This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2019.2891742, IEEE Transactions on Pattern Analysis and Machine Intelligence

Visibility graphs for image processing

Jacopo Iacovacci[1,2] and Lucas Lacasa[3*]
[1] *Department of Surgery and Cancer, Division of Computational and Systems Medicine, Imperial College London, London SW72AZ (United Kingdom)*
[2] *The Molecular Biology of Metabolism Laboratory, The Francis Crick Institute, London NW11AT (United Kingdom) and*
[3] *School of Mathematical Sciences, Queen Mary University of London, E14NS London (United Kingdom)*

The family of image visibility graphs (IVG/IHVGs) have been recently introduced as simple algorithms by which scalar fields can be mapped into graphs. Here we explore the usefulness of such an operator in the scenario of image processing and image classification. We demonstrate that the link architecture of the image visibility graphs encapsulates relevant information on the structure of the images and we explore their potential as image filters. We introduce several graph features, including the novel concept of Visibility Patches, and show through several examples that these features are highly informative, computationally efficient and universally applicable for general pattern recognition and image classification tasks.

I. INTRODUCTION

Visibility and Horizontal Visibility graphs (VG/HVGs) were proposed in [1–3] as a family of simple mappings between ordered sequences (e.g. time series) and graphs [4], this being one possible strategy to perform graph-theoretical time series analysis (see [5] and references therein for a recent overview). Consider an ordered sequence $\{\mathbf{x}(t)\}_{t=1}^N$, where $\mathbf{x}(t) \in \mathbb{R}^m$. For $m = 1$ a typical case of such sequence are univariate time series describing the activity of some system or one-dimensional dynamical systems, whereas for $m > 1$ we consider multivariate time series or trajectories from high-dimensional dynamical systems. In the former case, a time series of N data is mapped into a graph of N nodes such that two nodes are linked in the graph if particular geometric and ordering criteria hold in the sequence (in the multivariate case, there are m replicas of the set of nodes –each of these replicas representing a different layer– and the edge set is a priori different for each layer, thereby defining a *multiplex visibility graph* [3, 6, 7]). This mapping enables the possibility of performing graph-theoretical time series analysis and hence permits to build bridges between the theories of dynamical systems, signal processing and graph theory.

Theoretical research on visibility graphs has elaborated on mathematical methods [8–11] and some rigorous results on the properties of these graphs when associated to canonical models of complex dynamics have been obtained [12–15]. From a practical point of view, this method has been used as a feature extraction procedure to construct feature vectors from time series for statistical learning purposes (see [16–22] for just a few examples in the life sciences or [23–30] for other applications in the physical sciences).

Very recently [31], this paradigm has been theoretically extended to handle scalar fields. Here we consider the particular two-dimensional case $h(x, y) : \mathbb{R}^2 \to \mathbb{R}$ (this being conceptually closer to the original context of visibility graphs analysis as defined in Urban Planning [32]) and explore from a theoretical and practical point of view its usefulness in image processing and pattern recognition, proposing in this way a novel methodology in the field of *graph-based image analysis* [33]. We will show that, under this extension, visibility graphs can be extracted from images and subsequently used both for image filtering and as a universal feature extraction protocol for supervised learning purposes, complementing more standard approaches in the field [34].

The rest of the paper goes as follows: in section II we define the method of image visibility graphs and present some of its properties, along with a few key metrics which will be relevant as graph-theoretical image descriptors. In section III we explore the performance of image visibility graphs in the tasks of image filtering, and in section IV we evaluate their performance in the tasks of pattern recognition and texture classification. In section V we conclude.

II. IMAGE VISIBILITY GRAPHS

A. Definitions

For completeness, we start by introducing the definitions of two types of visibility graphs extracted from univariate time series, and one of their possible extensions to images.

Definition (VG). *Let* $\mathcal{S} = \{x_1, \dots, x_n\}$ *be a ordered sequence of n real-valued, scalar data. A Visibility Graph* (VG) *is an undirected graph of n nodes, where each node i is labelled according to the time order of its corresponding datum* x_i*. Hence* x_1 *is mapped into node* $i = 1$, x_2 *into node* $i = 2$*, and so on. Then, two nodes i and j (assume* $i < j$ *without loss of generality) are connected by an (undirected) link if and only if one can draw a straight*

---

[*] `j.iacovacci@imperial.ac.uk`; `l.lacasa@qmul.ac.uk`

line connecting x_i and x_j that does not intersect any intermediate datum 'height' x_k, $i < k < j$ (i.e. the vertical column of height x_k). Equivalently, i and j are connected if the following convexity criterion is fulfilled:

$$x_k < x_i + \frac{k-i}{j-i}[x_j - x_i],\ \forall k : i < k < j$$

A similar definition follows for a Horizontal Visibility Graph (HVG), but in this latter graph two nodes i, j (assume $i < j$ without loss of generality) are connected by a link if and only if one can draw a *horizontal* line connecting x_i and x_j that does not intersect any intermediate datum x_k, $i < k < j$. Equivalently, i and j in the HVG are connected if the following *ordering* criterion is fulfilled:

$$x_k < \inf(x_i, x_j),\ \forall k : i < k < j.$$

One can now extend the definition to handle not just ordered sequences but images [31]:

Definition (IVG). *Let $\mathcal{I}$ be a $N \times N$ matrix where $\mathcal{I}_{ij} \in \mathbb{R}$. The Image Visibility Graph* (IVG) *is a graph of N^2 nodes, where each node is labelled by the indices of its corresponding datum $\mathcal{I}_{ij}$, such that two nodes ij and $i'j'$ are linked if*

- $(i = i') \vee (j = j') \vee [(i = i' + p) \wedge (j = j' + p)]$, *for some integer p, and*
- *$\mathcal{I}_{ij}$ and $\mathcal{I}_{i'j'}$ are linked in the VG defined over the ordered sequence which includes ij and $i'j'$.*

The *Image Horizontal Visibility Graph* (IHVG) follows equivalently if in the second condition we make use of the definition HVG instead of VG. Note that the preceding definition indeed coincides with the definition of an IVG/IHVG in the so-called FCC extension class [31], this being one of other possibilities for the analysis of high-dimensional scalar fields, which we adopt here for convenience as this particular one is well-fitted for image processing. Intuitively, the IVG/IHVG extracts a graph from an image by considering pixels as nodes and wiring all nodes which lie in specific directions (rows, columns, diagonals) according to the visibility criteria defined in VG/HVG, see an example of this transformation for the specific case of IHVG in Fig. 2.

B. Preliminary analysis: robustness to noise of IVG and IHVG

We initially explore the robustness against noise pollution of the family of image visibility graphs. A natural way to quantify this is to measure the overall link overlap $\mathcal{Q}$ between the image visibility graph extracted from a given image and the respective image visibility graph extracted from the image contaminated with white Gaussian noise of different power noise –as quantified by a noise to signal ratio (NSR)– i.e.

$$\mathcal{Q} = \frac{\sum_{ij} A_{ij} A_{ij}^{\text{NSR}} - \mathcal{Q}_{\min}}{\sum_{ij} A_{ij} - \mathcal{Q}_{\min}} \tag{1}$$

where $\mathbf{A} = \{A_{ij}\}$ and $\mathbf{A}^{\text{NSR}} = \{A_{ij}^{\text{NSR}}\}$ denote the adjacency matrices associated to the image visibility graph from the original image and the polluted image, respectively (we recall that an adjacency matrix is a binary matrix where $A_{ij} = 1$ if nodes i and j are connected, and 0 otherwise). $\mathcal{Q}_{\min}$ accounts for the number of links that *by construction* two IVG/IHVGs of the same size always share because of the common lattice structure: $\mathcal{Q}_{\min} = 4(N-1)N + 4(N-1)^2 \sim 8N^2$.

As defined in Eq.1, the link overlap $\mathcal{Q}$ could a priori be used to assess the similarity between two image visibility graphs, however this quantity doesn't saturate to zero when comparing totally uncorrelated images e.g. two instances of spatial white noise with adjacency matrices A_{ij} and A'_{ij}. This last statement can be rigorouly justified as follows: from [31] we know that the degree distribution of an IHVG associated to a spatial white noise image is $P(k) = (1/9)(8/9)^{k-8}$, hence for a $N \times N$ image we get $\sum_{ij} A_{ij} = \sum_{ij} A'_{ij} \approx \sum_{k=8}^{\infty} kP(k)N^2 = 16N^2$. On the other hand, since the two images are in this case totally uncorrelated, then $\sum_{ij} A_{ij} A'_{ij}$, which computes the average number of common links between the visibility graphs of both images is analytically solvable. Let π_{ij} be the probability that nodes i and j are linked in the image visibility graph, then $\sum_{ij} A_{ij} A'_{ij} = \sum_{i=1}^{N} \sum_{j=1, j\neq i}^{N} \pi_{ij}^2$. Now, (i) for links of the common lattice structure $\pi_{ij} = 1$, (ii) for links outside the common lattice structure such that i and j belong to the same row, column or diagonal of the image, then one can prove [2] that $\pi_{ij} = 2/[|i-j|(|i-j|+1)]$, and (iii) for other pairs of nodes $\pi_{ij} = 0$. After some algebra, one has that, for two white noise images,

$$\sum_{ij} A_{ij} A'_{ij} = \mathcal{Q}_{\min} + \sum_{i=1}^{N-2} \sum_{j=i+2}^{N} \frac{4(N+1)}{[|i-j|(|i-j|+1)]^2} + \sum_{k=1}^{N-3} \sum_{i=1}^{N-k-2} \sum_{j=i+2}^{N-k} \frac{8}{[|i-j|(|i-j|+1)]^2}, \tag{2}$$

where the first term accounts for edges in the common lattice structure, the second term accounts for common edges across rows and columns, main diagonal and antidiagonal of the image and the third term accounts for common edges across off-diagonals and off-antidiagonals. The sum of the latter two terms in the expression is $\approx 1.32N^2$ and accordingly $\mathcal{Q}$ in Eq.1 saturates in this case to a non-null constant value $\mathcal{Q} \approx 0.165$, i.e. the range of $\mathcal{Q}$ is not between 0 and 1.

To account for this offset, we can then build a similarity function as it follows:

$$\mathcal{S} = \frac{\mathcal{Q} - \mathcal{Q}_{\infty}}{1 - \mathcal{Q}_{\infty}}, \tag{3}$$

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2019.2891742, IEEE Transactions on Pattern Analysis and Machine Intelligence

where $\mathcal{Q}_\infty$ is the raw similarity measure between the image visibility graph of an image and the same image extremely polluted with white noise ($\text{NSR} = 100$), e.g. $\mathcal{Q}_\infty = 0.165$ if we were just comparing two white noise images. By construction, $\mathcal{S}$ is generically equal to one for equal images and reaches zero when the image is polluted with noise with a $\text{NSR} = 100$.

