# A Training-free Classification Framework for Textures, Writers, and Materials

Radu Timofte[1]
http://homes.esat.kuleuven.be/~rtimofte
Luc Van Gool[1,2]
Luc.VanGool@esat.kuleuven.be

[1]ESAT-PSI-VISICS / IBBT
Catholic University of Leuven, Belgium
[2]D-ITET
ETH Zurich, Switzerland

**Abstract**

We advocate the idea of a training-free texture classification scheme. This we demonstrate not only for traditional texture benchmarks, but also for the identification of materials and of the writers of musical scores. State-of-the-art methods operate using local descriptors, their intermediate representation over *trained* dictionaries, and classifiers. For the first two steps, we work with pooled local Gaussian derivative filters and a small dictionary *not* obtained through training, resp. Moreover, we build a multi-level representation similar to a spatial pyramid which captures region-level information. An extra step robustifies the final representation by means of comparative reasoning. As to the classification step, we achieve robust results using nearest neighbor classification, and state-of-the-art results with a collaborative strategy. Also these classifiers need no training.

To the best of our knowledge, the proposed system yields top results on five standard benchmarks: 99.4% for CUReT, 97.3% for Brodatz, 99.5% for UMD, 99.4% for KTHTIPS, and 99% for UIUC. We significantly improve the state-of-the-art for three other benchmarks: KTHTIPS2b - 66.3% (from 58.1%), CVC-MUSCIMA - 99.8% (from 77.0%), and FMD - 55.8% (from 54%).

## 1 Introduction

Despite the fact that texture classification has received quite some attention, effective, highly accurate, and robust texture classification is still lacking. Some of the reasons are: the enormous variety of natural texture types, large intra-class variations coming from (non-)linear image projections, color changes due to varying illumination, corruptions like occlusion or texture blending, high demands on the hardware resources, and a need for systems that are easy to deploy [13, 19].

Standard texture classification systems aim at i) constructing a rich representation of the image and ii) providing an appropriate classification strategy. The representation typically entails local (texture) descriptors, similarity measures, aggregating strategies, and intermediate and global (image level) descriptors. The classification strategy usually adapts its metric to the representation and aims at fixing its flaws (*e.g.* growing the training sample pool by artificially generated, distorted samples, for reasons of robustness). Class models are then built using state-of-the-art classifiers.


http://dx.doi.org/10.5244/C.26.93

In the literature, the local texture descriptors are extracted over a sparse set of interest points [14], over a denser grid [13], or, more commonly, in every point in the image [20, 23, 31, 32]. To describe these points, the most successful filter banks are the rotation invariant basic image features (BIF) of [9] and MR8 of [32]. Alternatively, [14] uses modified SIFT and intensity domain SPIN images, [13] plain SIFT, [6, 20, 23] variants of LBP descriptors, [31] a grayscale image patch for its Joint descriptor and an MRF representation; [36] local fractals, and [18, 19] (sorted) random projections over small patches. Once the local descriptor types are fixed, there are two common approaches to further encode them: modeling images as histograms over a (learned) dictionary of features, or textons [32], or as signatures [13, 14] of features. For introducing spatial information into the global image descriptors, [15] proposes spatial pyramid matching (SPM) for a bag-of-words (BoW), an extension over [14]. The most common choices for similarity measures are the $\chi^2$ statistic [31], Bhattarcharyya distance [9], or Earth Mover's Distance [14] in conjunction with nearest neighbor classification. However, non-linear (kernel-based) SVMs proved to come with superior performance.

In this paper we propose a training-free multi-level texture classification framework. It combines the robustness and simplicity of local descriptors such as BIFs [9], layered, spatial information embedding similar to SPM [15] or a set of regions as in [13], the power of comparative reasoning [38], and state-of-the-art training-free classifiers [28]. The approach aims at addressing most of the problems encountered in texture classification, and exhibits the following advantages:

**1) No need for training, and thus data independence.** There is no need for learning a dictionary for the local descriptors (such as BIFs [9]), the system performs robustly with a fixed set of parameters on different textures, materials and handwritten score datasets.

**2) Robustness to intra-class variations.** The robustness is provided by the local descriptors, the layered, robustified representation, and the classifiers.

**3) Layered representation embedding spatial information.** Spatial information proved critical for object classification [15], and so it is for textures.

**4) Robustified representations by means of comparative reasoning.** The power of comparative reasoning (WTA hash [38]) enhances and robustifies the representations by adding resilience to perturbations in numeric values.

**5) Fast sparse and/or collaborative classification.** Lately, sparse and collaborative representation based classifiers performed best at various tasks such as face recognition [34, 40] or traffic sign recognition [28].

# 2 Proposed Framework

## 2.1 Local Texture Descriptor (BIF)

In our quest for the best local texture descriptor we are guided by the reported performances, dictionary size, the ease of training, complexity, and robustness to intra-class variance. We chose Basic Image Features (BIF) [9, 22] as our descriptor. Other tempting candidates are Local Binary Patterns (LBP) [23] and its derivations Extended LBP (ELBP) [20] and LBP Histogram Fourier (LBP-FH) [6], and the recently proposed Binary Gabor Pattern (BGP) [41], all taken at different scales.

