Assessing Texture Descriptors for Seismic Image Retrieval

Andrea Britto Mattos*, Rodrigo S. Ferreira*, Reinaldo M. da Gama e Silva*, Mateus Riva† and Emilio Vital Brazil*
*IBM Research, Rio de Janeiro, Brazil, +55-21-21323380
*IME-USP, São Paulo, Brazil, +55-11-30916101
{abritto|rosife|rmozart|evital}@br.ibm.com, mriva@ime.usp.br

***Abstract*—Much work has been done on the assessment of texture descriptors for image retrieval in many domains. In this work, we evaluate the accuracy and performance of three well-known texture descriptors – Gabor Filters, GLCM, and LBP – for seismic image retrieval. These subsurface images pose challenges yet not thoroughly investigated in previous works, which are addressed and evaluated in our experiments. We asked for domain experts to annotate two seismic cubes, Penobscot 3D and Netherlands F3, and used them to evaluate texture descriptors, corresponding parameters, and similarity metrics with the potential to retrieve visually similar regions of the considered datasets. While GLCM is used in the vast majority of works in this area, our findings indicate that LBP has the potential to produce satisfying results with lower computational cost.**

I. Introduction

Image information became more and more prevalent in the last two decades raising the demand for techniques to retrieve image data in the same manner one would search for text documents. A retrieval performed on image metadata is dependent on annotation quality; alternatively, in Content-Based Image Retrieval (CBIR), images in a database are queried based on their characteristics, using Computer Vision-based tools. CBIR is present in several domains, such as seismic interpretation, which consists of the analysis of seismic data to generate reasonable models and predictions about the properties and structures of the subsurface [1]. After data acquisition and post-processing, seismic structures – or *facies*[1] – may be visually differentiated by their textural features.

In seismic analysis, texture descriptors are used to compute additional attributes of the seismic image, thus providing new visualizations of the data for interpreters [3], [4] or assisting semi-/automatic classification and clustering techniques [5]–[8]. However, texture features are often considered simply second-order seismic attributes, and, to the best of our knowledge, despite sharing potentially valuable results, very few papers explored the segmentation, classification, or retrieval of seismic images focusing purely on Computer Vision (CV) [9], [10]. One factor that may have influenced on the limited participation of CV researchers in the seismic-related literature is the lack of public, high-quality, annotated data. Although synthetic seismic data may be useful in various applications, it usually does not represent the visual complexities of a real survey. To address this problem, two 3D cubes available at the *Open Seismic Repository* [11] – namely, Netherlands Offshore F3 Block and Penobscot 3D Survey – were manually annotated by geoscientists for this work.

Another aspect identified in the seismic literature is that the choice of image descriptors and corresponding parameters are often based on suggestions of other papers or the interpreter's experience. This statement may be verified in the next section, as the vast majority of works explore GLCM (Grey Level Co-occurrence Matrix) for texture-based seismic analysis, without a proper justification or explanation on the parameters' choice. Therefore, our goal is to conduct a deeper analysis by exploring additional texture features that are prominent in other domains, but have not received attention in the seismic domain yet. Namely, we investigate the ability of Gabor Filters and LBP (Local Binary Patterns) – this last, widely used for face recognition [12] – to retrieve similar regions of a seismic cube.

This paper is structured as follows. In section II, we list previous works addressing the automatic analysis of seismic images. In section III, we provide a brief explanation of the considered texture descriptors and characteristics of the used database. In section IV, we describe the methodology applied in our experiments and the results are presented and discussed in section V. The paper is concluded in section VI.

II. Related Work

The potential of CV methods to aid seismic interpretation is undisputed and has been increasing the amount of research in the field. Dumay and Fournier [5] explored multivariate statistical analysis of attributes such as power spectrum information, analytical signal modulus, and autocorrelation for recognition of seismic facies. Their work was one of the first to show the viability of automatic seismic facies recognition; however, they also indicate that the system's reasons for choosing variables are not clear for a human interpreter.

West *et al.* [6] proposed the use of Probabilistic Neural Networks for supervised classification of seismic facies in 2D images, using GLCM texture attributes (homogeneity, entropy, energy, and inertia) as input for the classifier. Their work was one of the first to successfully apply GLCM attributes to a machine learning-based analysis framework while pointing to

[1]Seismic facies are mappable, three-dimensional units composed of groups of reflections whose parameters differ from those of adjacent facies units [2].

the necessity of high-quality data and an interpreter's understanding of the technology to guarantee a correct analysis.

Chopra and Alexeev [4] also used GLCM attributes (specifically, homogeneity, entropy, energy, and contrast) for assisting expert interpretation of seismic images. Corradi *et al.* [8] classified a seismic volume using dip-steered GLCM-generated textural attributes as input for a statistical neural network. However, they did not specify the used GLCM attributes.

Gao [3] proposed the use of a Voxel Co-occurrence Matrix (VCM) as a 3D extension of GLCM that can carry more relevant information (at a higher computational cost), utilizing homogeneity, contrast, and randomness attributes for segmentation and classification of seismic regions.

Wallet and Pepper [13] applied mathematical morphological operations to the GLCM energy attribute for assisting visualization and analysis of salt bodies. Wilhelm and Li [14] extended existing layer-tracking algorithms by integrating texture information for performance improvement. The authors advocated that GLCM contrast and the angular second moment are capable of distinguishing salt regions by texture.

Zhao *et al.* [7] conducted a comparative review of many unsupervised and supervised methods for seismic facies recognition, such as k-Means, SOMs (Self-Organizing Maps), GTMs (Generative Topographic Mapping), ANNs (Artificial Neural Networks), and SVMs (Support Vector Machines). The texture attributes used were GLCM homogeneity and entropy. k-Means clustering provided a computationally inexpensive but rough segmentation results while SOMs and GTMs provided better results while requiring higher input quality.

Amin *et al.* [15] presented a mixture of texture attributes for detection of salt domes, combining GLCM entropy, energy, and variance and Gradient of Texture (GoT) attributes with a dictionary-based learning approach. The proposed approach presented good accuracy, outperforming many existing techniques.

Following an alternative method, Hegazy and Regib [16] proposed the thresholding of the combination of three simple texture attributes (directionality, smoothness, and edge content) for detection and differentiation of salt bodies in seismic volumes. While results are not as robust as other techniques, their findings are encouraging and, most importantly, the computational cost of the chosen attributes is very low.