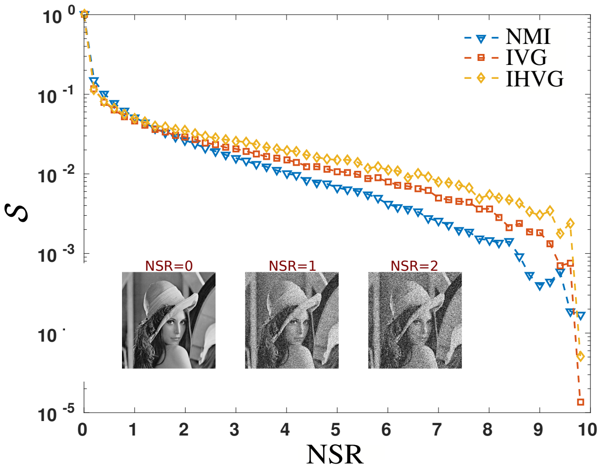

FIG. 1: Semi-log plot of Similarity measure between the IVG/IHVG of Lena and Lena polluted with a certain amount of Gaussian white noise, as a function of the Noise-to-Signal Ratio (NSR). The benchmark similarity measure computed directly on images is based on an appropriately rescaled version of the Normalized Mutual Information. Information mapped to the IVG/IHVG setting seems to be more robust to noise contamination than the raw information stored in the image.

As an application, we consider the standard grayscale Lena image. In Fig.1 we plot in semi-log scale $\mathcal{S}$ (computed in both IVG and IHVG) as a function of the NSR. For comparison, we also examine the robustness against noise *directly on the images*. In this case similarity is measured using Eq.3 where instead of the link overlap $\mathcal{Q}$ we use the Normalized Mutual Information (NMI) and apply the same definition than above to build a NMI_∞. Results suggest that information extracted in IVG/IHVG is more robust to noise contamination than the raw information present in the image when substantial amounts of noise are added. This result is on agreement with previous works that showcase equivalent noise robustness results in VG/HVGs [2, 11] and demonstrate that IVG/IHVGs encapsulate information which is robust against noise contamination. This virtue opens the possibility of using IVG/IHVGs to handle real images, which by construction are always polluted with noise.

C. Feature extraction

Here we define relevant graph features which can be extracted from IVG/IHVG and that will be of use to describe and characterise real images.

1. *Topological plots*

One of the simplest graph properties [4] is that related to local properties of nodes. For instance, the degree k_i of node i in an undirected graph is defined as the number of links incident to i, while the local clustering coefficient c_i of node i is defined as the percentage of nodes which are adjacent to i which are also adjacent to each other. One can assign labels to the nodes of a graph and can accordingly define the degree sequence, clustering sequence, etc of a given graph.

In the realm of VG/HVGs nodes have a pre-specified (temporal) ordering, and therefore these sequences inherit such order: the degree sequence $(k_1, k_2, \dots)$ is such that k_i is the degree of the node associated to datum x_i. These considerations can be extended to the image setting considered here, as similar labels can be attached to the nodes of an IVG/IHVG. By construction however, it is natural to order the node set of an IVG/IHVG in terms of an array (matrix) which represents the spatial positions of the associated pixels in the image. In this sense, one can therefore define the *degree plot*, the *node clustering plot*, etc, as primary topological properties of the IVG/IHVGs. For illustration, in Figure 2 we depict the process of extracting the degree plot of the IHVG associated to a 5×5 grayscale image. In the degree plot (also called *k-filtered image*, see section III) the pixel values $\{k_{ij}\}$ correspond to the degree of the nodes ij in the IHVG.

In section III we will use the topological plots for image filtering. Before that, we need to introduce a few other features which will be relevant for image classification.

2. *Global features: Degree distribution*

Assuming the premise that the IVG/IHVG inherits information of the associated image, a natural procedure is to extract features from these graphs and use them for image classification purposes. Now, graph properties can be classified into global properties (accounting for topological information of the whole graph), or local properties (which account for properties of small substructures). Global features include the degree distribution $P(k)$, defined as the percentage of nodes that have degree k, and other properties of the degree sequence, clustering distribution, spectral properties, distribution of cycles, etc.

In the context of visibility graphs, a recent theorem [9] proved that under suitable conditions, HVGs are in bijection with their degree sequence, suggesting that the degree distribution (marginal of the degree sequence) is

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2019.2891742, IEEE Transactions on Pattern Analysis and Machine Intelligence

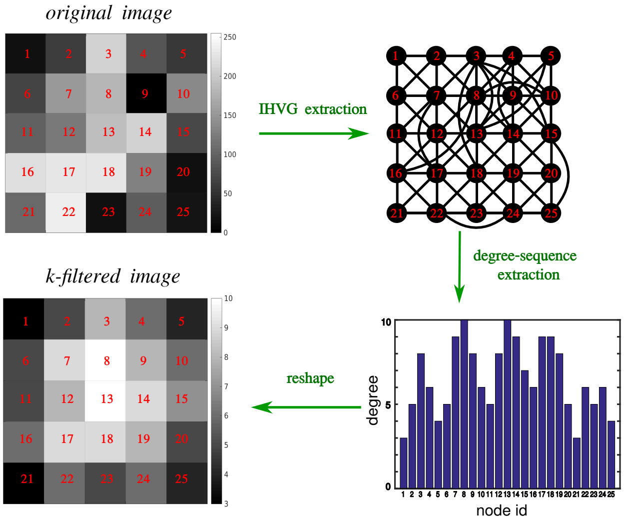

FIG. 2: Illustration of the process of extracting a topological plot from an image. Here we use IHVG for illustration, but a similar process applies to IVG. The image is initially mapped into its corresponding IVG or IHVG (here an IHVG) where the nodes properties inherit the spatial local information of the pixels according to the definition of an IVG or an IHVG. Then, for each node a specific topological property is measured –for example its degree– and the measured values are mapped back to form a matrix of the same size of the original image.

indeed a relevant and informative global property. Additionally, it has been shown that $P(k)$ can be computed exactly for simple images and is a good feature to distinguish different sorts of spatio-temporal dynamical systems [31]. Justified by these previous findings and for the sake of parsimony we will focus hereafter on $P(k)$ as the global feature under study.

3. *Local features: Visibility Patches*

Opposed to global features, we might be interested in studying local properties of the image visibility graphs, and in this subsection we introduce a set of novel features which aims at detecting such properties. The procedure consists in counting repetitions of small subgraphs in the IVG/IHVG associated to a given image. We call these subgraphs *Visibility Patches* (VPs) and we formally define them in the following way:

Definition (Visibility Patch, VP$_p$) *Consider an IVG of N^2 nodes associated to an $N \times N$ real-valued image matrix. A Visibility Patch of order p, VP$_p$, is any subgraph of the IVG formed by a set of p^2 nodes $\{ij\}_{i=s,j=s'}^{s+p-1,s'+p-1}$ for arbitrary s, s' that satisfy $1 \leq s, s' \leq N-p$.*

Note that the definition above extends naturally to

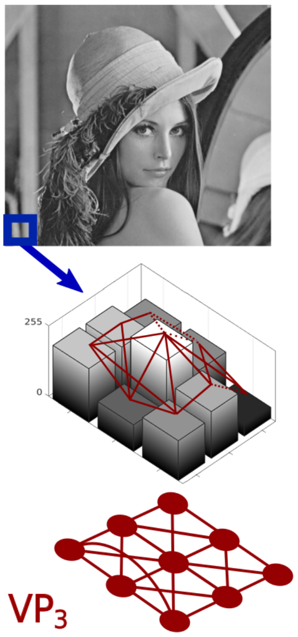

FIG. 3: Illustration of the process of measuring a Visibility Patch of order $p = 3$ (in this illustration we use VP$_3$, but HVP$_3$ can be extracted equivalently by using the horizontal criterion instead, see the text for definitions).

Horizontal Visibility Patches of order p (HVP$_p$) if the graph under study is IHVG instead of IVG. See Figure 3 for an illustration of how a particular visibility patch is extracted. The lowest order that yields nontrivial visibility patches is $p = 3$. These visibility patches can be detected by sliding a cell of size 3×3 pixels with stride 1 along the entire image, and extracting the corresponding IVG/IHVGs within the cell, as illustrated in Figure 3. For the sake of exposition we will focus on $p = 3$ in this work, but note that algorithms are scalable to higher orders if higher performance is needed.

Interestingly, note that the concept of visibility patch is the natural extension to images of the concept of sequential visibility graph motifs [11, 36]. As a matter of fact, algorithmic computation of visibility patches reduces to checking for the presence or absence of certain combination of visibility graph motifs. This fact enables mathematical tractability of Visibility Patches as well as the design of a linear time algorithm to extract visibility patches, as we now show below.

Enumeration. By construction, an (Horizontal) Visibility Patch of order p, (H)VP$_p$, is composed by an ordered sequence of $2p + 2$ sequential visibility graph motifs of order p [11, 36] which populates these motifs along the p rows, p columns and the main diagonal and anti-diagonal of the patch, supplemented by an additional number $q = 4(p - 2)$ of motifs of lower order associated to the off-diagonals and off-anti-diagonals: 4 motifs of order $p - 1$, 4 motifs of order $p - 2$, and so on. Accord-

motif	label	shape	HVG motif detection criterion	VG motif detection criterion
M0	0		$\{\forall x_0, x_2 : x_1 > x_0\} \cup \{\forall x_0 : x_1 < x_0, x_2 < x_1\}$	$\{\forall x_0, x_1 : x_2 \leq 2x_1 - x_0\}$
M1	1		$\{\forall x_0 : x_1 < x_0, x_2 > x_1\}$	$\{\forall x_0, x_1 : x_2 > 2x_1 - x_0\}$

TABLE I: Sequential VG/HVG motifs of order 3. Each motif can be characterized according to a set of inequalities in the associated time series (see [11, 36] for details on extension to higher orders).