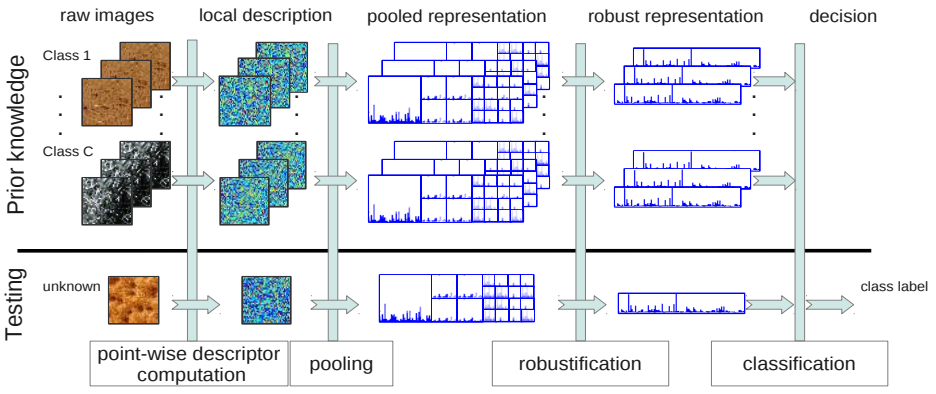


Figure 1: The scheme of our texture classification framework.

We briefly review BIF and its variants following closely the original work [9, 22]. Basic Image Features "are defined by a partition of the filter-response space (jet space) of a set of six Gaussian derivative (DtG) 2D filters" up to second order at some scale $\sigma$. Let $\{c_{ij} \in \mathbb{R} | \forall (i,j) \in \mathbb{N} \times \mathbb{N}, i+j \leq 2\}$ be the filter responses for the $i$-th derivative in $x$-direction and the $j$-th in $y$-direction, and $s_{ij} = \sigma^{i+j} c_{ij}$ their $\sigma$-scale normalizations. The jet space is partitioned further into seven regions, or BIFs, corresponding to distinct types of local image symmetry. Accordingly, the BIF scores and the assigned BIF type are represented by:

$$r = [\varepsilon s_{00}, 2\sqrt{s_{10}^2 + s_{01}^2}, \lambda, -\lambda, (\gamma+\lambda)/\sqrt{2}, (\gamma-\lambda)/\sqrt{2}, \gamma], \quad BIF_{code} = \arg\max_{k \in \{1,\cdots,7\}} \{r_k\} \quad (1)$$

where $\varepsilon \in \mathbb{R}$ is a threshold determining which features are considered flat, $\lambda = s_{20} + s_{02}$, and $\gamma = \sqrt{(s_{20} - s_{02})^2 + 4s_{11}^2}$.

By construction BIFs are rotation invariant. However, we can discretize orientations for BIF codes 2,5,6, and 7 (cf. [22]). We compute the quantifiable orientation $arctan(\frac{2s_{11}}{s_{02}-s_{20}})$ for BIF codes 5,6,7 and take $n$ discretized orientations, while for code 2, we take $atan2(s_{01}, s_{10})$ with $2n$ discretized orientations. In this way, the resulting Oriented Basic Image Features (oBIF) have $5n+3$ codes. Here, and in the original work [22], $n = 4$.

Thus, for a single scale the BIF descriptor has a dictionary of 7, while oBIF's dictionary has 23 entries. To create a more discriminative descriptor, [9] combines the descriptors at different scales. Empirically they found that the scale set $\sigma_i \in \{1,2,4,8\}$ is a good choice for BIFs, while for oBIFs [22], the number of scales is reduced to typically $\sigma_i \in \{1,4\}$. The multi-scale representations were called BIF-columns and oBIF-columns, respectively.

In practice, the regions where the flatness score (the first entry in the BIF dictionary) dominates, are discarded as non-informative, $c_{00}$ is fixed, and $\varepsilon$ is tuned. However, we fix $\varepsilon$ to a small value in all our experiments, $\varepsilon = 10^{-4}$. Taking the remaining dictionary entries, BIF with $p$ scales will generate $6^p$ distinct dictionary entries (for $p = 4$ there are 1296), while oBIF with $p$ scales will generate $22^p$ distinct dictionary entries (for $p = 2$ there are 484 oBIF entries). Regions are then described with BIF and oBIF histograms, of $6^p$ and $22^p$ bins, resp.

## 2.2 Multi-Level Pooled Representation (SPM, BoR)

The standard spatial pyramid matching (SPM) scheme has three or four pyramid levels with $\{1\times1, 2\times2, 4\times4\}$ or $\{1\times1, 2\times2, 4\times4, 8\times8\}$ pooling regions [15]. We add levels as long as the cell region size allows for meaningful histograms. The advantage of SPM is

that it brings spatial information into the image representation, while the downside is that it increases its size with an order equal to the total number of regions/cells. Since our features provide single codes per each image point, we consider as pooling method for a region the sum pooling (thus cumulating the histogram statistics).

Another recently proposed approach [13] uses multi-levels, similarly to SPM, for creating orderless region parts, allowing for overlap. Moreover, the regions are not concatenated into an image descriptor as in SPM but taken as a set of regions, called here Bag-of-Regions (BoR), sharing the same image label. The advantage of this approach is that it covers a much larger variance in scale, translation, rotation, viewpoint, illumination by enlarging the training pool. For the test image represented as BoR, the classification score is computed for each class and each region. At image level (or BoR level) the label is taken as the class with the best cumulative score over the BoR. For this multi-level BoR case we consider the same sum pooling strategies as in the case of SPM. The disadvantage of the method is that increases the size of the training pool and also requires a number of classification operations increased with an order equal to the total number of regions from BoR.