Long *et al.* [9] compared the capabilities of different texture attributes for characterization of seismic image retrieval. The studied attributes were Steerable Pyramid, Curvelet Transform, Local Binary Pattern, and Local Radius Index. The study strengthens the hypothesis that texture attributes are capable of seismic characterization and, to the best of our knowledge, is the first to explore LBP for seismic retrieval. Similarly, Ferreira *et al.* [10] evaluated several descriptors – more precisely, GLCM, Gabor Filters, Histogram features, Gradient features, Hessian features, and Structure Tensor – for detecting salt domes based on multi-scale clustering.

III. Fundamental Concepts

In this section, we present the texture descriptors used in this work and describe how the retrieval database was built.

A. Texture Descriptors

Many techniques exist for codifying the neighborhood information of a pixel, creating a vector of descriptive values. Such vector (*Texture Descriptor*) is key for seismic image retrieval, as they are represented by only one channel of grey-scale values.

1) Local Binary Patterns: Since first proposed in 1996 [17], LBP has become a popular texture descriptor, mainly because of its simplicity and efficiency, being used in a wide range of biometric applications.

LBP is a technique that encodes the neighborhood information of a pixel to a binary number. Given a distance $d \geq 1$ and a number of neighbours n, we define a neighborhood for a pixel p as $N = \{p_1, \ldots, p_n\}$ where all p_i are d distant of p. The code of pixel p based on its neighborhood N is defined as a binary number with n digits, where the i^{th} digit is 0 if $p_i \leq p$ and 1 otherwise. See Figure 1 for an example. The feature vector will be the histogram of the binary codes of a given window. Notice that there are 2^n possible codes, then the vector can have up to 2^n dimensions.

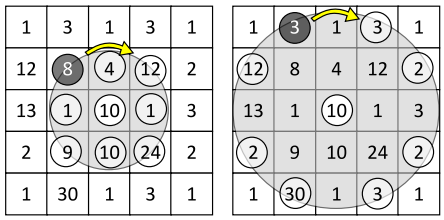


Fig. 1. LBP scheme. The white discs represent the encoded pixel while the dark discs represent the first pixel to be considered (p_1). The left one has $d = 1$ and the right one has $d = 2$, both with $n = 8$. Their binary codes are 00101000 and 0000101, respectively.

2) Grey Level Co-occurrence Matrix: GLCM was proposed by Haralick [18] and is a traditional image processing technique used to describe textures that has been widely applied in seismic interpretation (as seen in section II). For each window, GLCM creates a set of matrices $k \times k$, where k is the number of gray levels of the image. These matrices encode the co-occurrence of patterns in the window and are used to calculate specific features. Each matrix is defined by two parameters, the angle ϕ and the distance d, creating a mask to assemble the co-occurrence matrix. In Table I, we can observe the four GLCMs associated with the examples in Figure 2. Notice that the GLCM matrix is symmetric by construction.

After calculating the GLCMs, we use the features proposed by Haralick to obtain the descriptors based on each matrix, i.e. for each chosen pair of ϕ and d, we have a distinct set of features, which are calculated using the entries $A_{i,j}$ of the GLCM A. Several features exist, and here we list some of them:

- contrast = $\sum_{ij}(i-j)^2 A_{ij}$

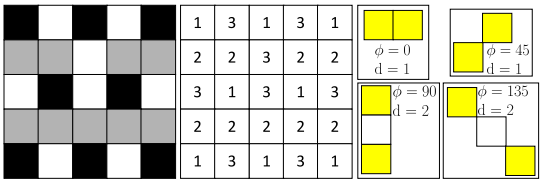


Fig. 2. An example of a three-gray-level image and four GLCM masks.

TABLE I
GLCM OF MASKS SHOWED IN FIGURE 2.

$\phi = 0,\ d = 1$	$\phi = 45,\ d = 1$	$\phi = 90,\ d = 2$	$\phi = 135,\ d = 2$
$\begin{pmatrix} 0 & 0 & 12 \\ 0 & 6 & 2 \\ 12 & 2 & 0 \end{pmatrix}$	$\begin{pmatrix} 0 & 7 & 1 \\ 7 & 0 & 7 \\ 1 & 7 & 1 \end{pmatrix}$	$\begin{pmatrix} 0 & 0 & 10 \\ 0 & 4 & 1 \\ 10 & 1 & 0 \end{pmatrix}$	$\begin{pmatrix} 0 & 0 & 6 \\ 0 & 2 & 1 \\ 6 & 1 & 0 \end{pmatrix}$

- dissimilarity = $\sum_{ij} |i-j| A_{ij}$
- homogeneity = $\sum_{ij} (A_{ij})^2$
- energy = $\sum_{ij} A_{ij}$, etc.

3) Gabor Filters: The CV community uses Gabor filter banks for many applications, such as edge detection and texture analysis [19]. The idea is to convolve the image using the Gabor filter with different parameters and use the statistics of the result (usually mean and variance) as texture descriptors. The discrete Gabor filters (sin and cos) are defined as:

$$G_c(i,j) = B \exp\left(-\frac{i^2+j^2}{2\sigma^2}\right) \cos\left(2\pi f \left(i\cos\theta + j\sin\theta\right)\right),$$

$$G_s(i,j) = C \exp\left(-\frac{i^2+j^2}{2\sigma^2}\right) \sin\left(2\pi f \left(i\cos\theta + j\sin\theta\right)\right).$$

The B and C parameters, respectively, are:

$$B = \sum_{ij} \exp\left(-\frac{i^2+j^2}{2\sigma^2}\right) \cos\left(2\pi f \left(i\cos\theta + j\sin\theta\right)\right),$$

$$C = \sum_{ij} \exp\left(-\frac{i^2+j^2}{2\sigma^2}\right) \sin\left(2\pi f \left(i\cos\theta + j\sin\theta\right)\right).$$

The other factors are radial frequency f and the orientation of the filter θ. In Figure 3, we can observe 12 Gabor filters.