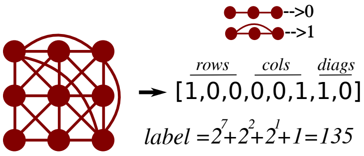

FIG. 4: Illustration of the labelling of a concrete HVP_3 by making use of the sequential visibility graph motifs of order 3 (see table I) and a binary expansion.

ingly, a natural enumeration of a given (H)VP_p is via a (2p+2+q)-dimensional vector $\mathbf{V}$ given by

$$\mathbf{V} = [r_1, r_2, \dots, r_p, c_1, c_2 \dots, c_p, d_1, d_2, \dots],$$

where each element of this vector enumerates the label of the actual visibility graph motif in the direction it represents. For example, in figure 4 we consider an example for the simpler nontrivial case $p = 3$. There are only two possible (horizontal) sequential visibility graph motifs of order 3 (see table I) –which we can label 0 and 1– and a single possibility for motifs of order two –and therefore these do not contribute to the combinatorics–. This yields a total of $2^{(2p+2)} = 256$ different possible patches, which can be enumerated by a $2p + 2 = 8$ dimensional vector $\mathbf{V} = (V_1, \dots, V_8)$. As there are just two non-trivial motifs of order 3, one can enumerate all possible patches of order 3 by interpreting $\mathbf{V}$ as the binary expansion of its label $\mathcal{L}$, i.e.

$$\mathcal{L}[\mathbf{V}] = \sum_{i=1}^{8} V_i 2^{8-i} = r_1 2^7 + r_2 2^6 + \cdots + d_1 2^1 + d_2 2^0 + 1.$$

For instance, in the particular example reported in Fig. 4, we find the motif label '0' in all rows, columns and diagonals except the first row, the last column and the main diagonal, i.e. $\mathcal{L}[\mathbf{V}] = 2^7 + 2^2 + 2^1 + 1 = 135$, i.e. corresponding to patch label no. 135.

Linear time recognition algorithm. A relevant quantity that summarises the abundance of visibility patches is given by the frequency distribution of each of these patches, labelled *(Horizontal) Visibility Patch Profile* $\mathbf{Z}$ (we label this profile with the bold letter $\mathbf{Z}$ indistinctively for patches computed on IHVG and IVG, and context will suggest which one of the two it applied to). In particular, for order $p = 3$ there are a total of 256 possible patches so $\mathbf{Z} = (Z_1, \dots, Z_{256})$ where Z_j provides the relative frequency of patch j. By further capitalising on the relation between Visibility Patches and Visibility motifs, we can build an algorithm to estimate the Visibility Patch profile $\mathbf{Z}$ of a given image which scales linearly with the number of pixels of the image. In particular, recognition of motifs of any order p in a time series can be done in linear time by checking the inequality set in Table I (see [11, 36] for an explicit inequality set for $p = 4$ and additional details). For an illustration, consider the concrete case of HVP_3. An algorithm to compute the patch profile $\mathbf{Z}$ proceeds by sliding a 3×3 cell over the original image, and for each cell the algorithm recognizes (labels) the patch associated to that cell by checking the inequality set for the 3 rows, 3 columns and two diagonals of the cell. Since labelling of each patch is performed in linear time, then extracting the whole patch profile $\mathbf{Z}$ scales linearly with the number of pixels. A more concrete algorithm PatchProfile() is displayed below (the algorithm considers computation of HVP_3, but VP_3 would follow equivalently if we change the condition in the function $\text{M1}(x_0, x_1, x_2)$ according to table I).

Algorithm 1 PatchProfile()

Input: image $\mathcal{I}$
Output: profile $\mathbf{Z}$ of horizontal visibility patches of order 3

```
for $k_1, k_2 = 1$ to $N - 3$ do
    $i \leftarrow k_1$
    $j \leftarrow k_2$
    $\mathbf{V} \leftarrow \mathbf{0}$
    if $\text{M1}(\mathcal{I}_{i,j}, \mathcal{I}_{i,j+1}, \mathcal{I}_{i,j+2})$ then $V(1) \leftarrow 1$
    if $\text{M1}(\mathcal{I}_{i+1,j}, \mathcal{I}_{i+1,j+1}, \mathcal{I}_{i+1,j+2})$ then $V(2) \leftarrow 1$
    if $\text{M1}(\mathcal{I}_{i+2,j}, \mathcal{I}_{i+2,j+1}, \mathcal{I}_{i+2,j+2})$ then $V(3) \leftarrow 1$
    if $\text{M1}(\mathcal{I}_{i,j}, \mathcal{I}_{i+1,j}, \mathcal{I}_{i+2,j})$ then $V(4) \leftarrow 1$
    if $\text{M1}(\mathcal{I}_{i,j+1}, \mathcal{I}_{i+1,j+1}, \mathcal{I}_{i+2,j+1})$ then $V(5) \leftarrow 1$
    if $\text{M1}(\mathcal{I}_{i,j+2}, \mathcal{I}_{i+1,j+2}, \mathcal{I}_{i+2,j+2})$ then $V(6) \leftarrow 1$
    if $\text{M1}(\mathcal{I}_{i,j}, \mathcal{I}_{i+1,j+1}, \mathcal{I}_{i+2,j+2})$ then $V(7) \leftarrow 1$
    if $\text{M1}(\mathcal{I}_{i,j+2}, \mathcal{I}_{i+1,j+1}, \mathcal{I}_{i+2,j})$ then $V(8) \leftarrow 1$
    $\mathcal{L} \leftarrow 1 + \sum_{i=1}^{8} V(i) 2^{8-i}$
    $Z(\mathcal{L})$ ++
return $\mathbf{Z}$
```

```
function $\text{M1}(x_0, x_1, x_2)$
    M1 ← .FALSE.
    if $(x_1 < x_0) \wedge (x_2 > x_1)$ then M1 ← .TRUE.
    return M1
```

In Fig. 5 we plot how runtime scales as a function of the size of the image (in number of pixels), for randomly-generated grayscale images $I_{ij} \sim \mathcal{U}\{0, \dots, 255\}$, showcasing a linear time complexity and therefore an efficient feature extraction.

Theoretical analysis of visibility patches, including analytical results for the visibility patch profiles, is in principle possible and will be published elsewhere [37]. Here we are interested in exploring the performance of these features in practical situations. In section IV we will explore the performance of global features (e.g. degree distribution) and local features (patch profiles of order 3, as computed from (H)VP$_3$, these being the simplest nontrivial patches) in different image classification tasks.

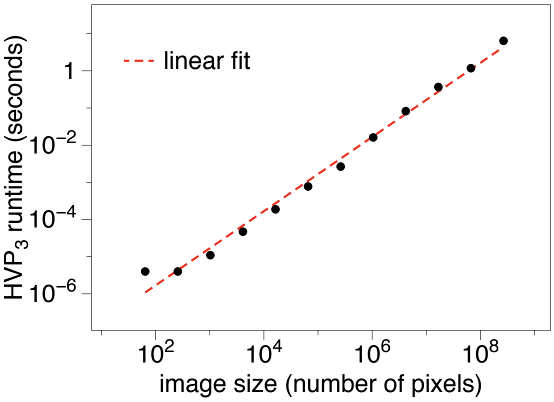

FIG. 5: Running time of the Horizontal Visibility Patch Profile algorithm of order 3 as a function of the size of the image. The input of the algorithm is the image and the output is the patch profile **Z** which records the frequency distribution of each patch. Feature extraction scales linearly (computation performed on a 2.5GHz IntelCore i7 processor).

4. Multiplex features

To round off this section, we now consider the situation where one might wish to handle a set of images at once. Examples include (i) RGB images, where a given image is actually built by combining three channels (R, G and B), (ii) a given image filtered in different ways (e.g. a set of different topological plots, or image filters at different frequency bands, etc), (iii) a 'temporal' image comprised of a certain number of snapshots, etc. In every case, the image under study is actually a set of m images $\mathcal{I} = \{\mathcal{I}^{(1)}, \dots, \mathcal{I}^{(m)}\}$ (for an RGB image, $m = 3$). One might then be interested in extracting the image visibility graph of each sub-image separately, or construct the image visibility graph of the image set at once. In this latter case, this naturally leads to the construction of an *Image Multiplex Visibility Graph*, i.e. the image version of a multiplex visibility graph [3]:

Definition (Image Multiplex Visibility Graph) *Let* $\mathcal{I} = \{\mathcal{I}^{(1)}, \dots, \mathcal{I}^{(m)}\}$ *be an ordered sequence of* m $N \times N$ *grayscale images. The image multiplex visibility graph is a multiplex visibility graph of* m *layers, where layer* $\alpha = 1, \dots, m$ *encodes the image visibility graph of grayscale image* $\mathcal{I}^{(\alpha)}$.

Note that the resulting graph is a multiplex network, a specific type of multilayer network [6, 7] characterised by a unique set of nodes which is connected differently on several independent layers or networks (see [38] for another recent proposal for multilayer graph-based analysis of RGB images). This architecture is possible thanks to the natural alignment of every pixel in each sub-image ($\mathcal{I}_{ij}^{(\alpha)}$ aligned with $\mathcal{I}_{ij}^{(\alpha')}$ $\forall(\alpha, \alpha')$). As it is customary in multiplex networks [6, 7, 44], one can proceed to extract features from the image multiplex visibility graph of several kinds:

1. *Intra-layer descriptors*: these are any graph measures computed independently on each layer [4, 45] (e.g. global or local features such as the ones discussed above).
2. *Inter-layer descriptors*: these are in general measures of correlation or interdependence of the features between different layers; examples include the edge overlap [46], degree-based correlation measures [3, 47], and correlations between network mesoscale descriptors such as node clusters [48].
3. *Intrinsically multiplex descriptors*: these are features that can be extracted only by considering the overall multiplex architecture of the network [50, 56], for example multiplex clustering coefficients and multiplex motifs [54, 55], node multiplex centrality measures [52, 53], and multiplex communities [49, 51].

For the sake of parsimony, in this work we only explore the performance of efficiently combining simple intra-layer descriptors, whereas analysing the performance of inter-layer and intrinsically multiplex descriptors is left for future work. In order to efficiently combine information of intra-layer descriptors across layers, here we apply a method of dimensionality reduction such as Principal Component Analysis to project the concatenated feature vector of intra-layer descriptors into an adequate subspace which we call the *pseudo-multiplex descriptor subspace* (see Fig. 6 for an illustration). In section IV we will illustrate the use of image multiplex visibility graphs as a feature extraction method for texture classification.

III. FILTERING

A. Definitions

In this subsection we consider employing the topological plots of IVG/IHVGs as filters.