## 2.3 Robustified Representation - (WTA-hash)

Recently, the power of comparative reasoning [38] has been shown and a Winner Take All (WTA) hash technique proposed. WTA-hash is a sparse embedding method transforming the input feature space into binary codes. In the resulting space the Hamming distance closely correlates with rank similarity measures. The rank correlation measures are resilient to perturbations in numeric values and WTA-hash brings perturbation robustness to the original feature space representation. We use it to robustify the regions representations to numerical perturbations.

## 2.4 Sparse and Collaborative Classification - (SRC, CRC, INNC)

We pick classifiers that do not require parameter tuning for different datasets. The Sparse Representation Classifier (SRC) based on $l_1$-regularized least squares decomposition [34] and the Collaborative Representation Classifier (CRC) based on $l_2$-regularized least squares decomposition [40], that both define the state-of-the-art in face recognition. Whereas SRC usually is a slow method optimizing for each new input sample, CRC solves a ridge regression, for which one can precompute a projection matrix, such that for each new input sample the main operation is just a linear projection. Therefore, and because CRC tends to have on par performance with SRC for high-dimensional data, CRC is attractive [29]. The recently proposed Iterative Nearest Neighbors Classifier (INNC) [28], which is based on the fast approximated solution of an $l_1$-constrained least square decomposition, is another classifier equaling the performance of SRC and CRC, but at a much lower computational cost. However, INNC does not scale well with high-dimensional data.

A note about CRC is due here [29]. When the number of data samples exceeds data dimensionality, the computation of the projection matrix can be troublesome. A solution comes from the Moore-Penrose pseudoinverse [25]: one can work on the transposed data in order to compute the pseudoinverse. This allows CRC to scale well with either very large datasets or a very high dimensionality of the data. According the experiments from [28], working on a neighborhood of fixed size and not on the full data decreases the performance for CRC, as well as for SRC or INNC. Instead of resorting to neighborhoods [13], we use the pseudoinverse.

Table 1: Summary of texture, material, and score datasets used in our experiments.

| Dataset | Dataset Notation | Dataset Type | Image Rotation | Controlled Illumination | Scale Variation | Significant Viewpoint | Number Classes | Sample Size | Samples per Class | Samples in Total |
|---|---|---|---|---|---|---|---|---|---|---|
| CUReT [2, 8, 32] | $D^C$ | texture | ✓ | ✓ | | | 61 | 200×200 | 92 | 5612 |
| UIUC [3, 14] | $D^{UIUC}$ | texture | ✓ | | ✓ | ✓ | 25 | 640×480 | 40 | 1000 |
| UMD [4, 36] | $D^{UMD}$ | texture | ✓ | | ✓ | ✓ | 25 | 640×480 | 40 | 1000 |
| Brodatz [1, 14, 39] | $D^B$ | texture | | | | | 111 | 213×213 | 9 | 999 |
| KTHTIPS [8, 24] | $D^{KT}$ | texture | | ✓ | ✓ | | 10 | 200×200 | 81 | 810 |
| KTHTIPS2b [8, 24] | $D^{KT2b}$ | texture | | ✓ | ✓ | ✓ | 11 | 200×200 | 4(×9×12) | 4752 |
| FMD [17, 26] | $D^{FMD}$ | material | ✓ | | ✓ | ✓ | 10 | 512×384 | 100 | 1000 |
| CVCMUSCIMA [10, 11] | $D^{CM}$ | handwritten scores | | | | | 50 | ~2000×2000 | 20 | 1000 |

# 3 Experiments

## 3.1 Implementation details

The local **descriptors** we use are the BIFs and oBIFs (see Section 2.1). Our DtG filters use a $7 \times 7$ image patches for the reference scale $\sigma = 1$, since outside the filter values are negligible. The difficulty of obtaining filter responses at the borders is a known issue [9]. Authors have simply dropped filters falling outside or used circular padding [9]. We use a mirrored texture padding. Instead of adapting the basic DtG filters to a different scale, we prefer to rescale the image and run with the same filter bank with $7 \times 7$ image patch support. Since we work with scales larger than $\sigma = 1$, this approximation saves considerable computation time. After extracting the BIFs/oBIFs in the scaled image, the compound/combined descriptors are built based on the correspondence to the points at the reference scale $\sigma = 1$. Besides these approximations, we prefer to not tune the flatness threshold but fix it to the small value $\varepsilon = 10^{-4}$.

The **classifiers** we use are to a large extent training-free, i.e. do not change their parameters for different datasets. The classifiers used are the standard nearest neighbor classifier (NNC), the nearest mean class classifier (NMC), the sparse representation-based classifier (SRC) [34], the collaborative representation-based classifier with regularized least squares (CRC) [40], and the iterative nearest neighbor classifier (INNC) [28]. CRC has an algebraic solution and in practice can be much faster than the SRC solver (here we use the *feature sign* algorithm [16]), while INNC provides on par performance especially for low-dimensional data with an even lower computational time. If not mentioned otherwise, the regulatory parameter $\lambda$ is set to 0.001 for CRC, 0.05 for SRC, and 0.05 for INNC, which provide good trade-off performances in the literature [28, 34, 40].

Table 1 gives an overview of the **benchmarks** that we used. These comprise texture datasets covering various artificial and natural settings. In addition, we report on challenging datasets for materials and handwritten musical scores. For each benchmark we give the performance for the most common split of the data into training and testing parts. In Table 2, the number of training samples as used per each class for each dataset is given in brackets. If not mentioned otherwise, our result is the mean of the results obtained for 100 random generated partitions of a fixed size, as used in the literature. We always operate on the grayscale images. Also, the histogram descriptors are $l_1$ normalized and square-rooted prior to classification. This provides usually on par improvements with WTA-hash and binary rankings for NNC classification.

