, NY, USA: Springer New York Inc., 2001.

[6] C. Timmermans, L. Delsol, and R. von Sachs, "Using Bagidis in non-parametric functional data analysis: predicting from curves with sharp local features." 2011, submitted. http://hdl.handle.net/2078.1/81200. [Online]. Available: http://hdl.handle.net/2078.1/81200

[7] C. Timmermans, P. de Tullio, V. Lambert, M. Frdrich, R. Rousseau, and R. von Sachs, "Advantages of the bagidis methodology for metabonomics analyses: application to a spectroscopic study of age-related macular degeneration," in *Proceedings of the twelve European Symposium on Statistical Methods for the Food Industry*, 2012, pp. 399–408.

[8] L. Lebart, A. Morineau, and M. Piron, *Statistique exploratoire multidimensionnelle.*, 3rd ed. Dunod, October 2004. [Online]. Available: http://www.worldcat.org/isbn/2100488481

[9] J. A. Hartigan and M. A. Wong, "Algorithm AS 136: A k-means clustering algorithm," *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, vol. 28, no. 1, pp. 100–108, 1979. [Online]. Available: http://dx.doi.org/10.2307/2346830

[10] A. Cohen, "Wavelets and multiscale methods in image processing," 1997.

[11] S. Mallat, *A Wavelet Tour of Signal Processing*, 2nd ed. Academic Press, Sep. 1999.

[12] E. J. Candès and D. L. Donoho, "Curvelets and curvilinear integrals," *Journal of Approximation Theory*, vol. 113, no. 1, pp. 59–90, 2001.

[13] G. Peyr, "A review of adaptive image representations," *IEEE Journal of Selected Topics in Signal Processing*, vol. 5, no. 5, pp. 896–911, 2011. [Online]. Available: http://hal.archives-ouvertes.fr/hal-00519525/

[14] E. Le Pennec and S. Mallat, "Sparse geometrical image representation with bandelets," *IEEE transaction on Image Processing*, vol. 14, no. 4, pp. 423–438, 2005.

[15] S. Mallat and G. Peyr, "Orthogonal bandlet bases for geometric images approximation," *Communications on Pure and Applied Mathematics*, vol. 61, no. 9, pp. 1173–1212, 2008. [Online]. Available: http://hal.archives-ouvertes.fr/hal-00359740/

[16] D. L. Donoho, "Wedgelets: nearly-minimax estimation of edges," *Annals of Statistics*, pp. 859–897, 1999.

[17] R. L. Claypoole and R. G. Baraniuk, "A multiresolution wedgelet transform for image processing," in *SPIE Technical Conference on Wavelet Applications in Signal Processing VIII*, San Diego, Jul. 2000.

[18] J. Krommweh, "Tetrolet transform: A new adaptive haar wavelet algorithm for sparse image representation," *Journal of Visual Communication and Image Representation*, vol. 21, no. 4, pp. 364–374, 2010.

[19] G. Plonka, "The easy path wavelet transform: A new adaptive wavelet transform for sparse representation of two-dimensional data," *Multiscale Modeling and Simulation*, vol. 7, no. 3, pp. 1474–1496, 2009.

[20] S. Mallat, "Geometrical grouplets," *Applied and Computational Harmonic Analysis*, vol. 26, no. 2, pp. 161–180, 2009. [Online]. Available: http://linkinghub.elsevier.com/retrieve/pii/S1063520308000444

[21] J. Polzehl and V. Spokoiny, "Adaptive weights smoothing with applications to image restoration," *Journal of the Royal Statistical Society, Series B*, vol. 62, pp. 335–354, 2000.

[22] A. Kovac and A. Smith, "Nonparametric regression on a graph," *Journal of Computational and Graphical Statistics*, vol. 20, pp. 432–447, 2011.