Definition (Visibility filter). *Let* $\mathcal{I}$ *be a* $N \times N$ *matrix where* $\mathcal{I}_{ij} \in \mathbb{R}$. *We define the visibility filtered image, or simply visibility filter* $\mathcal{F}_{VG}(\mathcal{I})$ *as a* $N \times N$ *matrix whose* ij *element corresponds to the value of a*

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2019.2891742, IEEE Transactions on Pattern Analysis and Machine Intelligence

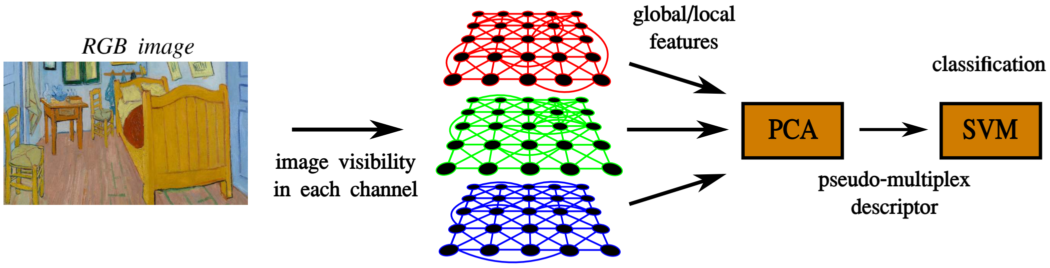

FIG. 6: Illustration of the process of extraction an image multiplex visibility graph (here applied to a RGB image, hence yielding a multiplex graph with $m = 3$ layers). Features extracted from each of the layers are combined using a Principal Component Analysis to obtain a pseudo-multiplex descriptor, that is fed into a classifier.

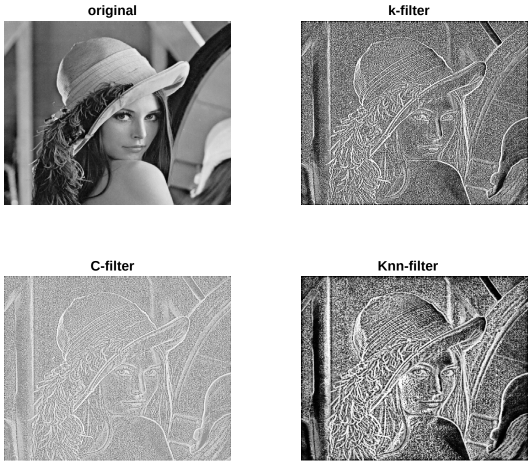

FIG. 7: Several Visibility Filters $\mathcal{F}_{\text{HVG}}$ applied on Lena (we have applied an histogram equalization to balance contrast on the filtered images).

specific node property of the node ij *of the associated IVG, such as its degree or its clustering coefficient. If the property is a real scalar quantity, then this quantity is opportunely rescaled to grayscale integer values. The horizontal visibility filter* $\mathcal{F}_{HVG}(\mathcal{I})$ *follows equivalently.*

The above definition of a visibility filter reduces to a particular topological plot with a trivial rescaling, which allows to compare across different node properties in comparable ranges (grayscale intensity range in $\{0, 255\}$). A priori one can use many different node properties to define such filters: here we focus on three of the simplest properties, namely degree, clustering and degree-degree [4] to define three filters $\mathcal{F}_{\text{HVG}}(\mathcal{I})$:

- k-filter: when we assign to ij the degree of node ij. No rescaling is necessary here.
- C-filter: when we assign to ij the local clustering coefficient (rescaled in $\{0, 255\}$) of node ij.
- Knn-filter: when we assign to ij the closest integer value from the average nearest-neighbors degree of node ij.

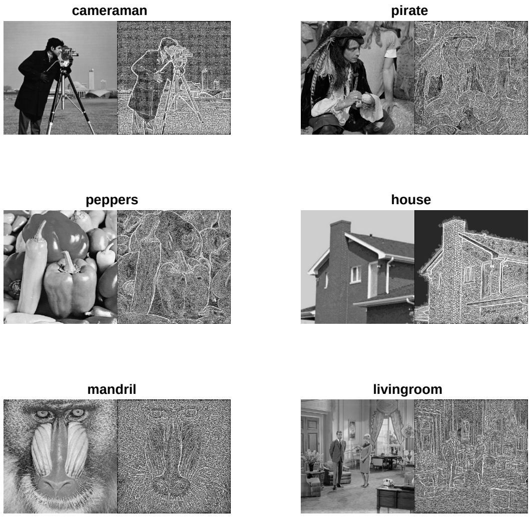

FIG. 8: Set of 512×512 standard grayscale images often found in literature along with their IHVG k-filter.

For illustration, in Fig. 7 we plot the original grayscale Lena image (512×512) together with three HVG filters (histogram equalization was applied to the k-filter and to the Knn-filter, and the C-filter is plotted in reverse grayscale to allow a better visualization of the details). Additional examples of k-filters (histogram equalized) associated to several 512×512 standard grayscale images[1] are displayed in Fig. 8.

A first observation when looking at both figures is that these filters share some similarities with classic edge detection filters (e.g. Sobol, Canny, entropy filters). However, at odds with edge detection filters, in our case additional fine-grain information of the image seems to be

---

[1] `http://imageprocessingplace.com`

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2019.2891742, IEEE Transactions on Pattern Analysis and Machine Intelligence

preserved. It's actually easy to notice that many important details (for example details in the Lena face and in the hat) are still recognizable, and more particularly the texture of the image is somehow preserved. This fact will be quantitatively assessed in §IV A where we make use of these filters to perform texture classification of certain materials.
In the next subsection we analyse with some detail the size of the filtered images, and will conclude that one can reach considerable size reductions in the filtered images while keeping most of the information.

B. Size Reduction

In grayscale, a generic squared image $\mathcal{I}$ has N^2 pixels and each pixel can take 2^8 values, so the image size is $|\mathcal{I}| = N^2$ bytes. In the IHVG k-filter, each pixel corresponds now to the degree k of the associated node. We know that by construction in IHVGs, neglecting boundary effects, we have $8 \leq k \leq N^2 - 1$. Now, a recently proved theorem [31] guarantees that for white noise images, the degree distribution of the IHVG decays exponentially, in such a way that the probability of finding a degree larger than a certain κ is

$$P(k > \kappa) = \sum_{k=\kappa}^{\infty} \left(\frac{1}{9}\right)\left(\frac{8}{9}\right)^{k-8} = \left(\frac{8}{9}\right)^{\kappa-8}$$

To give an idea of the order of magnitude, for $\kappa \approx 24$, this probability is roughly 0.15, meaning that the expected number of nodes having degree larger than 24 is around 15%. In other words, if we decided to cut-off information of the degree and assign $k = 24$ for all nodes whose degree is larger or equal to 24, effectively we would only *saturate* 15% of the pixels in the k-filter of a white noise image. In general, for a white noise image the percentage of *saturated pixels* would simply be

$$\mathrm{SP}(\kappa) = 100\left(\frac{8}{9}\right)^{\kappa-8}$$

Now, what would be the effect of this saturation on the size of the filtered image? If no cut-off is set on the degree there is an exponentially vanishing (although not zero) probability of finding a node with degree k, for any $k < N^2 - 1$. In practice, by setting a cut-off value $\kappa < N^2 - 1$, by construction, the size of the filtered image would be

$$|\mathcal{F}_{\mathrm{HVG}}(\mathcal{I})| = \frac{\log_2(\kappa - 8)}{8} N^2 \text{ bytes.}$$

Note that for $\kappa = 24$ we only need $N^2/2$ bytes, so we have a factor 2 size reduction with minimal degradation. Defining the *reduction factor* $\mathrm{RF}(\kappa)$ as

$$\mathrm{RF}(\kappa) = \frac{|\mathcal{I}|}{|\mathcal{F}_{\mathrm{HVG}}(\mathcal{I})|} = \frac{8}{\log_2(\kappa - 8)},$$

our theoretical analysis predicts an analytical relation for the RF-SP plane in the case of white noise images:

$$\mathrm{RF} = \frac{8}{\log_2(\log(\mathrm{SP}/100)/\log(8/9))} \tag{4}$$

Now, at least in the 1D (time series) version it is known that the degree distribution of HVGs also decays exponentially in the *typical* case (not just for white noise), so we expect that a similar behaviour takes place in the image setting, and thus the probability of finding large degrees is, generically, exponentially small. In Fig.9 we plot the RF-SP plane for the theoretical curve along with experimental results on the k-filtered Lena. For comparison, a similar saturating procedure is applied to the original Lena image, and results are also shown. We can see that the filtered image allows for a larger size reduction than the original image, and that even for large values of saturating percentage one can clearly identify Lena in the saturated filtered image.

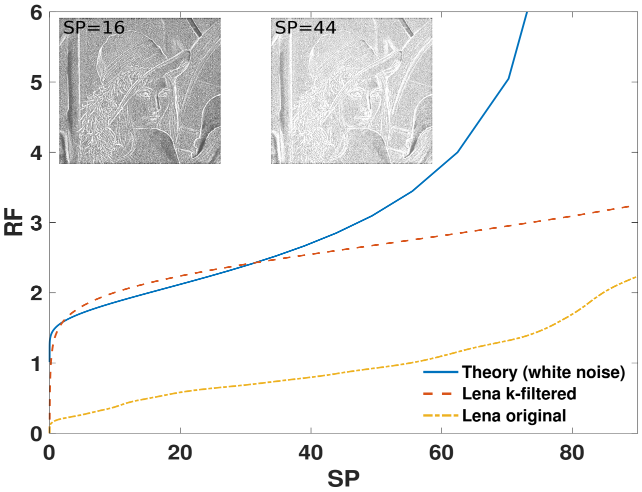

FIG. 9: Reduction factor (RF) in function of the percentage of saturated pixels (SP) obtained with the k-filter on a white noise image (solid blue line) and on Lena (brown dashed line). The curve RF-SP is also shown for the case of Lena without filter (dotted yellow line) for comparison.

C. IHVG filters for image pre-processing: comparison

Consider a standard problem in image processing: visual face recognition. The canonical approach in feature based classification consists on extracting from each image a feature vector and subsequently feed classifiers with these vectors, for supervised learning. In some cases the performance of the classifier can be improved if images are pre-processed (i.e. before feature extraction) [69]. For example local binary patterns (LBP) [65–67] that are robust to monotonic grayscale variations are widely used in face recognition problems to overcome the

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2019.2891742, IEEE Transactions on Pattern Analysis and Machine Intelligence

fact that often images are taken in different illumination conditions or with different resolutions.

Here we test the potential of the different IVG/IHVG filters defined above as pre-processing filters for the improvement of image texture analysis [69]. To simulate real-world acquisition variability we transform an image of Lena to different resolution scales by applying to the original image a 5×5 Gaussian kernel with increasing values of standard deviation. As a standard feature extraction method we then make use of the gray level co-occurrence matrix (GLCM) [35, 69], a well known texture analysis technique. For a given resolution (kernel std value) the GLCM is computed from the image and four so-called Haralick features are extracted: Contrast (measuring local variations), Correlation (measuring joint probability occurrence of pairs), Energy (sum of squared elements of the matrix) and Homogeneity (measuring the closeness of the distribution of elements in the matrix to its diagonal), see [35] for details. All images are then mapped into points in the 4-dimensional space spanned by the appropriately normalized Haralick features (see Fig. 11). The distance $\mathcal{D}$ between low resolution images and the original one is subsequently computed (here we use city-block (L_1) norm, results hold similar for an Euclidean metric) . Intuitively, the lower $\mathcal{D}$, the more similar a low-resolution image is from its high-resolution version in feature space, and accordingly a classifier trained with the high-resolution version would a priori find less difficulties to correctly classify low-resolution images for the filter with overall lower values of $\mathcal{D}$.