## 3.2 Texture Classification

We evaluate the performance of our classification framework on six public datasets: CUReT [2, 8, 32], Brodatz [1, 14, 39], KTHTIPS [8, 24], KTHTIPS2b [8, 24], UIUC [3, 14], and UMD [4, 36]. The main characteristics of these datasets are provided in Table 1.

**CUReT** is one of the most used texture benchmarks. As in previous works we use a sub-

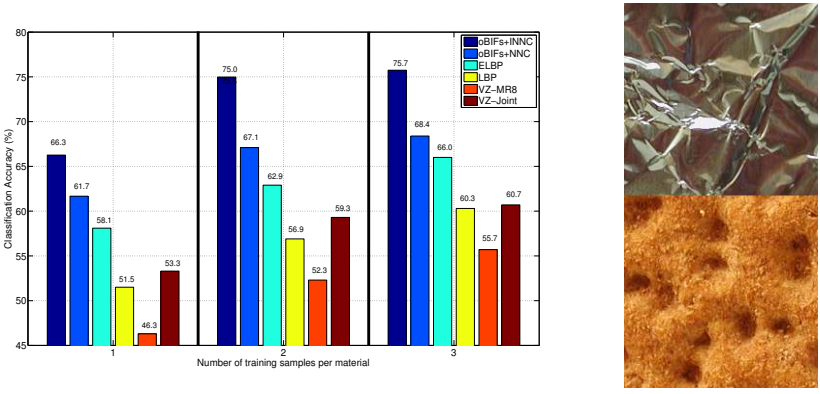


Figure 2: Comparison with state-of-the-art methods on KTHTIPS2b as reported by [20]. Note that Extended LBP (ELBP) [20] is evaluated using NN classifier.

set of the original dataset, containing 92 samples per each of the 61 classes [32]. With NNC, we achieve 96.5%, 97.2%, and 97.4% with oBIFs at one ($\sigma = 1$), two ($\sigma \in \{1,4\}$), and three scales ($\sigma \in \{1,4,8\}$), resp. With SRC, these go up to 97.5%, 98.7%, and 98.9% for the same settings. Rotations are present in CUReT and, as expected, using BIFs (meant for rotation invariance) at four scales (*sigma* $\in \{1,2,4,8\}$) provides us with a better performance: 98.0 with NNC (similar with the 98.6% performance reported in [9] for a multi-scaled classification strategy), 96.70% with CRC, 99.0% with SRC, and 98.9% with INNC. The BIF results already are on par and better than all reported results (Table 2) except for the Sorted Random Projections (SRP) approach [19] with 99.37%, which uses learning, however. If we add spatial information into the image descriptor by means of BoRs and use a spatial pyramid with 3 levels (1x1,2x2,3x3 or a total of 14 cells/regions) on top of BIFs we obtain 99.42% and 99.23% with CRC and NNC, resp.

**Brodatz** contains 111 texture images that are used to extract 9 samples as non-overlapping regions for each original texture image. For BIFs ($\sigma \in \{1,2,3,4\}$) we reach 91.8% with NNC, 93.6% with SRC, 94.1% with INNC, and 76.8% with CRC. These results are slightly worse than the SIFT-based set of regions approach [13] with 96.61% and the SRP approach [19] with 97.16% (Table 2). Using Oriented BIFs ($\sigma \in \{1,4\}$) we get slightly better results than with BIFs: 91.9% with NNC and 94.5% with INNC. Applying BoRs with 3 levels (1x1,2x2,3x3) for BIFs ($\sigma \in \{1,2,3,4\}$) brings us to top results: 95.5% with NNC and 96.5% with CRC, while with oBIFs ($\sigma \in \{1,4\}$) we obtain 96.37%-NNC and 97.26%.

**KTHTIPS** benchmarks scale variation handling in CUReT textures. For BIFs ($\sigma \in \{1,2,4,8\}$) we obtain 97.5% with NNC (behind 98.50% for BIFs with the multi-scale classification strategy of [9]), 93.4% with CRC, 98.6% with SRC, and 98.7% with INNC. Allowing for orientation variance in the local descriptor is beneficial since with oBIFs ($\sigma \in \{1,4\}$) the performance achieved is: 97.7% with NNC, 98.4% with CRC, 99.35% with SRC, and 99.1% with INNC. 99.35% is better than the top SRP result of 99.32% [19](Table 2).