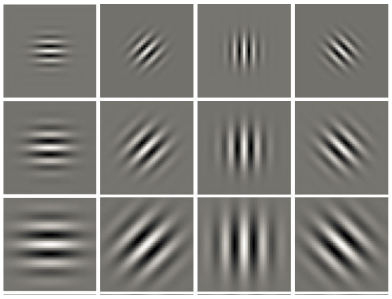


Fig. 3. An exemple of a Gabor filter set, $\theta \in \{0, 45, 90, 135\}$ and $f \in \{0.05, 0.2, 0.4\}$

B. Seismic Data

In this section, we will better define seismic data and how it is processed for image retrieval. Seismic images are acquired in a process in which a source generates vibrations (or an impulse) that travel into the Earth, pass through strata with different physical properties, and return to the surface being recorded as seismic data. This data is processed and then analyzed by interpreters who study the subsurface's properties and structures.

The seismic data used in this work can be seen as a 3D volume composed of a horizontal stack of 2D seismic images that we call *slices*. These vertical slices can be considered along the inline- and crossline-axes (lateral axes), while the vertical axis may represent depth or time, depending on how the data was processed. The experiments used as input three inlines from the Netherlands F3 dataset, with 951×462 pixels and three crosslines with 651×462 pixels. Also, we considered three inlines from the Penobscot dataset, with 481×1501 pixels and three crosslines with 601×1501 pixels. For each experiment, the central slice was selected as the query, and the two remaining slices were assigned to the database.

Experts previously interpreted each slice, thus assigning each pixel to one of 8 categories (Netherlands) or 7 categories (Penobscot). We remark that the labeling and the discussions of this study focus on image analysis aspects rather than geophysical or geological ones. Some samples of the used data may be seen in Figure 4.

The retrieval database was built by dividing each seismic slice into patches; we describe this step in more detail in section IV. Since a single seismic data volume can span dozens of gigabytes, the trade-off between online and offline retrieval must be evaluated according to the requirements of each application. In this work, we consider the offline case, *i.e.,* for each patch in the database, a texture descriptor is computed and stored for posterior comparison. As data storage and processing time may become issues depending on the adopted strategy (online or offline retrieval), the selection of the texture descriptor and its parameters must also take these aspects into consideration. Below, we list the size of the feature vector for each texture descriptor as a function of its parameters.

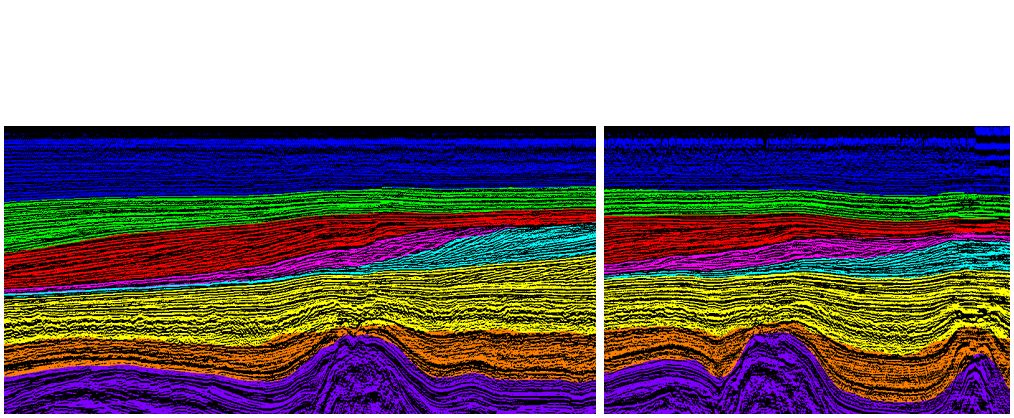
a) LBP: Despite the several variations of LBP, we have considered the original operator and the rotation-invariant version, both yielding 256 values for each patch.

b) GLCM: The size of the feature vector can be computed as $\#(features) \times \#(distances) \times \#(angles)$.

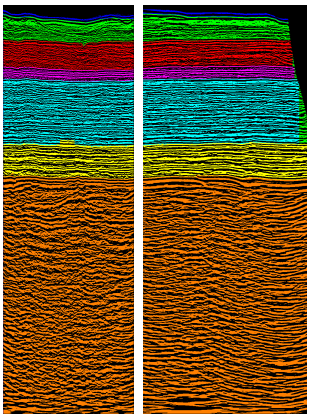
c) Gabor Filters: The size of the feature vector can be computed as $\#(features) \times \#(frequencies) \times \#(angles)$.

IV. METHODOLOGY

The general image retrieval process consists of the following steps: (i) receiving a query image from the user; (ii) searching in a database of multiple images; and (iii) retrieving the images from the database which are most visually similar to the query image. The visual similarity is computed by extracting texture features on both query and database and comparing the extracted feature vector according to a similarity metric,



(a) Inline (left) and crossline (right) from the Netherlands dataset.



(b) Inline (left) and crossline (right) from the Penobscot dataset.

Fig. 4. Examples of slices used in the experiments (re-scaled). The original grayscale images used as input are displayed above with an overlay of colors highlighting the regions resultant from the image annotation.

i.e., no training process is needed. This work aims to assist the interpreter in the task of manually selecting a region-of-interest (ROI) in a given slice – representing the query image – and searching in the remaining of the cube – representing the database – for the regions which are most similar to the query. In this paper, we address homogenous retrieval, *i.e.,* we assume that the ROI belongs to a single seismic category (as well as the candidates from the database).

As described in section III, each slice is subject to a process of division in patches, to which we refer as *tiles*. Each tile receives a unique label among the categories annotated by the experts. Then, given a query tile T, our goal is to return N tiles of the same seismic category of T.

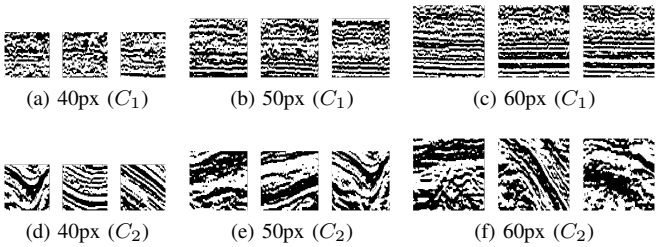
Thus, the goal of our experiments is twofold: (i) to determine the best combination of parameters (per descriptor) for the retrieval of homogeneous regions; and (ii) to compare the descriptors' accuracy and performance when using the optimal parameters. To this end, we implemented a Python-based framework for running empirical analysis on images. The next section describes the experimental setup in detail.