The procedure is repeated several times for (i) the non-filtered image, and for (ii) filtered images using all six visibility filters (k-filter, $Knn-$filter, and $C-$filter for IVG and IHVG). A comparison with the classic LBP and Texton [70, 71] filters is reported as well for completeness. To obtain the Texton a filter bank of 11 filters (Gaussian at sigma scales 1, 2, 4; Laplacian of Gaussian at sigma scales 1, 2, 4, 8; horizontal and vertical Derivative of Gaussian at sigma scales 2 and 4; filtering window of size 5x5 pixels) was implemented and a k-means algorithm (5 clusters) was used to combine everything in a final filter. Results are shown in Fig. 10 b), where we plot in log-log $\mathcal{D}+1$ as a function of the kernel standard deviation (i.e. the larger this parameter, the lower the image's resolution). In panel a) of the same Figure, the filter effect for the different filters used in the analysis on the Lena face is shown for illustration. We find that in the case of both IVG and IHVG Knn-filters, the IVG k-filter and the Texton filter, $\mathcal{D}$ appears systematically below the one for the non-filtered image, suggesting that all low-resolution images will be systematically closer (in feature space) to the high-resolution version after these specific filtering pre-processing.

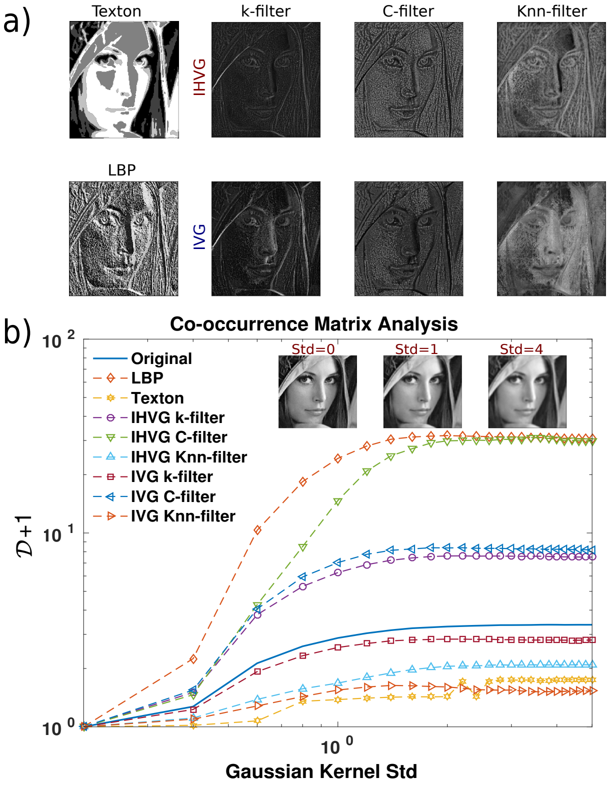

FIG. 10: Performance of the IVG/IHVG filters in image pre-processing. a) The IVG and IHVG filters are shown together with the LBP (local binary patterns) filter and the Texton filter as applied to the Lena face image. b) Distance $\mathcal{D}$ between the Lena face image and the same image at different resolution scales is measured in the 4-dimensional space of the (normalized) Haralick features, and plotted as a function of the standard deviation of the Gaussian 5x5 kernel applied to obtain low-resolution images. $\mathcal{D}$ is computed for the original unfiltered image as well as for the same image after being pre-processed using the filters shown in a). $\mathcal{D}$ is an indicator of the performance of the filters as pre-processors (the lowest the better, see the text).

IV. PATTERN RECOGNITION AND TEXTURE CLASSIFICATION

Once we have discussed the usefulness of image visibility graphs for filtering purposes, we now turn and explore their applicability as efficient and universal feature-extraction methods for the task of texture detection and classification, a classical problem in computer vision [57, 58].

Despite no rigorous mathematical definition of texture is widely accepted, textures can be intuitively defined as characteristic visual patterns arising from the spatial distribution of the pixel intensity values in images [38], and therefore they are of fundamental importance in many machine vision applications such as object tracking, face recognition and automated medical diagnosis [60]. Following recent efforts in the direction of cataloguing textures and texture-related datasets

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2019.2891742, IEEE Transactions on Pattern Analysis and Machine Intelligence

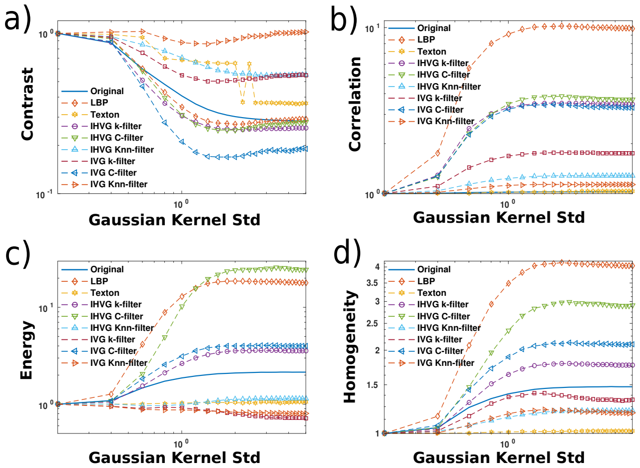

FIG. 11: Haralick features Contrast, Correlation, Energy and Homogeneity extracted from the Lena image when Gaussian kernels of different variance are applied to it and different IHVG/IVH filters are used to pre-process the image before the extraction. The features are all normalized (rescaled) with respect to the value of the original image (Gaussian kernel std=0).

[59, 60], we will test the performance of image visibility related descriptors on different types of textures, namely textures of materials, bio-medical textures and natural textures.

A. Global features for Materials Texture Recognition

We start by showing that the degree distribution $P(k)$ extracted from either IVG or IHVG is an ideal *global feature* vector for the automatic recognition of *material textures*. Incidentally, let us observe that the degree distribution $P(k)$ of an IVG or IHVG associated to a given image is mathematically equivalent to the pixel intensity histogram of the k-filter of the image, therefore using $P(k)$ for classification purposes is equivalent to using basic properties of the k-filter (intensity histogram) for the same purpose. Actually, in §III A we claimed that visibility filters do indeed capture fine-grained information about the image, and in this section we justify this claim quantitatively. We also compare the performance of the classification task with results obtained from the LBP and the Texton filters (a sorted histogram of pixel intensities was used as the feature vector in these cases).

Material textures are a specific class of textures derived by imaging the surface of various types of materials for the purpose of classifying them according to the characteristic structural patterns captured in this way. The dataset we used is a reduced version of the Kylberg Texture Dataset [39], formed by grayscale texture images from 6 different materials (canvas surface, cushion surface, line seeds, sand, seat surface and stone surface) with 40 samples per class. This dataset is often used as a benchmark for testing material texture descriptors because of the image homogeneity in terms of perspective, scale, and illumination across the different classes [40, 41]. Because of the grayscale nature of the images and the uniform texture of the materials, this is ideal to test the capability of IVG/IHVGs global features to capture the surface structural correlations and patterns.

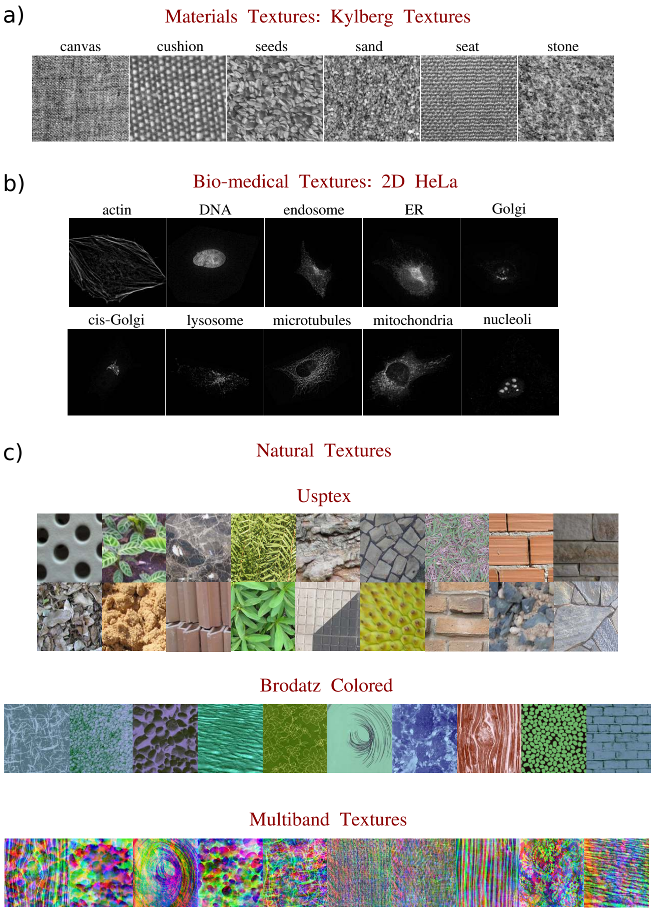

FIG. 12: (Color online). Datasets used to validate the usefulness of image visibility features in texture recognition: a) the Kylbert reduced dataset includes grayscale images of texture from six different materials. b) 2D HeLa contains grayscale grayscale fluorescence microscopy images of ten organelle-specific proteins. c) Three dataset are used for the task of natural texture classification: the Upstex dataset, the Brodatz Colored dataset, and the Multiband dataset (MTB). In the Upstex dataset each image is a sample from of a class (191 classes in total, 12 images per class), while for Brodatz and MTB each image shown include all the images of a class and its 16 non-overlapping sub-images represent the samples (151 classes in Brodatz, 154 classes in MTB).