**UIUC** has 25 classes with images under uncontrolled illumination. BIFs ($\sigma \in \{1,2,4,8\}$) yield the best reported performance with a multi-scale classification strategy explicitly dealing with scale changes [9]. These authors suggest potential improvements by using trained classifiers such as SVMs. We, on the contrary, propose training-free classifiers for better performance and do not explicitly deal with scale mismatch (except where we use the BoR strategy). For BIFs ($\sigma \in \{1,2,4,8\}$) we get 97.1% with NNC, 97.7% with CRC, 99.0% with SRC, and 98.6% with INNC. In this setting we already have a marginal improvement over

Table 2: Comparison of the best classification results for our approaches with those achieved by 22 state-of-the-art methods. The results are original, except for those reported in [39](*), in [20] (**), and in [41](***). In brackets is the number of training samples.

| | $D^C(46)$ | $D^B(3)$ | $D^{KT}(41)$ | $D^{UIUC}(20)$ | $D^{UMD}(20)$ | $D^{KT2b}(1)$ | $D^{CM}(10)$ | $D^{FMD}(50)$ |
|---|---|---|---|---|---|---|---|---|
| 1. Our Results | **99.42** | **97.26** | **99.35** | **99.01** | **99.54** | **66.26** | **99.80** | **55.78** |
| 2. VZ-MR8 [32] | 97.43 | | | | | 46.30(**) | | |
| 3. VZ-Joint [33] | 98.03 | 92.90(*) | 92.40(*) | 97.83 | | 53.30(**) | | |
| 4. Caputo *et al.* [8] | 98.46 | 95.00(*) | 94.80(*) | 92.00(*) | | | | |
| 5. Lezebnik *et al.* [14] | 72.50(*) | 88.15 | 91.30(*) | 96.03 | | | | |
| 6. Mellor *et al.* [21] | | 89.71 | | | | | | |
| 7. J.Zhang *et al.* [39] | 95.30 | 95.90 | 96.10 | 98.70 | | | | |
| 8. Varma and Ray [30] | | | | 98.76 | | | | |
| 9. Crosier and Griffin [9] | 98.60 | | 98.50 | 98.80 | | | | |
| 10. Xu *et al.*-MFS [36] | | | | 92.74 | 93.93 | | | |
| 11. Xu *et al.*-OTF [35] | | | | 97.40 | 98.49 | | | |
| 12. Xu *et al.*-WMFS [37] | | | | 98.60 | 98.68 | | | |
| 13. L.Liu *et al.*-CS [18] | 98.52 | 96.34 | 97.71 | 96.27 | 99.13 | | | |
| 14. L.Liu *et al.*-SRP [19] | 99.37 | 97.16 | 99.29 | 98.56 | 99.30 | | | 48.2 |
| 15. L.Liu *et al.*-ELBP [20] | 97.29 | | | | | 58.10 | | |
| 16. Kong and Wang [13] | | 96.61 | 99.32 | | 99.32 | | | |
| 17. PRIP02 [10] | | | | | | | 77.00 | |
| 18. TUA03 [10] | | | | | | | 76.60 | |
| 19. L.Zhang *et al.*-BGF [41] | 98.70 | | | | | | | |
| 20. Ojala *et al.*-LBP [23] | 98.10(***) | | | | | 51.50(**) | | |
| 21. Ahonen *et al.*-LBP-FH [6] | | | | | | 54.60 | | |
| 22. C.Liu *et al.*-aLDA [17] | | | | | | | | 44.60 |
| 23. Hu *et al.* [12] | | | | | | | | 54.00 |

the best reported result [9] (99.0% vs. 98.8%, Table 2). Note that on this dataset, oBIFs instead of BIFs considerably worsen the results. oBIFs ($\sigma \in \{1,4\}$) yield 73.8% with NNC, 91.2% with CRC, 93.3% with SRC and 84.5% with INNC.

**UMD** is similar to UIUC, but with textures from the wild. BIFs ($\sigma \in \{1,2,4,8\}$) provide robust top performance: 99.1% with NNC, 99.16% with CRC, 99.54% with SRC, 99.28% with INNC. The previous top results were reported for the trained approaches based on SRP (99.32% [19]) and on SIFT with BoRs and Locally constrained CRC (99.32%-[13], Table 2). oBIFs ($\sigma \in \{1,4\}$) achieve also top results: 97.1% with NNC, 98.7% with CRC, 98.9% with SRC, and 98.4% with INNC.

**KTHTIPS2b** is a very challenging texture benchmark. It contains only 4 samples per class, but each sample is further represented by 108 images (combination of 9 scales, 3 viewpoints and 4 illumination conditions). The samples – with all their 108 images - are distributed over either the training or the testing set. For validation, for each class we randomly select $k$ samples for training and the remaining $4-k$ for testing. In our experiments $k \in \{1,2,3\}$, and for each $k$ we have 10 trials. The results are depicted in Fig. 2 and Table 2. We significantly improve the state-of-the-art performance. For oBIFs ($\sigma \in \{1,2,4\}$) and $k=1$ we have: 61.67% with NNC, 64.73% with CRC, and 66.26% with INNC.

## 3.3 Material Recognition

The Flickr Material Database (**FMD**) [17, 26](Table 1) is a challenging dataset designated for material recognition, which is closely related to texture classification but at the same time different in many ways. In our experiments, we use 10 trials of 50 randomly picked training and testing images per class.

We consider that for materials an intermediate local representation is useful, especially when a large training set is available as in the case of FMD. For this purpose, we consider BoR representations, where the image is split into $l^2$ non-overlapping cells/regions at level $l$. For each cell the histogram is computed from all its points. All cell histograms from training are taken individually in the training pool, while for the test images the cell histograms are the Bag-of-Regions image representations. Note that the WTA-hash robustification is

Table 3: Spatial granularity affects material recognition (BoRs+NNC,%) on FMD.