[23] P. Milanfar, "A tour of modern image filtering," *To appear as a feature article in IEEE Signal Processing Magazine*, 2011. [Online]. Available: http://users.soe.ucsc.edu/∼milanfar/publications/index.html

[24] G. Beumer, Q. Tao, A. Bazen, and R. Veldhuis, "Comparing landmarking methods for face recognition," in *Proceedings of ProRISC 2005, 16th Annual Workshop on Circuits, Systems and Signal Processing*, 2005, pp. 594–597. [Online]. Available: http://doc.utwente.nl/59557/

[25] R. Hu, S. M. Rüger, D. Song, H. Liu, and Z. Huang, "Dissimilarity measures for content-based image retrieval," in *ICME*. IEEE, 2008, pp. 1365–1368.

[26] A. Frome, Y. Singer, and J. Malik, "Image retrieval and classification using local distance functions," in *NIPS*, B. Schölkopf, J. C. Platt, and T. Hoffman, Eds. MIT Press, 2006, pp. 417–424.

[27] D. P. Huttenlocher, G. A. Klanderman, and W. J. Rucklidge, "Comparing images using the hausdorff distance," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 15, pp. 850–863, 1993.

[28] L. Wang, Y. Zhang, and J. Feng, "On the euclidean distance of images," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 8, pp. 1334–1339, 2005.

[29] B. Zitov and J. Flusser, "Image registration methods: a survey," *Image and Vision Computing*, vol. 21, pp. 977–1000, 2003.

[30] D. K. Hammond, P. Vandergheynst, and R. Gribonval, "Wavelets on graphs via spectral graph theory," *Applied and Computational Harmonic Analysis*, vol. 30, no. 2, pp. 129–150, 2009. [Online]. Available: http://arxiv.org/abs/0912.3848

[31] J.-P. Antoine, D. Rosca, and P. Vandergheynst, "Wavelet transform on manifolds: Old and new approaches," *Applied and Computational Harmonic Analysis*, vol. 28, no. 2, pp. 189 – 202, 2010.

[32] R. R. Coifman and M. Maggioni, "Diffusion wavelets," *Applied and Computational Harmonic Analysis*, vol. 21, no. 1, pp. 53–94, 2006.

[33] A. D. Szlam, M. Maggioni, R. R. Coifman, and J. C. J. Bremer, "Diffusion-driven multiscale analysis on manifolds and graphs: top-down and bottom-up constructions," M. Papadakis, A. F. Laine, and M. A. Unser, Eds., vol. 5914, no. 1. SPIE, 2005.

[34] M. Maggioni, J. C. J. Bremer, R. R. Coifman, and A. D. Szlam, "Biorthogonal diffusion wavelets for multiscale representations on manifolds and graphs," M. Papadakis, A. F. Laine, and M. A. Unser, Eds., vol. 5914, no. 1. SPIE, 2005.

[35] M. Crovella and E. Kolaczyk, "Graph wavelets for spatial traffic analysis," in *Proceedings of IEEE Infocom*, Apr. 2003.

[36] M. Jansen, G. Nason, and B. Silverman, "Multiscale methods for data on graphs and irregular multidimensional situations," *Journal of the Royal Statistical Society B, Statistical Methodology*, vol. 71, pp. 97–125, 2009.

[37] W. Sweldens, "Wavelets and the lifting scheme: A 5 minute tour," *Zeitschrift fr Angewandte Mathematik und Mechanik*, vol. 76 (Suppl. 2), pp. 41–44, 1996.

[38] F. Murtagh, "The haar wavelet transform of a dendrogram," *Journal of Classification*, vol. 24, p. 2006, 2007.

[39] M. Gavish, B. Nadler, and R. R. Coifman, "Multiscale wavelets on trees, graphs and high dimensional data: Theory and applications to semi supervised learning," in *ICML*, 2010, pp. 367–374.

[40] A. Singh, R. D. Nowak, and A. R. Calderbank, "Detecting weak but hierarchically-structured patterns in networks," *Journal of Machine Learning Research - Proceedings Track*, vol. 9, pp. 749–756, 2010.

[41] A. B. Lee, B. Nadler, and L. Wasserman, "Treelets - an adaptive multi-scale basis for sparse unordered data," *Annals of Applied Statistics*, vol. 2, pp. 435–471, 2008.