A. Global Experimental setup

Before serving as input for the experiments described next, each grayscale image (in `.tiff` format) underwent a pre-processing step in which its intensity values were rescaled to 64 gray levels (*i.e.*, every pixel received values in the range [0,63]), followed by a histogram equalization.

1) Tiling: We partitioned each slice in tiles of varying sizes, and each tile received a label according to its annotation. More precisely, if at least 75% of the tile's pixels belonged to a class C, the tile received label C and if a tile contained more that 25% of noise, it was discarded. We used different tile sizes to estimate the robustness of the descriptors when working with images at different scales. The considered tile sizes in our experiments depend on the image resolution and are listed below. A few samples are displayed in Figure 5.

- Netherlands: 40×40 px; 50×50 px; and 60×60 px.
- Penobscot: 40×40 px; 60×60 px; and 80×80 px.



(a) 40px (C_1) (b) 50px (C_1) (c) 60px (C_1)

(d) 40px (C_2) (e) 50px (C_2) (f) 60px (C_2)

Fig. 5. Samples of tiles with varying sizes from two different categories – C_1 and C_2. All samples were extracted from Netherlands' crosslines.

We evaluated two different strategies for dividing the slices into tiles: first, the tiles were assigned right next to each other (*i.e.*, no overlap); second, we considered 50% of overlap during division. Table II displays the number of generated tiles when considering different sizes and division modes.

TABLE II
TILE SIZE × NUMBER OF TILES GENERATED FOR QUERY AND DATASET, WHEN CONSIDERING THE TWO OVERLAPPING MODES.

	No overlap		50% overlap	
Size	Query	Dataset	Query	Dataset
Netherlands – inlines				
40 px	177	341	711	1421
50 px	120	241	393	781
60 px	53	112	225	454
Netherlands – crosslines				
40 px	114	220	461	950
50 px	82	155	271	514
60 px	40	72	144	295
Penobscot – inlines				
40 px	506	1033	1563	3126
60 px	213	417	643	1290
80 px	101	206	330	659
Penobscot – crosslines				
40 px	411	820	1955	3921
60 px	172	350	807	1618
80 px	86	171	412	829

2) Distance metrics: Given two feature vectors p and q with n elements, we selected and evaluated two similarity metrics: Euclidean distance and Manhattan distance.

3) Number of retrieved samples: We compare each query tile to all tiles from the database evaluating the visual similarity between them. Next, we return the top N most similar candidates from the database. In our experiments, we tested different values of N, namely, 5, 10 and 15. Nevertheless, we observed that the behavior of descriptors and corresponding parameters were consistent regardless the value of N. Therefore, for the sake of brevity, the results described in section V will only contemplate the values for $N = 5$.

4) Evaluation: We performed the descriptors' analysis by considering both the ability to retrieve similar regions and the time required to compute the feature vectors. Since each tile is associated with a single category, we could verify the number of candidates on the top N retrieved samples which indeed belonged to the same category as the query image. Thus, we compute the accuracy by taking the ratio between the retrieved samples of the query's category and the total of samples retrieved. We carried out the performance analysis by storing the elapsed time for each retrieval task.

5) Per-category accuracy: Finally, we not only consolidated the retrieval accuracy and performance for all the query tiles but also computed individually for the tiles of each category. The goal of the second analysis was to assess whether a combination of descriptors and corresponding parameters would work better for regions with specific visual characteristics. Moreover, because the data from Penobscot is highly unbalanced (see Figure 4b), we wanted to conduct a more detailed analysis.

B. Evaluating Descriptors' Parameters

Each descriptor was evaluated according to the experimental setup previously described. We also tested, for each descriptor, how different parameters values affected the retrieval quality.

We conducted several pre-experiments with parameters varying in different ranges, took the initial optimal values, and, finally, performed the experiments with parameters varying from ranges close to the initial optimal values. For the sake of brevity, the range values considered in the pre-experiments are omitted from this paper. The evaluated parameters and corresponding selected values are listed below.

- LBP
  - Radiuses: 1, 2, 3, 4, 5, and 6
  - Methods: `Default` and `ROR` (rotation-invariant)
  - Neighbors: 8 (fixed)
    - The computation for larger values is too slow and requires a lot of memory.
- GLCM
  - Features: `ASM`, `autocorr`, `contrast`, `correlation`, `cshade`, `cprominence`, `diffentropy`, `diffvar`, `dissimilarity`, `energy`, `entropy`, `homogeneity`, `invdiff`, `maxprob`, `mean`, `sumavg`, `sumentropy`, `sumvar`, and `variance`
  - Distances: 1, 2, and [1, 2] (combined)
  - Angles: [0°, 45°, 90°, 135°] (combined – fixed)
- Gabor
  - Frequencies: 0.25, 0.5, 0.75, and 1
  - Feature: `variance` (fixed)
  - Angles: [0°, 45°, 90°, 135°] (combined – fixed)

V. Results

We performed several experiments and computed the descriptors' accuracy for the top five most similar retrieved candidates, as described in section IV. The results for Netherlands cube are consolidated in Tables III and IV, and the results for Penobscot cube are shown in Tables V and VI. These tables were built similarly and each column displays, for each tile size, tile-overlapping mode, distance metric, and descriptor, the following values: (i) the parameter which yielded the highest accuracy; (ii) the highest accuracy obtained; and (iii) the time – in seconds – for each corresponding-retrieval task.