| number cells | 1x1 | 2x2 | 3x3 | 4x4 | 5x5 | 6x6 | 7x7 | 8x8 | $1^2+2^2+4^2+8^2$ | $7^2+8^2$ | $1^2+\cdots+8^2$ |
|---|---|---|---|---|---|---|---|---|---|---|---|
| BIFs, $\sigma \in \{1\}$ | 21.1 | 23.6 | 23.3 | 22.6 | 22.4 | 22.7 | 22.5 | 22.4 | 32.1 | 32.4 | 34.5 |
| BIFs, $\sigma \in \{1,4\}$ | 24.8 | 31.8 | 35.5 | 37.0 | 38.1 | 39.0 | 39.7 | 40.4 | 41.8 | 42.1 | 42.5 |
| BIFs, $\sigma \in \{1,2,4,8\}$ | 26.9 | 32.8 | 34.5 | 36.2 | 36.4 | 37.4 | 38.0 | 39.5 | 40.6 | 41.7 | 42.1 |
| oBIFs, $\sigma \in \{1\}$ | 21.2 | 27.2 | 33.5 | 34.1 | 36.4 | 37.7 | 38.4 | 40.6 | 41.2 | 41.4 | 42.0 |
| oBIFs, $\sigma \in \{1,4\}$ | 29.3 | 31.5 | 34.4 | 33.9 | 34.2 | 35.4 | 37.0 | 38.1 | 41.3 | 41.5 | 42.6 |
| oBIFs, $\sigma \in \{1,2,4\}$ | 29.9 | 30.1 | 31.6 | 32.5 | | | | | | | |

able to improve the performance, but not significantly, especially when compared to the $l_1$ normalized and square-rooted descriptors that we use in our settings. The classification is done in the Naive Bayes framework where the decision is built using the distance of each cell histogram to the labeled training pool of cell samples [7, 13, 28]. The impact of the number of cells and scales is evaluated. Table 3 depicts the results of our system with BIFs and oBIFs using an NNC, which leads to the standard Naive Bayes Nearest Neighbor classifier [7]. If we only use the whole image (rather than a pyramid) the best results are achieved using oBIFs with $\sigma \in \{1,2,4\}$: 29.9%-NNC,48.6%-CRC,49.1%-SRC,41.5%-INNC.

On this material recognition task, increasing the number of cells and thus refining the spatial representation, improves the results. Thus, for BIFs with 2 scales, while at the coarsest setting, one cell for the whole image, we obtain 25% with NNC and 47% with CRC; at the finest setting, with 64 cells, the recognition improves to 40.5% with NNC and 54.5% with CRC. Combining regions of different sizes improves the classification, thus putting together the cells from an 8-level pyramid pushes the performance of BIFs with $\sigma \in \{1,4\}$ and NNC to 42.5%. The larger the number of cells/regions, usually the better the performance (as shown also in [13]), but at the cost of slowing down the computation.

## 3.4 Writer Identification

For writer identification we use the recently introduced **CVCMUSCIMA** [10, 11] dataset. This dataset has 50 writers and 20 music scores pages per (Table 1). The music pages do not exhibit significant rotations or scale changes and, moreover, the writers used the same pen. The staff lines were removed and the images were binarized, thus the input images are quite clean and of low scale, rotation, and viewpoint variance. In the original form the dataset is proposed with a single split such that for each writer there are 10 randomly selected pages for training and the remaining 10 pages are for testing.

With oBIF features with 2 scales (1,4) and without SPM or BoR, we achieve 50.4% identification performance with NNC, 60.4% with the nearest mean class descriptor classifier (NMC), 74.2% with INNC ($\lambda = 0.02$), 98.6% with SRC (feature sign [16] $\lambda = 0.001$), and 99.8% with CRC ($\lambda = 0.001$). With BIF features at 4 scales (1,2,4,8), without SPM and BoR, we get only 38.2% identification performance with NNC, 49.8% with NMC, 54.6% with INNC ($\lambda = 0.02$), 71.2% with SRC (feature sign $\lambda = 0.001$), and 83.4% with CRC ($\lambda = 0.001$). As expected, because of the lack of rotations in the dataset, BIF features, despite of using more scales and larger descriptors (1296 vs. 484), perform poorer than the oBIFs.

Because of the excellent results obtained with oBIFs and sparse or collaborative classifiers – 99.8% vs. 77%, the best reported result so far [10] – we proceed further to validate the system with different train/test partitions of CVCMUSCIMA. Thus, we consider splits with 1 training page per writer, up to 10 training pages, and the remaining pages being the testing pages. We randomly generate 100 trials for each such partition type and report the mean performance (Fig. 3) for each considered framework setting.

**Learning embeddings.** The usual applications aim at efficiency and best performance,

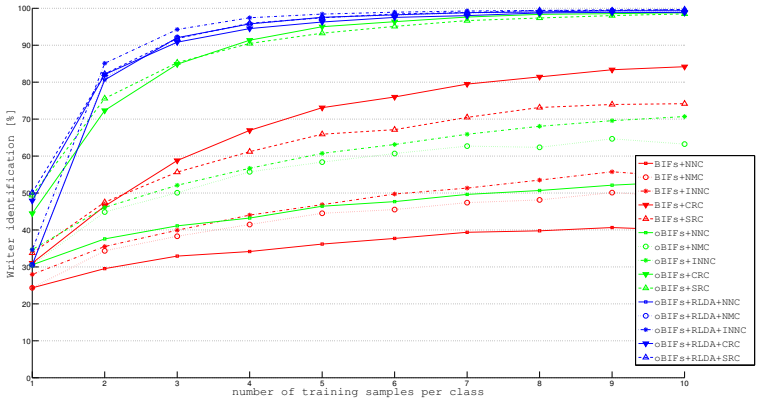


Figure 3: Writer identification versus training samples on CVCMUSCIMA dataset.

usually by learning the most from the training data. While up to now we considered a purely training-free texture classification system, here we prove that adding discriminative learning in any of the components of our proposed system is likely to improve the overall performance on the considered datasets. For this, we use Linear Discriminant Analysis (LDA), as in [27], to project the oBIFs features from their original 484-dimensional space into a lower, more discriminative 49-dimensional space (we further $l_2$-normalize them before the classification step). LDA has as regularization parameter $\lambda$, which we put to 0.001. The results (Fig. 3) show a considerable improvement over the training-free variants: 86.6%-INNC vs. 76.7%-SRC for 2 training scores per writer.