In Figure 12 a) we provide an illustration of the tex-

	Complex Tree	Linear Discriminant	linear SVM	quadratic SVM	weighted KNN	Ensemble Bagged Tree
IHVG P(k)	**100%**	96.7%	**100%**	**100%**	98.8%	**100%**
IHVG P(k)+PCA (5 PCs)	**100%**	96.7%	**100%**	**100%**	**100%**	99.2%
IVG P(k)	95.3%	95.8%	99.2%	**100%**	94.2%	98.8%
IVG P(k)+PCA (5 PCs)	96.7%	97%	97.9%	**100%**	**100%**	98.8%
LBP	95%	98.8%	**100%**	**100%**	**100%**	**100%**
LBP+PCA (5 PCs)	97.1%	99.2%	**100%**	**100%**	**100%**	98.3%
Texton	79.6%	74.2%	84.2%	93.3%	89.2%	86.2%

TABLE II: Kylberg Texture Dataset: performance in classification accuracy obtained with different algorithms. The classifiers used are standard classifiers implemented in the MATLAB Classification Learner App. The degree distribution $P(k)$ can be understood as the intensity histogram of the k-filter and, for comparison, we include results obtained by extracting the sorted histogram of pixel intensities of the standard LBP and Texton filters.

ture classes. Each image in the dataset is transformed to the corresponding IHVG and its degree distribution $P(k)$ is extracted and then given in input (as a vector of features) to different classifiers. Principal component analysis (PCA) is applied to the input vectors to reduce the number of features and to avoid eventual overfitting, and a 5−fold cross validation procedure is adopted. A 100% (average by class) classification accuracy –defined as the percentage of samples correctly assigned to their true class– can be obtained by mean of a linear/quadratic-kernel Multiclass one-vs-one Support Vector Machine (ovo-lSVM/qSVM), for IHVG and with a quadratic-kernel SVM for IVG. Interestingly, this level of accuracy is already obtained only using the first 5 principal components of the PCA projection. To further understand why the degree distribution is such an informative feature, in Fig. 13 we plot the degree distribution for all the samples (colored by class as a guide for the eye) for the IHVGs (a) and for the IVGs (b). Distributions cluster by class and already by visual inspection we can determine the clusters. In Table II we report the classification accuracies reached by different standard classifiers implemented in the MATLAB Classification Learner App. For IHVGs it is possible to obtain an accuracy of 100% by using a Complex Tree classifier or a Bagged Tree and with a weighted KNN algorithm as well after applying PCA. For completeness, we also report results for the same classification task using standard filters (LBP and Texton) instead of our k-filter. Results are equally good for LBP and the performance drops for Texton.

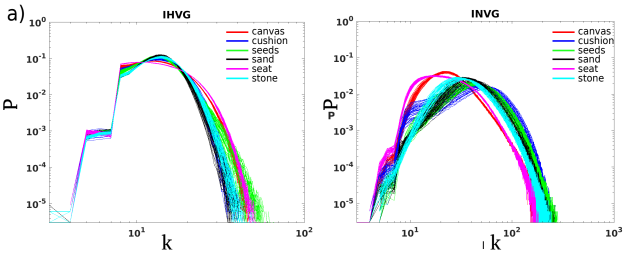

FIG. 13: (color online) Log-log plot of the degree distributions $P(k)$ for a) all IHVGs and for b) all IVGs extracted from the 6 classes of the Kylberg Texture Dataset. This global features appears to be capturing a great deal of the differences between textures, reaching a 100% classification accuracy using a linear Support Vector Machine (see table II).

B. Combining Global and Local Features for Bio-medical Texture Classification

In a second step, we now focus on a more complex image dataset called 2D HeLa. This is a bio-medical dataset of grayscale fluorescence microscopy images of HeLa cells stained with various organelle-specific fluorescent dyes able to reveal characteristic structural patterns of proteins so that 10 classes can be identified: Actin, DNA, Endosome, ER, cis-Golgi, Golgi, Lysosome, Microtubules, Mitochondria and Nucleoli. For each of these classes there is a different number of sample images (between 73 and 98), for a total of 862 images. In Fig. 12 (b) we illustrate one representative sample for each class.This dataset is exemplar with respect to the bio-medical challenge of the automated interpretation of sub-cellular protein patterns in fluorescence microscope images, which can help characterize many genes whose function is still unknown.

Unlike material textures, these images show several complex biological structures that occupy only a certain subregion of the image. This fact precludes using only global features, justifying the necessity to complement these with local features to attain high classification accuracies. Accordingly, we test the performance of the Visibility Patches.

Again, we feed the feature vector with global descriptors given by the first entries of the degree distribution $P(k)$ of IHVGs (up to $k = 40$), the degree distribution of IVGs (up to $k = 80$), and a concatenation of both. Additionally, we add local features given by the visibility patch profiles $\mathbf{Z}$ of HVP$_3$ (188 patches detected) and VP$_3$ (249 patches detected), and a concatenation of the two. To avoid overfitting, PCA is applied to reduce the number of features before classification. The PCA-transformed features are then given in input again to a linear-kernel Multiclass one-vs-one Support Vector Machine. The best classification accuracy is obtained by using a combination of global and local features from

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2019.2891742, IEEE Transactions on Pattern Analysis and Machine Intelligence

the blend of IHVG and IVG descriptors, reaching 83.5%.

In Table III we report, for each set of descriptors, the values of the best average classification accuracy and corresponding average (over the classes) area under the ROC curve (AUC) obtained over 30 realizations of the classifier, together with the average classification accuracy over of the model realizations (Model Average Accuracy MAA). In Fig. 14 we depict the details of the confusion matrix and of the ROC curves by class for the best model trained. The highest off-diagonal values in the confusion matrix correspond to pairs of classes that are expected to be hard to distinguish, for example ER and mitochondrial proteins and Golgi and cis-Golgi proteins. The optimal number of PCs for each feature set (local/global, IVG/IHVG) is found with an exhaustive search in the space of the principal components number $\mathbf{N_{PCs}} \in [5, ..., 25]$ by measuring the average classification accuracy of the model (MAA) over 30 model realizations (see Figure 15 (a) and Table III). The values of MAA for each descriptor tend to quickly saturate by increasing the number of components, meaning that the final model is robust with respect to the choice of the set of $\mathbf{N_{PCs}}$). Interestingly, we observe that while global features from the IHVG are more informative than for the IVG, the opposite is true for local features, as these seem to be more informative for IVG than for IHVG. For comparison, in the original paper [42] of the 2D HeLa dataset the authors obtained an average class accuracy of 84% by training a Back-Propagation Neural Network with specifically designed (ad-hoc) features. In other words, our parsimonious approach, using a set of non-specific graph-based features is competitive with respect to ad-hoc, context-dependent features. The highest published performance we are aware of is 95.3% [43], obtained with a multi-resolution feature extraction approach combined with a neural network classifier, and it is an open problem whether one could reach competitive results by complementing the non-specific approach designed here with problem-specific strategies.

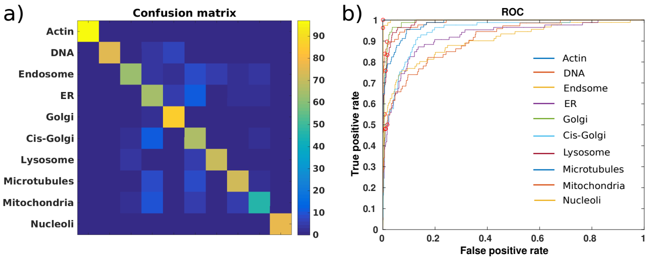

FIG. 14: Confusion matrix (panel a) and AUCs (panel b) for the classification of the 2D HeLa Dataset, using a quadratic kernel Support Vector Machine on a blend of Global (degree distribution) and Local (Visibility Patches) from IHVG and IVG.

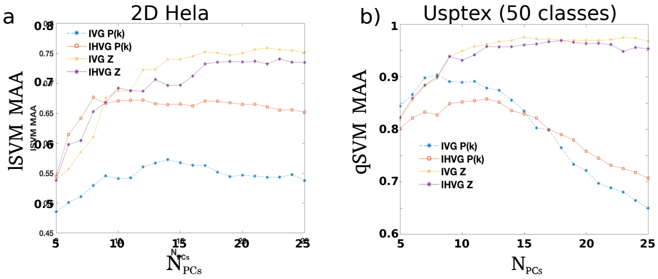

FIG. 15: Model average classification accuracy (30 model realizations) in function of the number of principal components $\mathbf{N_{PCs}}$ obtained with basic global/local descriptors using a) a linear-kernel Multiclass one-vs-one Support Vector Machine on the 2DHela dataset and b) a quadratic-kernel Multiclass one-vs-all Support Vector Machine on a subset of Usptext where only the first 50 classes are included.

ovo-lSVM models	Accuracy	⟨AUC⟩	MAA
Global features: P(k)+PCA IVG+IHVG	75.4%	0.939	74.5%
Local features: **Z** + PCA IVG+IHVG	79.5%	0.961	78.6%
Global +*Local* (67 features) IVG+IHVG	**83.5%**	**0.973**	**83%**
2D Hela descriptors		$\mathbf{N_{PCs}}$	MAA
P(k) IVG		14	57.2%
P(k) IHVG		8	67.6%
Z IVG		22	75.8%
Z IHVG		23	74%

TABLE III: 2D HeLa dataset. (Top) Best classification accuracy, best average by class AUC and model average classification accuracy MAA obtained with linear-kernel Multiclass one-vs-one Support Vector Machine (ovo-lSVM) classifier algorithm over 30 models realizations, for different sets of image visibility features. The best results are found when using a blend of global (degree distribution) and local features (visibility patches). (Bottom) The number of principal components $\mathbf{N_{PCs}}$ for each descriptor is tuned via exhaustive search in the space $\mathbf{N_{PCs}} \in [5, ..., 25]$ by maximizing the model average classification accuracy MMA over 30 model realizations.

C. Image Visibility Multiplex for RGB Images: Natural Textures

In natural scenes, more complicated textures can arise, e.g. patterns of waves on the surface of a lake or conformation and disposition of leaves from a plant when pictured from above. These kind of textures are generally classified as *natural textures* and are very relevant because they can be seen as the building blocks of more complex features arising in real world scenes.

Accordingly, in this subsection we consider three well known collections of natural textures: the Upstex dataset [61], the Colored Brodatz dataset [63], and the Multiband (MTB) dataset [63]. The Upstex dataset includes 191 classes of 12 samples each showing textures from outdoor scenes such as walls, pavements,

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2019.2891742, IEEE Transactions on Pattern Analysis and Machine Intelligence

vegetation, soil, plant leaves, rocks (in Figure 12 c) we show some samples from different classes). The Brodatz dataset in turn is among the most widely used datasets in the literature of texture analysis and comprehend scanned images from an album. In the colored version [63] of the dataset there are 112 images of size 640×640, defining a total of 112 different classes (in Figure 12 c) we report some examples). For each class there are 16 samples defined as the non-overlapping subimages of size 160×160. Finally, the Multiband dataset is derived from the Brodatz dataset by aggregating triplets of original grayscale images in various combinations such that the resulting RGB images display a mixture of channel-dependent textures (see some examples in Figure 12 c)), totalling 154 classes and 16 samples per class.