As seen in the tables, the optimal parameters of each descriptor may be affected by the tile size but, in general, the following rules apply:

- For the Netherlands dataset, the optimal radius value for LBP was 1 (see Tables III and IV - rows 1,2,7,8,13,14). More precisely, it was able to achieve the optimal accuracy for 70% of the tests performed with inlines, and 91% for crosslines.
- For the Penobscot dataset, the optimal radius value for LBP was around 1-2 for inlines (see Table V - rows 1,2,7,8,13,14) and 3-4 for crosslines (see Table VI - same rows). More precisely, for inlines, the radius 1 was the best parameter for 52% of the tests and radius 2 performed better at 47% of the tests; for crosslines, the radius 2 was selected at 63% of the tests and radius 3 at 36% of the tests.
- For both datasets, the default LBP operator tends to work better than the rotation-invariant version (LBP ROR), except for some of the larger tiles (compare rows 1 and 2, 7 and 8, 13 and 14, of Tables III-VI).
- For both datasets, the GLCM descriptor parametrized with distance [1,2] (combined) tends to outperform GLCM with distances 1 and 2 alone, however, the elapsed time is substantially superior (see rows 3-5, 9-11, 15-17, of Tables III-VI).
- For both datasets, the best features of GLCM were `dissimilarity` and `diffentropy`, followed by `contrast` (see rows 3-5, 9-11, 15-17, of Tables III-VI). More precisely, for the Netherlands dataset, the feature `dissimilarity` was the best parameter in 45% of the tests (inlines) and 31% of the tests (crosslines); while `diffentropy` was selected in 27% of the tests (inlines) and 39% of the tests (crosslines). For the Penobscot, `dissimilarity` was the best parameter in 29% of the tests (inlines) and 39% of the tests (crosslines); while

TABLE III
GLOBAL RESULTS FOR **NETHERLANDS INLINES**: BEST PARAMETER VALUE, ACCURACY OF BEST PARAMETER, AND ELAPSED TIME.

	Overlap	No overlap						50% overlap					
	Metric	Euclidean			Manhattan			Euclidean			Manhattan		
	Tile size	40px											
1	LBP Default	1	**77.7%**	**1.33s**	1	**78.1%**	2.03s	1	**78.2%**	**10.73s**	1	**78.7%**	**10.71s**
2	LBP ROR	1	70.8%	2.47s	1	72.6%	**1.93s**	2	70.2%	11.06s	1	70.6%	11.43s
3	GLCM 1	dissim.	71.4%	4.63s	diffent.	71.1%	21.68s	dissim.	72.2%	24.33s	diffent.	72.0%	85.91s
4	GLCM 2	dissim.	70.7%	4.72s	dissim.	70.1%	4.21s	dissim.	71.0%	22.85s	dissim.	70.1%	21.48s
5	GLCM [1,2]	dissim.	73.5%	6.20s	diffent.	71.6%	40.51s	diffent.	76.8%	172.16s	diffent.	75.6%	171.06s
6	Gabor Var.	0.5	65.8%	3.98s	0.5	67.6%	4.25s	0.5	70.6%	19.23s	0.5	70.7%	19.67s
	Tile size	50px											
7	LBP Default	1	**75.8%**	**1.42s**	1	**78.8%**	1.67s	1	**82.8%**	5.13s	1	**85.2%**	4.58s
8	LBP ROR	2	70.6%	1.46s	1	72.6%	**1.66s**	1	76.5%	**4.53s**	1	77.9%	**4.15s**
9	GLCM 1	diffent.	75.0%	13.09s	diffent.	75.0%	15.39s	dissim.	77.3%	12.90s	dissim.	76.9%	10.20s
10	GLCM 2	cprom.	75.8%	10.48s	cprom.	75.1%	10.63s	dissim.	76.7%	10.74s	dissim.	76.0%	8.65s
11	GLCM [1,2]	diffent.	78.6%	25.56s	diffent.	76.8%	28.46s	diffent.	80.5%	58.81s	dissim.	79.0%	15.89s
12	Gabor Var.	0.5	73.0%	3.21s	0.5	76.3%	3.29s	0.5	75.2%	8.86s	0.5	76.2%	7.34s
	Tile size	60px											
13	LBP Default	1	**82.6%**	0.89s	2	**84.9%**	0.94s	1	**82.2%**	2.76s	1	**84.0%**	2.73s
14	LBP ROR	2	77.7%	**0.72s**	2	78.8%	**0.61s**	2	79.6%	**2.73s**	2	80.9%	**2.59s**
15	GLCM 1	dissim.	78.4%	1.52s	contrast	78.8%	1.35s	dissim.	77.0%	5.64s	dissim.	76.6%	4.65s
16	GLCM 2	sumvar	79.2%	7.92s	diffent.	78.1%	6.75s	dissim.	79.2%	6.83s	dissim.	77.2%	5.72s
17	GLCM [1,2]	dissim.	80.3%	2.24s	cprom.	81.1%	8.47s	diffent.	80.0%	48.53s	diffent.	78.8%	43.76s
18	Gabor Var.	0.5	77.3%	1.67s	0.5	78.4%	1.71s	0.5	77.2%	6.03s	0.5	77.9%	5.50s

TABLE IV
GLOBAL RESULTS FOR **NETHERLANDS CROSSLINES**: BEST PARAMETER VALUE, ACCURACY OF BEST PARAMETER, AND ELAPSED TIME.