## 3.5 Discussion

We have shown that training-free pipelines can outperform several state-of-the-art texture classification methods. We populated the proposed scheme of Fig. 1 with state-of-the-art components, each robust with respect to the choice of their parameters. Due to a lack of space, we have analyzed just a handful of components and combinations. We are conservative in our experiments, in that further finetuning would be possible, i.e. we only went up to the point where the methods would outperform or get on par with the state-of-the-art, training-based methods. Moreover, we did not report all the experimental results for the combinations that were considered. Furthermore, we only report results using the basic image features (BIFs) and its variants as local descriptors [9, 22], SPM [15] with one level and a simple Bag-of-Regions model [13] as intermediate representations, and classifiers such as NNC, SRC [34], CRC [40], or INNC [28].

As a matter of fact, we were somewhat surprised by the strong performance of these methods, regardless of the dataset and/or task. Within the context of our classification scheme, for object classification one would want to deploy rotation variant features and additional, spatial constraints. Further research can try and robustify the image/patch descriptors by means of promising techniques such as WTA-hash, recently introduced in [38]. Also, we believe that starting from our classification framework and adding training at any level can improve performance further. We validated this in the writer identification task by using LDA projections for the image descriptors. Significant improvements were achieved, both in identification accuracy and in running time (the feature dimensionality is greatly reduced).

# 4 Conclusions

This paper proposes an effective training-free texture classification system. It uses (o)BIF columns for pixel level description, an SPM like multi-level image representation, the power of collaborative reasoning, and sparse or collaborative classifiers. The proposed approach is computationally simple. To a large extent, it also is training-free and data-independent. The system was validated for texture classification, material recognition and writer identification on several benchmarks. We obtain results that are at least on-par, but sometimes substantially better than state-of-the-art performance.

**Acknowledgments.** This work was partly supported by the IWT/SBO ALAMIRE project and the European Commission FP7 ICT-269980 AXES project.

# References

[1] Brodatz texture dataset. Available at: http://www.ux.uis.no/~tranden/brodatz.html.

[2] CUReT 92 subset, gray images. Available at: http://www.robots.ox.ac.uk/~vgg/research/texclass/data/curetgrey.zip.

[3] UIUC texture dataset. Available at: http://www-cvr.ai.uiuc.edu/ponce_grp/data/.

[4] UMD texture dataset. Available at: http://www.cfar.umd.edu/~fer/website-texture/texture.htm.

[5] *The Twenty-Third IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2010, San Francisco, CA, USA, 13-18 June 2010*, 2010. IEEE.

[6] Timo Ahonen, Jiri Matas, Chu He, and Matti Pietikäinen. Rotation invariant image description with local binary pattern histogram fourier features. In Arnt-Børre Salberg, Jon Yngve Hardeberg, and Robert Jenssen, editors, *SCIA*, volume 5575 of *Lecture Notes in Computer Science*, pages 61–70. Springer, 2009. ISBN 978-3-642-02229-6.

[7] Oren Boiman, Eli Shechtman, and Michal Irani. In defense of nearest-neighbor based image classification. In *CVPR*. IEEE Computer Society, 2008.

[8] Barbara Caputo, Eric Hayman, Mario Fritz, and Jan-Olof Eklundh. Classifying materials in the real world. *Image Vision Comput.*, 28(1):150–163, 2010.

[9] Michael Crosier and Lewis D. Griffin. Using basic image features for texture classification. *International Journal of Computer Vision*, 88(3):447–460, 2010.

[10] Alicia Fornés, Anjan Dutta, Albert Gordo, and Josep Lladós. The ICDAR 2011 music scores competition: Staff removal and writer identification. In *ICDAR*, pages 1511–1515, 2011.

[11] Alicia Fornés, Anjan Dutta, Albert Gordo, and Josep Lladós. CVC-MUSCIMA database for writer identification - ICDAR/GREC 2011 competition. Available at: http://www.cvc.uab.es/cvcmuscima/competition/, 2011.

[12] Diane Hu, Liefeng Bo, and Xiaofeng Ren. Toward robust material recognition for everyday objects. In *Proceedings of the British Machine Vision Conference*, pages 48.1–48.11. BMVA Press, 2011. ISBN 1-901725-43-X. http://dx.doi.org/10.5244/C.25.48.

[13] Shu Kong and Donghui Wang. Multi-level feature descriptor for robust texture classification via locality-constrained collaborative strategy. *CoRR*, abs/1203.0488, 2012.

[14] Svetlana Lazebnik, Cordelia Schmid, and Jean Ponce. A sparse texture representation using local affine regions. *IEEE Trans. Pattern Anal. Mach. Intell.*, 27(8):1265–1278, 2005.

[15] Svetlana Lazebnik, Cordelia Schmid, and Jean Ponce. Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In *CVPR (2)*, pages 2169–2178. IEEE Computer Society, 2006. ISBN 0-7695-2597-0.