All these three datasets are characterised by containing RGB images, many different classes, and a relatively small number of samples per class. Our approach here is to extract from each image the multiplex IVG/IHVG (by extracting the single-layer IVGs/IHVGs form each of the image channels: R,G,B) and obtain pseudo-multiplex descriptors by blending global/local features of each channel via PCA (see Figure 6). To obtain the highest classification performance we extracted global and local pseudo-multiplex features from both the multiplex IHVG and the multiplex IVG and we used a quadratic-kernel Multiclass one-vs-all Support Vector Machine (ova-qSVM), 10-fold cross validation.
In order to optimize the number of components for each of the multiplex local/global features we follow a methodology analogous to the one used for the 2DHela dataset, but this time, to prove the efficiency of our classifier we tuned the number of principal components of the features only once for all the datasets by optimising the average classification accuracy of the model (MMA) over 30 model realizations measured on the sub-set of data of the Usptex dataset which includes only the first 50 classes (Figure 15 (b)). In Table IV we report for each of the datasets the values of accuracy, average AUC per class and MAA obtained with the model trained with global, local and global+local multiplex features.

Our best classification accuracy for Usptex, Brodatz and Multiband is respectively 98.4%, 99.8%, 99.6% and is obtained with only 53 features that are a combination of multiplex local and global features from IHVG and IVG. Interestingly, in a very recent paper [38] the authors evaluated the performance of several pre-trained state-of-the-art convolutional neural network to two of these dataset (Usptex and Multiband), blended together with the performance obtained by extracting features from a so-called Multilayer Complex Network (Multilayer CN) in a similar spirit with respect to our multiplex approach. They report that 99.8% as the best accuracy ever published for Usptex, that they obtain with a pre-trained ResNet50 2016, and 97.1% the best accuracy for Multiband obtained with their Multilayer Complex Network classifier. To the best of our knowledge we obtained the highest performance ever on the Multiband dataset, which suggests that our multiplex approach is ideal for detecting multi-band texture features and outperforms traditional CNN feedforward architectures, including the Multilayer CN proposed in [38], that perhaps drop in performance because they are designed to learn texture features both from intra-band and inter-band pixel correlations.
Regarding the Brodatz colored dataset, we couldn't find any reference value of performance reported in literature, however for the grayscale original dataset a classification accuracy of 99.2% was recently obtained via an hybrid classifier which combines a Convolutional Neural Network and a Support Vector Machine [64].

V. CONCLUSION

The family of image visibility graphs IVG/IHVGs extract graphs from images and extend the concept of visibility graph time series analysis to image processing. In this paper we have showcased that this mapping can be leveraged to make image filtering and classification by extracting simple, universal (non-specific) and computationally efficient (i.e. fast and scalable extraction) features.

In a first step, we have shown how these graphs can be used to define filters for image preprocessing, which show good performance and are based on a methodology which is conceptually different from standard space/frequency domain filtering techniques.
Furthermore, by defining both global features (e.g. degree distribution) and local features (e.g. visibility patches, a novel tool which extends the concept of sequential visibility graph motifs [11, 36] to images) we are able to reach competitive results in image classification tasks, sometimes superior to state of the art results obtained by sophisticated algorithms with highly specialised context-dependent descriptors. We found that global features seem to be more informative when extracted from the IHVG than from IVG, while the opposite is true for local features (e.g. patches), at least for low-order patches ($p = 3$). We shall highlight that results are performed using low-order visibility patches ($p = 3$), so better performance is likely to be obtained with higher order (e.g. $p = 4$), for which linear time algorithms can also be designed [11, 36]. An hybrid combination of our newly defined set of visibility graph features with more standard ad-hoc (i.e. problem-dependent) features and automatic feature extraction methods (e.g. artificial neural networks) is likely to provide even better results: this is an open problem for further research.

We should emphasize that the set of features extracted from IVG/IHVG –such as degree distributions or visi-

ova-qSVM	Accuracy	⟨AUC⟩	MAA
Global multiplex features: P(k) in R,G,B + PCA IVG+IHVG (8 PCs+12 PCs)			
Usptex	93.6%	0.9829	93.2%
Brodatz Colored	98.7%	0.9929	98.3%
Multiband	98.2%	0.9982	97.8%
Local multiplex features: **Z** in R,G,B + PCA IVG+IHVG (15 PCs+18 PCs)			
Usptex	94.9%	0.9952	94.3%
Brodatz Colored	98.6%	0.9967	98.3%
Multiband	99.4%	0.9994	99.2%
Global + Local multiplex features IVG+IHVG (53 features)			
Usptex	**98.4%**	0.9975	98%
Brodatz Colored	**99.8%**	0.9993	99.7%
Multiband	**99.6%**	0.9996	99.4%

TABLE IV: Natural Color Textures Datasets: best classification accuracy, best average by class AUC and model average classification accuracy MAA obtained with a quadratic-kernel Multiclass one-vs-all Support Vector Machine (ova-qSVM) classifier algorithm over 30 models realizations, for different sets of multiplex image visibility features. Very slightly lower performance can be obtained by using a linear kernel SVM.

bility patches– is universal, in the sense that they are not designed to exploit particular, context-dependent properties of the image and can therefore be measured across classification tasks of different nature. Quoting Nanni [68], "The present trend in machine learning is focused on building optimal classification systems for very specific, well-defined problems. Another research focus, however, would be to work on building General-Purpose (GP) systems that are capable of handling a broader range of problems as well as multiple data types. Ideally, GP systems would work well out of the box, requiring little to no parameter tuning but would still perform competitively against less flexible systems that have been optimized for very specific problems and datasets."

Accordingly, we envisage that this methodology can have a potential impact in pattern recognition and image classification tasks across the disciplines, opening a plethora of further research directions. From a theoretical perspective, we recall that visibility graphs designed for time series analysis purposes have been shown previously to be amenable to analytic insight [10]. We conjecture that theoretical analysis –including analytical results– for image visibility graphs is also possible. Other interesting open problems for further research include: (i) capitalizing on the properties of multiplex networks for extracting correlation-based multiplex features from RGB images, such as degree-degree correlations across layers [44], and the (ii) using the whole multiplex setting –and more particularly the inter-layer and intrinsically multiplex descriptors– to study those situations where interdependence between layers is likely to take place, such as for instance in temporally ordered sequence of images (videos).

Acknowledgments. We thank N. Sadawi for interesting discussions and anonymous referees for helpful comments. LL acknowledges funding from EPSRC Fellowship EP/P01660X/1. JI acknowledges funding from Wellcome Trust Metaboflow Grant 202952Glen.

Codes. All codes used in this paper are available under request.

---

[1] L. Lacasa, B. Luque, F.J. Ballesteros, J. Luque, and J.C. Nuno, From time series to complex networks: the visibility graph, *Proc. Natl. Acad. Sci. USA* **105**, 13 (2008).

[2] B. Luque, L. Lacasa, J. Luque, F.J. Ballesteros, Horizontal visibility graphs: exact results for random time series, *Phys. Rev. E* **80**, 046103 (2009).

[3] L. Lacasa, V. Nicosia, V. Latora, Network Structure of Multivariate Time Series, *Sci. Rep.* **5**, 15508 (2015)

[4] V. Latora, V. Nicosia and G. Russo, *Complex Networks: Principles, Methods and Applications* (Cambridge University Press, 2017)

[5] Z-K Gao, M. Small and J. Kurths, Complex network analysis of time series, *EPL* **116**, 5 (2017).

[6] M. Kivela, A. Arenas, M. Barthelemy, J.P. Gleeson, Y. Moreno and M.A. Porter, Multilayer networks. *J. Complex Networks* **2** (3), 203-271 (2014).

[7] S. Boccaletti, G. Bianconi, R. Criado, C.I. del Genio, J. Gomez-Gardenes, M. Romance, I. Sendina-Nadal, Z. Wang, and M. Zanin. The structure and dynamics of multilayer networks. *Phys. Rep.* **544** (1), 1-122 (2014).

[8] S. Severini, G. Gutin, T. Mansour, A characterization of horizontal visibility graphs and combinatorics on words, *Physica A* **390**, 12 (2011) 2421-2428.

[9] B. Luque, L. Lacasa, Canonical horizontal visibility

graphs are uniquely determined by their degree sequence, *Eur. Phys. J. Sp. Top.* 226, 383 (2017).

[10] L. Lacasa, On the degree distribution of horizontal visibility graphs associated to Markov processes and dynamical systems: diagrammatic and variational approaches, *Nonlinearity* **27**, 2063-2093 (2014).

[11] J. Iacovacci and L. Lacasa, Sequential visibility-graph motifs, *Phys. Rev. E* 93, 042309 (2016)

[12] L. Lacasa, B. Luque, J. Luque and J.C. Nuño, The Visibility Graph: a new method for estimating the Hurst exponent of fractional Brownian motion, *EPL* 86 (2009) 30001.

[13] B. Luque, L. Lacasa, F. Ballesteros, A. Robledo, Analytical properties of horizontal visibility graphs in the Feigenbaum scenario, *Chaos* **22**, 1 (2012) 013109.

[14] B. Luque, A. Núñez, F. Ballesteros, A. Robledo, Quasiperiodic Graphs: Structural Design, Scaling and Entropic Properties, *Journal of Nonlinear Science* **23**, 2, (2012) 335-342.

[15] A.M. Núñez, B. Luque, L. Lacasa, J.P. Gómez, A. Robledo, Horizontal Visibility graphs generated by type-I intermittency, *Phys. Rev. E*, **87** (2013) 052801.

[16] Shao, Z. G. Network analysis of human heartbeat dynamics. Applied Physics Letters, 96, 7 (2010) 073703.

[17] Ahmadlou M, Adeli H, Adeli A. New diagnostic EEG markers of the Alzheimer's disease using visibility graph. *Journal of neural transmission* 117(9) (2010)1099-109.

[18] Ahmadlou, M., Adeli, H., & Adeli, A. Improved visibility graph fractality with application for the diagnosis of autism spectrum disorder. *Physica A* 391, 20 (2012) 4720-4726.

[19] Bhaduri, S., & Ghosh, D. Electroencephalographic data analysis with visibility graph technique for quantitative assessment of brain dysfunction. *Clinical EEG and neuroscience* 46, 3 (2015), 218-223.

[20] Sannino, S., Stramaglia, S., Lacasa, L., & Marinazzo, D. Visibility graphs for fMRI data: multiplex temporal graphs and their modulations across resting state networks. *Network Neuroscience* (2017)

[21] S. Jiang, C. Bian, X. Ning and Q.D.Y. Ma, Visibility graph analysis on heartbeat dynamics of meditation training, *Appl. Phys. Lett.* **102** 253702 (2013).