	Overlap	No overlap						50% overlap					
	Metric	Euclidean			Manhattan			Euclidean			Manhattan		
	Tile size	40px											
1	LBP Default	1	69.3%	**0.87s**	1	**70.0%**	1.33s	1	**67.9%**	6.52s	1	**68.3%**	8.08s
2	LBP ROR	1	65.9%	1.34s	1	67.0%	**1.24s**	1	59.7%	**6.00s**	1	59.6%	**6.44s**
3	GLCM 1	dissim.	65.9%	2.81s	dissim.	66.1%	2.90s	dissim.	64.5%	15.91s	cprom.	64.5%	42.49s
4	GLCM 2	dissim.	67.1%	2.85s	contrast	65.9%	2.84s	contrast	66.0%	15.89s	contrast.	64.5%	13.68s
5	GLCM [1,2]	contrast	67.8%	3.83s	contrast	67.0%	4.01s	contrast	69.8%	18.13s	diffent.	67.9%	101.18s
6	Gabor Var.	0.5	**69.4%**	2.27s	0.5	**70.0%**	2.52s	0.5	65.9%	11.11s	0.5	66.5%	11.70s
	Tile size	50px											
7	LBP Default	1	70.4%	**0.89s**	1	69.2%	0.93s	1	76.3%	3.46s	1	76.5%	**3.55s**
8	LBP ROR	2	68.0%	1.00s	2	68.0%	**0.91s**	1	68.7%	**3.29s**	1	69.5%	3.58s
9	GLCM 1	diffent.	71.9%	9.69s	diffent.	69.7%	8.94s	diffent.	72.9%	32.97s	diffent.	72.9%	32.12s
10	GLCM 2	diffent.	68.5%	10.45s	diffent.	70.0%	9.07s	diffent.	74.6%	31.18s	diffent.	73.5%	32.47s
11	GLCM [1,2]	diffent.	**74.3%**	19.26s	diffent.	71.2%	19.21s	diffent.	**79.6%**	62.63s	diffent.	**78.1%**	63.76s
12	Gabor Var.	0.5	71.7%	1.91s	0.5	**71.9%**	1.99s	0.5	71.8%	7.52s	0.5	73.5%	6.89s
	Tile size	60px											
13	LBP Default	1	77.5%	**0.47s**	1	77.5%	0.49s	1	80.0%	1.92s	1	81.3%	**2.07s**
14	LBP ROR	1	77.0%	0.49s	1	**78.5%**	**0.46s**	1	75.0%	**1.84s**	1	77.0%	**2.07s**
15	GLCM 1	correl.	**79.0%**	1.97s	sumvar	77.0%	5.89s	diffent.	77.5%	19.20s	diffent.	77.3%	18.51s
16	GLCM 2	dissim.	72.0%	1.10s	diffent.	71.0%	4.76s	contrast	76.3%	4.10s	sument.	74.0%	25.46s
17	GLCM [1,2]	dissim.	76.0%	1.48s	dissim.	77.0%	1.47s	diffent.	79.7%	36.28s	diffent.	78.3%	34.29s
18	Gabor Var.	0.5	74.5%	1.08s	0.5	77.0%	1.07s	0.5	**80.6%**	4.52s	0.5	**82.78%**	4.30s

`diffentropy` was selected in 45% of the tests (inlines) and 12% of the tests (crosslines).

- For the Netherlands dataset, the optimal frequency of Gabor Filters was 0.5 in all the tests (see Tables III and IV - rows 6,12,18).
- For Penobscot, the optimal frequency value for Gabor was around 0.5-1.0 for inlines (see Table V - rows 6,12,18) and 1.0 for crosslines (see Table VI - same rows).

A. Experiments per-category

In addition to the global results previously reported, we also ran tests per-category, *i.e.*, for the query tiles of each category, the retrieval was performed for the entire dataset, and we computed the individual accuracy for each category. Tables VII and VIII consolidate the results for Netherlands and Penobscot datasets, respectively. Each line of the table displays the mean accuracy (*i.e.*, the average between values for different tile sizes, overlapping modes, and distance metrics) for each category when using the optimal parameters for each descriptor. Because the fourth layer (top-down) of Netherlands cube generates a set of tiles with insufficient size for our analysis, the results are omitted.

B. Discussion

Regardless of the tile size and overlapping mode, our experiments indicate that LBP, Gabor, and GLCM, when applied

TABLE V
GLOBAL RESULTS FOR **PENOBSCOT INLINES**: BEST PARAMETER VALUE, ACCURACY OF BEST PARAMETER, AND ELAPSED TIME.

	Overlap	No overlap						50% overlap					
	Metric	Euclidean			Manhattan			Euclidean			Manhattan		
	Tile size	40px											
1	LBP Default	1	85.6%	5.61s	1	**87.4%**	5.80s	1	**88.4%**	21.05s	1	**89.05%**	22.27s
2	LBP ROR	2	82.2%	**5.38s**	2	83.0%	**4.93s**	2	84.5%	**20.33s**	2	84.8%	**20.35s**
3	GLCM 1	diffent.	82.0%	52.18s	diffent.	82.3%	51.34s	diffent.	84.5%	113.74s	diffent.	84.4%	114.61s
4	GLCM 2	contrast	82.2%	11.33s	contrast	81.4%	11.97s	contrast	84.5%	35.69s	contrast	84.4%	36.84s
5	GLCM [1,2]	diffent.	**85.7%**	87.64s	diffent.	84.4%	99.27s	diffent.	88.5%	216.22s	diffent.	87.5%	211.13s
6	Gabor Var.	0.5	80.9%	10.73s	0.75	81.1%	8.57s	0.5	83.2%	31.61s	0.5	83.4%	31.08s
	Tile size	60px											
7	LBP Default	1	91.7%	**2.48s**	2	93.6%	3.04s	1	92.5%	10.86s	1	93.5%	8.74s
8	LBP ROR	2	90.4%	3.08s	2	91.5%	**2.37s**	2	92.0%	**8.59s**	2	92.2%	**7.60s**
9	GLCM 1	dissim.	95.7%	4.87s	dissim.	95.8%	5.57s	diffent.	92.1%	83.64s	diffent.	92.0%	64.32s
10	GLCM 2	dissim.	95.1%	5.11s	dissim.	94.8%	5.42s	dissim.	92.9%	17.38s	dissim.	92.6%	15.05s
11	GLCM [1,2]	dissim.	**96.2%**	6.87s	dissim.	**96.2%**	8.37s	diffent.	**93.9%**	127.75s	diffent.	**93.9%**	126.74s
12	Gabor Var.	1	91.6%	3.30s	0.75	91.2%	3.37s	1	89.4%	12.28s	1	89.7%	9.64s
	Tile size	80px											
13	LBP Default	4	92.5%	**1.55s**	3	93.7%	**1.44s**	1	94.3%	**5.71s**	1	**95.8%**	6.20s
14	LBP ROR	3	94.6%	1.93s	3	93.9%	1.58s	3	94.3%	6.55s	2	94.8%	**5.84s**
15	GLCM 1	diffent.	93.0%	10.16s	diffent.	93.2%	10.57s	diffent.	94.3%	46.28s	diffent.	94.5%	43.09s
16	GLCM 2	dissim.	**95.3%**	2.63s	diffent.	**95.1%**	11.10s	contrast	94.5%	11.18s	correl.	94.2%	19.91s
17	GLCM [1,2]	correl.	94.6%	7.51s	diffent.	94.4%	22.26s	diffent.	**95.1%**	84.85s	diffent.	95.1%	84.08s
18	Gabor Var.	1	95.1%	1.72s	0.75	**95.1%**	1.92s	1	93.3%	7.78s	1	92.9%	7.56s

TABLE VI
GLOBAL RESULTS FOR **PENOBSCOT CROSSLINES**: BEST PARAMETER VALUE, ACCURACY OF BEST PARAMETER, AND ELAPSED TIME.