[16] Honglak Lee, Alexis Battle, Rajat Raina, and Andrew Y. Ng. Efficient sparse coding algorithms. In *NIPS*, 2006.

[17] Ce Liu, Lavanya Sharan, Edward H. Adelson, and Ruth Rosenholtz. Exploring features in a bayesian framework for material recognition. In *CVPR* DBL [5], pages 239–246.

[18] Li Liu, Paul W. Fieguth, and Gangyao Kuang. Compressed sensing for robust texture classification. In Ron Kimmel, Reinhard Klette, and Akihiro Sugimoto, editors, *ACCV (1)*, volume 6492 of *Lecture Notes in Computer Science*, pages 383–396. Springer, 2010. ISBN 978-3-642-19314-9.

[19] Li Liu, Paul W. Fieguth, David A. Clausi, and Gangyao Kuang. Sorted random projections for robust rotation-invariant texture classification. *Pattern Recognition*, 45(6): 2405–2418, 2012.

[20] Li Liu, Lingjun Zhao, Yunli Long, Gangyao Kuang, and Paul W. Fieguth. Extended local binary patterns for texture classification. *Image Vision Comput.*, 30(2):86–99, 2012.

[21] Matthew Mellor, Byung-Woo Hong, and Michael Brady. Locally rotation, contrast, and scale invariant descriptors for texture analysis. *IEEE Trans. Pattern Anal. Mach. Intell.*, 30(1):52–61, 2008.

[22] Andrew J. Newell and Lewis D. Griffin. Multiscale histogram of oriented gradient descriptors for robust character recognition. In *ICDAR*, pages 1085–1089. IEEE, 2011. ISBN 978-1-4577-1350-7.

[23] T. Ojala, M. Pietikäinen, and T. Mäenpää. Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *PAMI*, 2002.

[24] A.T. Targhi E. Hayman B. Caputo P. Mallikarjuna, M. Fritz and J.-O. Eklundh. The KTH-TIPS and KTH-TIPS2 databases. Available at: `http://www.nada.kth.se/cvap/databases/kth-tips/`, 2006.

[25] Roger Penrose. A generalized inverse for matrices. *Mathematical Proceedings of the Cambridge Philosophical Society*, 51:406–413, 1955. doi: 10.1017/S0305004100030401.

[26] Rosenholtz R. Sharan, L. and E. H Adelson. Flickr material database (FMD). Available at: http://people.csail.mit.edu/celiu/CVPR2010/FMD/, 2009.

[27] Radu Timofte and Luc Van Gool. Sparse representation based projections. In *BMVC*, 2011.

[28] Radu Timofte and Luc Van Gool. Iterative nearest neighbors for classification and dimensionality reduction. In *CVPR*, 2012.

[29] Radu Timofte and Luc Van Gool. Weighted collaborative representation and classification of images. In *ICPR*, 2012.

[30] Manik Varma and Debajyoti Ray. Learning the discriminative power-invariance trade-off. In *ICCV*, pages 1–8. IEEE, 2007.

[31] Manik Varma and Andrew Zisserman. Texture classification: Are filter banks necessary? In *CVPR (2)*, pages 691–698. IEEE Computer Society, 2003. ISBN 0-7695-1900-8.

[32] Manik Varma and Andrew Zisserman. A statistical approach to texture classification from single images. *International Journal of Computer Vision*, 62(1-2):61–81, 2005.

[33] Manik Varma and Andrew Zisserman. A statistical approach to material classification using image patch exemplars. *IEEE Trans. Pattern Anal. Mach. Intell.*, 31(11):2032–2047, 2009.

[34] John Wright, Allen Y. Yang, Arvind Ganesh, Shankar Sastry, and Yi Ma. Robust face recognition via sparse representation. *PAMI*, 31(2), February 2009.

[35] Yong Xu, Si-Bin Huang, Hui Ji, and Cornelia Fermüller. Combining powerful local and global statistics for texture description. In *CVPR*, pages 573–580. IEEE, 2009. ISBN 978-1-4244-3992-8.

[36] Yong Xu, Hui Ji, and Cornelia Fermüller. Viewpoint invariant texture description using fractal analysis. *International Journal of Computer Vision*, 83(1):85–100, 2009.

[37] Yong Xu, Xiong Yang, Haibin Ling, and Hui Ji. A new texture descriptor using multifractal analysis in multi-orientation wavelet pyramid. In *CVPR* DBL [5], pages 161–168.

[38] Jay Yagnik, Dennis Strelow, David A. Ross, and Ruei-Sung Lin. The power of comparative reasoning. In Dimitris N. Metaxas, Long Quan, Alberto Sanfeliu, and Luc J. Van Gool, editors, *ICCV*, pages 2431–2438. IEEE, 2011. ISBN 978-1-4577-1101-5.

[39] Jianguo Zhang, Marcin Marszalek, Svetlana Lazebnik, and Cordelia Schmid. Local features and kernels for classification of texture and object categories: A comprehensive study. *International Journal of Computer Vision*, 73(2):213–238, 2007.

[40] Lei Zhang, Meng Yang, and Xiangchu Feng. Sparse representation or collaborative representation: Which helps face recognition? In *ICCV*, 2011.

[41] Lin Zhang, Zhiqiang Zhou, and Hongyu Li. Binary gabor pattern: An efficient and robust descriptor for texture classification. In *ICIP*, 2012.