[22] U. Hasson, J. Iacovacci, B. Davis, R. Flanagan, E. Tagliazucchi, H. Laufs, L. Lacasa A combinatorial framework for peak/pit asymmetry in complex dynamics, *Sci. Rep.* **8**, 3557 (2018).

[23] A. Aragoneses, L. Carpi, N. Tarasov, D.V. Churkin, M.C. Torrent, C. Masoller, and S.K. Turitsyn, Unveiling Temporal Correlations Characteristic of a Phase Transition in the Output Intensity of a Fiber Laser, *Phys. Rev. Lett.* 116, 033902 (2016).

[24] M. Murugesana and R.I. Sujitha1, Combustion noise is scale-free: transition from scale-free to order at the onset of thermoacoustic instability, *J. Fluid Mech.* 772 (2015).

[25] A. Charakopoulos, T.E. Karakasidis, P.N. Papanicolaou and A. Liakopoulos, The application of complex network time series analysis in turbulent heated jets, *Chaos* 24, 024408 (2014).

[26] P. Manshour, M.R. Rahimi Tabar and J. Peinche, Fully developed turbulence in the view of horizontal visibility graphs, *J. Stat. Mech.* (2015) P08031.

[27] RV Donner, JF Donges, Visibility graph analysis of geophysical time series: Potentials and possible pitfalls, *Acta Geophysica* 60, 3 (2012).

[28] V. Suyal, A. Prasad, H.P. Singh, Visibility-Graph Analysis of the Solar Wind Velocity, *Solar Physics* **289**, 379-389 (2014)

[29] Y. Zou, R.V. Donner, N. Marwan, M. Small, and J. Kurths, Long-term changes in the north-south asymmetry of solar activity: a nonlinear dynamics characterization using visibility graphs, *Nonlin. Processes Geophys.* **21**, 1113-1126 (2014).

[30] J.F. Donges, R.V. Donner and J. Kurths, Testing time series irreversibility using complex network methods, *EPL* 102, 10004 (2013).

[31] L. Lacasa, J. Iacovacci, Visibility graphs of random scalar fields and spatial data *Phys. Rev. E* **96**, 012318 (2017)

[32] A. Turner, M. Doxa, D. O'sullivan, and A. Penn, From isovists to visibility graphs: a methodology for the analysis of architectural space. *Environment and Planning B: Planning and design* 28 (1), 103-121 (2001).

[33] Lézoray, O. and Grady, L. eds., 2012. Image processing and analysis with graphs: theory and practice. CRC Press.

[34] R. Gonzalez, Rafael, *Digital Image Processing, 3rd* (Pearson Hall, 2008).

[35] Haralick, R.M. and Shanmugam, K., 1973. Textural features for image classification. IEEE Transactions on systems, man, and cybernetics, (6), pp.610-621.

[36] J. Iacovacci and L. Lacasa, Sequential motif profile of natural visibility graphs *Phys. Rev. E* 94, 052309 (2016).

[37] L. Lacasa, J. Iacovacci, Theory of visibility patches, in preparation.

[38] L.F.S. Scabini, R.H.M. Condori, W.N. Gonçalves, O.M.Bruno, Multilayer Complex Network Descriptors for Color-Texture Characterization, arXiv:1804.00501v1

[39] G. Kylberg. The Kylberg Texture Dataset v. 1.0, Centre for Image Analysis, Swedish University of Agricultural Sciences and Uppsala University, External report (Blue series) No. 35. Available online at: http://www.cb.uu.se/?gustaf/texture/

[40] Kylberg G., and Sintorn I.-M. Evaluation of Noise Robustness for Local Binary Pattern Descriptors in Texture Classification EURASIP Journal on Image and Video Processing, 2013, 17.

[41] Andrearczyk, V. and Whelan, P.F., 2016. Using filter banks in convolutional neural networks for texture classification. Pattern Recognition Letters, 84, pp.63-69.

[42] Boland, M.V. and Murphy, R.F., 2001. A neural network classifier capable of recognizing the patterns of all major subcellular structures in fluorescence microscope images of HeLa cells. Bioinformatics, 17(12), pp.1213-1223.

[43] A. Chebira, Y. Barbotin, C. Jackson, T. Merryman, G. Srinivasa, R.F. Murphy, and J. Kovacevic (2007). A multiresolution approach to automated classification of protein subcellular location images. BMC Bioinformatics 8:210.

[44] F. Battiston, V. Nicosia, and V. Latora, Structural measures for multiplex networks, *Phys. Rev. E* 89, 032804 (2014).

[45] Boccaletti S, Bianconi G, Criado R, Del Genio CI, Gómez-Gardenes J, Romance M, et al. The structure and dynamics of multilayer networks. Physics Reports 544(1):1–122 (2014).

[46] Bianconi G. Statistical mechanics of multiplex networks: Entropy and overlap. Physical Review E. 87(6):062806 (2013)

[47] Nicosia V, Latora V. Measuring and modeling cor-

relations in multiplex networks. Physical Review E. 92(3):032805 (2015)

[48] Iacovacci J, Wu Z, Bianconi G. Mesoscopic structures reveal the network between the layers of multiplex data sets. Physical Review E 92(4) 042806 (2015)

[49] De Domenico M, Lancichinetti A, Arenas A, Rosvall M. Identifying modular flows on multilayer networks reveals highly overlapping organization in interconnected systems. Physical Review X. 2015;5(1):011027.

[50] Iacovacci, J. and Bianconi, G., 2016. Extracting information from multiplex networks. Chaos: An Interdisciplinary Journal of Nonlinear Science, 26(6), p.065306.

[51] Mondragon, R.J., Iacovacci, J. and Bianconi, G., 2018. Multilink communities of multiplex networks. PloS one, 13(3), p.e0193821.

[52] Iacovacci, J., Rahmede, C., Arenas, A. and Bianconi, G., 2016. Functional multiplex pagerank. EPL (Europhysics Letters), 116(2), p.28004.

[53] Halu, A., Mondragón, R.J., Panzarasa, P. and Bianconi, G., 2013. Multiplex pagerank. PloS one, 8(10), p.e78293.

[54] Cozzo, E., Kivelä, M., De Domenico, M., Solé-Ribalta, A., Arenas, A., Gómez, S., Porter, M.A. and Moreno, Y., 2015. Structure of triadic relations in multiplex networks. New Journal of Physics, 17(7), p.073029.

[55] Battiston, F., Nicosia, V., Chavez, M. and Latora, V., 2017. Multilayer motif analysis of brain networks. Chaos: An Interdisciplinary Journal of Nonlinear Science, 27(4), p.047404.

[56] Cardillo A, Gómez-Gardenes J, Zanin M, Romance M, Papo D, Del Pozo F, et al. Emergence of network features from multiplexity. Scientific reports. 2013;3:1344.

[57] Sonka, M., Hlavac, V. and Boyle, R., 2014. Image processing, analysis, and machine vision. Cengage Learning.

[58] Liu, L., Chen, J., Fieguth, P., Zhao, G., Chellappa, R. and Pietikainen, M., 2018. A Survey of Recent Advances in Texture Representation. arXiv preprint arXiv:1801.10324.

[59] Hossain, S. and Serikawa, S., 2013. Texture databases: a comprehensive survey. *Pattern Recognition Letters* 34(15), pp.2007-2022.

[60] Bianconi, F. and Fernández, A., 2014. An appendix to Texture databases: A comprehensive survey. *Pattern recognition letters*, 45, pp.33-38.

[61] Backes, A.R., Casanova, D. and Bruno, O.M., 2012. Color texture analysis based on fractal descriptors. Pattern Recognition, 45(5), pp.1984-1992.

[62] Brodatz, P., 1966. A Photographic Album for Arts and Design.

[63] Abdelmounaime, S. and Dong-Chen, H., 2013. New Brodatz-based image databases for grayscale color and multiband texture analysis. *ISRN Machine Vision* 2013.

[64] Adly, H. and Moustafa, M., 2017. A hybrid deep learning approach for texture analysis. arXiv preprint arXiv:1703.08366.

[65] Nanni, L., Lumini, A. and Brahnam, S. Survey on LBP based texture descriptors for image classification. *Expert Systems with Applications* 39, 3 (2012) pp.3634-3641.

[66] Jain, L.C., Nanni, L. and Lumini, A, eds. *Local binary patterns: new variants and applications* (Springer, 2016)

[67] Ojala, T., Pietikäinen, M. and Harwood, D., 1996. A comparative study of texture measures with classification based on featured distributions. Pattern recognition, 29(1), pp.51-59.

[68] Nanni, L., Brahnam, S., Ghidoni, S. and Lumini, A., 2015. Toward a general-purpose heterogeneous ensemble for pattern classification. Computational intelligence and neuroscience, 2015, p.85.

[69] Nanni, L., Brahnam, S., Ghidoni, S. and Menegatti, E., 2015. Improving the descriptors extracted from the cooccurrence matrix using preprocessing approaches. Expert Systems with Applications, 42(22), pp.8989-9000.

[70] Leung, T. and Jitendra M.. "Representing and recognizing the visual appearance of materials using three-dimensional textons." International journal of computer vision 43, no. 1 (2001): 29-44.

[71] Varma, Manik, and Andrew Zisserman. "A statistical approach to texture classification from single images." International journal of computer vision 62, no. 1-2 (2005): 61-81.

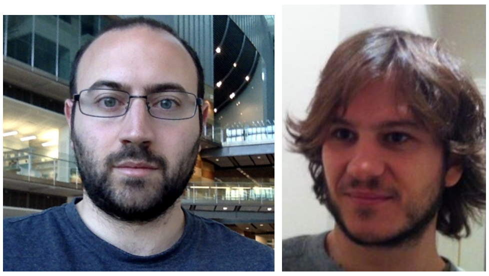

FIG. 16: (Left) Jacopo Iacovacci received a B.S. degree in Physics from University of Rome 'La Sapienza' (2010), and a M.S. degree in Biophysics (2012) from the same university. In 2017 he obtained his Ph.D. degree in Mathematics from Queen Mary University of London. He is currently a postdoctoral research associate at Imperial College London and at the Francis Crick institute, and his research interests include machine learning and network-based analysis techniques for data-driven biology. (Right) Lucas Lacasa received the B.S. degree in Physics from the Universidad Complutense de Madrid and the PhD degree in Physics of Complex Systems from the Technical University of Madrid in 2009. Currently, he is Senior Lecturer in Applied Mathematics at Queen Mary University of London and an EPSRC Early Career Fellow. Prior to that he held a lectureship at QMUL and an assistant professorship at Technical University of Madrid. His main research focus is on Complexity Science, and more concretely on the interface between time series analysis and network science. Other areas of interest are the applications of statistical physics and nonlinear dynamics methods to the analysis of complex systems.