	Overlap	No overlap						50% overlap					
	Metric	Euclidean			Manhattan			Euclidean			Manhattan		
	Tile size	40px											
1	LBP Default	2	**71.7%**	7.74s	2	**72.8%**	**7.13s**	2	72.6%	29.75s	2	**75.1%**	28.69s
2	LBP ROR	2	68.3%	**6.98s**	2	69.8%	7.55s	3	69.6%	**29.13s**	2	71.1%	**27.60s**
3	GLCM 1	diffent.	69.4%	65.66s	diffent.	69.2%	63.51s	diffent.	70.0%	163.59s	diffent.	70.0%	151.14s
4	GLCM 2	sumvar	68.9%	82.07s	contrast	68.2%	14.60s	contrast	70.0%	46.07s	contrast	70.0%	44.44s
5	GLCM [1,2]	diffent.	69.8%	111.43s	diffent.	69.3%	130.47s	diffent.	**72.7%**	274.55s	diffent.	72.0%	254.91s
6	Gabor Var.	0.75	66.4%	9.59s	1	67.7%	8.68s	1	68.3%	34.38s	1	68.5%	66.28s
	Tile size	60px											
7	LBP Default	3	**77.1%**	**3.00s**	2	**77.5%**	**2.84s**	2	**78.5%**	13.19s	2	**80.7%**	**9.04s**
8	LBP ROR	4	74.8%	3.46s	3	76.0%	2.99s	3	77.7%	**8.83s**	2	78.7%	9.22s
9	GLCM 1	dissim.	74.6%	5.78s	dissim.	74.0%	6.22s	diffent.	75.0%	84.43s	dissim.	74.9%	16.95s
10	GLCM 2	diffent.	74.4%	27.17s	diffent.	74.4%	27.39s	correl.	75.5%	41.39s	contrast	75.3%	16.95s
11	GLCM [1,2]	diffent.	74.7%	56.42s	sument.	74.9%	68.59s	diffent.	77.3%	177.22s	diffent.	76.5%	107.85s
12	Gabor Var.	1	70.7%	3.81s	1	71.1%	3.69s	1	73.4%	12.46s	1	73.8%	11.63s
	Tile size	80px											
13	LBP Default	4	80.7%	2.11s	2	82.5%	**1.87s**	2	**82.2%**	7.77s	2	**85.3%**	**6.42s**
14	LBP ROR	3	81.5%	**1.93s**	3	**83.7%**	2.03s	3	81.3%	**7.61s**	3	83.8%	6.85s
15	GLCM 1	diffvar	80.5%	13.47s	diffent.	80.0%	12.05s	diffvar	80.1%	52.93s	contrast	80.1%	13.70s
16	GLCM 2	diffvar	81.5%	13.14s	diffvar	82.1%	12.03s	diffvar	81.8%	55.99s	diffvar	81.3%	33.17s
17	GLCM [1,2]	diffvar	**81.9%**	24.10s	diffvar	81.7%	22.22s	diffvar	81.7%	90.01s	correl.	81.1%	33.91s
18	Gabor Var.	1	80.5%	2.27s	1	78.8%	2.20s	1	79.8%	9.97s	1	80.3%	8.80s

with their optimal parameters, can produce similar results (compare Tables III-VI - rows 1,5,6; 7,11,12; and 13,17,18). The per-layer analysis of Penobscot, however, suggests that, while Gabor produces similar results for the bottom layer – which corresponds to $\sim$63% of the cube – the other descriptors tend to work better for the upper layers.

The advantage of LBP over Gabor and GLCM, on the one hand, is performance: LPB has demonstrated to be substantially faster than the others – in average, 2.2 times faster than Gabor and seven times faster than GLCM [1,2]. On the other hand, because the feature vector of LBP is rather large (256 positions), it requires a much greater amount of memory. This memory demand may represent a bottleneck in case one desires to pre-compute features of several seismic cubes. Even still, it has demonstrated to be a robust descriptor, despite not being explored in the seismic classification literature.

Our experiments also demonstrated that using Euclidean Distance or Manhattan Distance is not significant, as both metrics can achieve similar results, with Manhattan being slightly better in some cases. The decision of using overlapping can slightly increase the accuracy in some scenarios. However, the processing time is much slower, as the number of tiles generated is inversely proportional to $(1 - overlap)^2$.

Regarding Gabor's optimal parameters, our results indicate that the best frequency value may depend on the image size; the same happens to LBP's radius parameter, as the larger the

TABLE VII
PER-LAYER RESULTS: NETHERLANDS INLINES (LEFT) AND CROSSLINES.

#	LBP	GLCM	Gabor
1	99.9%	100.0%	98.3%
2	81.4%	77.2%	69.6%
3	76.8%	68.4%	48.5%
4	-	-	-
5	59.0%	50.9%	42.2%
6	71.4%	64.4%	70.9%
7	71.9%	59.5%	61.4%
8	72.7%	76.3%	71.6%

#	LBP	GLCM	Gabor
1	97.6%	96.3%	91.9%
2	72.8%	68.1%	48.0%
3	14.2%	18.8%	20.0%
4	-	-	-
5	50.1%	43.8%	33.1%
6	78.9%	72.3%	77.0%
7	69.7%	66.7%	72.4%
8	53.1%	65.7%	72.7%

TABLE VIII
PER-LAYER RESULTS: PENOBSCOT INLINES (LEFT) AND CROSSLINES.

#	LBP	GLCM	Gabor
1	92.0%	92.0%	53.2%
2	58.5%	60.5%	48.0%
3	81.0%	87.4%	83.5%
4	52.2%	28.6%	26.2%
5	94.9%	87.6%	86.8%
6	59.8%	52.2%	69.3%
7	98.7%	98.8%	98.6%

#	LBP	GLCM	Gabor
1	82.3%	81.1%	52.0%
2	42.2%	31.8%	22.2%
3	60.5%	60.7%	55.5%
4	40.4%	27.9%	26.8%
5	69.2%	60.7%	57.1%
6	41.5%	38.0%	33.2%
7	92.4%	90.2%	90.2%

image dimension (*e.g.,* Penobscot crosslines), the higher the optimal value. The rotation invariance discussion of LBP also depends on the data, and our analysis per-category indicate that the default version is better for stratified data (*e.g.*, Figure 5 – top row), while the rotation-invariant version works better on chaotic data (*e.g.*, Figure 5 – bottom row).

In fact, the per-layer analysis is able to provide us an extra insight: we noticed that some categories are easier to produce good retrieval results than others, and this behavior tends to be consistent among the three descriptors.

VI. CONCLUSION

This paper reports several experiments on seismic image retrieval with the goal of assessing texture descriptors yet not addressed in the geological community – namely, LBP and Gabor Filters – in comparison with a more detailed analysis on GLCM – the most popular descriptor in this domain. Our findings indicate that LBP and Gabor Filters also have the potential to bring satisfying results, while LBP descriptor has the additional advantage of being computationally efficient.

Our experiments demonstrate that the parametrization of the descriptors is complex and may depend on the seismic data considered. However, our results indicate that, while `contrast` is a commonly used feature for GLCM-based analysis of seismic data, `dissimilarity` and `diffentropy` should also be investigated in detail. To the best of our knowledge, a similar investigation focusing on the best parameters values for seismic image retrieval have not been addressed before and such findings have not been previously reported.

In future work, we wish to polish the annotation of the seismic cubes used in this research, with the goal of providing it publicly, so other researchers from the Computer Vision community may be encouraged to intensify their studies on seismic image retrieval, segmentation, and classification.

ACKNOWLEDGMENT

The authors would like to thank dGB Earth Sciences for maintaining and sharing the Open Seismic Repository (OSR) and Dr. Marco Ferraz for the valuable contributions.

REFERENCES

[1] "The oilfield glossary: Where the oil field meets the dictionary," http://www.glossary.oilfield.slb.com/; Schlumberger, accessed: 2017 August 7th.

[2] R. M. Mitchum, P. R. Vail, and J. B. Sangree, "Seismic stratigraphy and global changes in sea level, part 6: stratigraphic interpretations of seismic reflection patterns in depositional sequences," in *Seismic Stratigraphy and Applications to Hydrocarbon Exploration: AAPG Memoir*, 1977, pp. 117–133.

[3] D. Gao, "Volume texture extraction for 3D seismic visualization and interpretation," *Geophysics*, vol. 68, no. 4, pp. 1294–1302, 2003.

[4] S. Chopra and V. Alexeev, "Applications of texture attribute analysis to 3D seismic data," *The Leading Edge*, vol. 25, no. 8, pp. 934–940, 2006.

[5] J. Dumay and F. Fournier, "Multivariate statistical analyses applied to seismic facies recognition," vol. 53, no. 9. Society of Exploration Geophysicists, 1988, pp. 1151–1159.

[6] B. P. West, S. R. May, J. E. Eastwood, and C. Rossen, "Interactive seismic facies classification using textural attributes and neural networks," *The Leading Edge*, vol. 21, no. 10, pp. 1042–1049, 2002.

[7] T. Zhao, V. Jayaram, A. Roy, and K. J. Marfurt, "A comparison of classification techniques for seismic facies recognition," *Interpretation*, vol. 3, no. 4, pp. SAE29–SAE58, 2015.

[8] A. Corradi, P. Ruffo, A. Corrao, and C. Visentin, "3D hydrocarbon migration by percolation technique in an alternate sand–shale environment described by a seismic facies classified volume," *Marine and Petroleum Geology*, vol. 26, no. 4, pp. 495–503, 2009.

[9] Z. Long, Y. Alaudah, M. A. Qureshi, M. Al Farraj, Z. Wang, A. Amin, M. Deriche, and G. AlRegib, "Characterization of migrated seismic volumes using texture attributes: a comparative study," in *2015 SEG Annual Meeting*. Society of Exploration Geophysicists, 2015, pp. 1744–1748.

[10] R. S. Ferreira, A. B. Mattos, E. V. Brazil, R. Cerqueira, M. Ferraz, and S. Cersosimo, "Multi-scale evaluation of texture features for salt dome detection," in *2016 IEEE International Symposium on Multimedia (ISM)*, Dec 2016, pp. 632–635.

[11] "Open seismic repository," https://opendtect.org/osr/index.php, accessed: 2017 August 7th.

[12] E. T. Pereira, H. M. Gomes, and J. M. d. Carvalho, "Integral local binary patterns: A novel approach suitable for texture-based object detection tasks," in *23rd SIBGRAPI Conference on Graphics, Patterns and Images*. IEEE Computer Society, 2010, pp. 201–208.

[13] B. C. Wallet and R. E. F. Pepper, "Using mathematical morphology in an attribute workflow to improve the interpretability of salt bodies in the gulf of mexico," in *SEG Technical Program Expanded Abstracts 2013*. Society of Exploration Geophysicists, 2013, pp. 1324–1328.

[14] K. Wilhelm and Y. Li, "Incorporation of seismic texture constraints to improve salt interface interpretation," in *SEG Technical Program Expanded Abstracts 2014*. Society of Exploration Geophysicists, 2014, pp. 1450–1454.

[15] A. Amin, M. Deriche, T. Hegazy, Z. Wang, and G. AlRegib, "A novel approach for salt dome detection using a dictionary-based classifier," in *SEG Annual Meeting*. Society of Exploration Geophysicists, 2015, pp. 1816–1820.

[16] T. Hegazy and G. AlRegib, "Texture attributes for detecting salt bodies in seismic data," in *SEG Technical Program Expanded Abstracts 2014*. Society of Exploration Geophysicists, 2014, pp. 1455–1459.

[17] T. Ojala, M. Pietikäinen, and D. Harwood, "A comparative study of texture measures with classification based on featured distributions," *Pattern recognition*, vol. 29, no. 1, pp. 51–59, 1996.

[18] R. M. Haralick, K. Shanmugam, and I. Dinstein, "Textural features for image classification," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. SMC-3, no. 6, pp. 610–621, Nov 1973.

[19] A. F. Costa, G. Humpire-Mamani, and A. J. M. Traina, "An efficient algorithm for fractal analysis of textures," in *2012 25th SIBGRAPI Conference on Graphics, Patterns and Images*, Aug 2012, pp. 39–46.