

Robust Texture Recognition Using Credal Classifiers

Giorgio Corani
giorgio@idsia.ch

Alessandro Giusti
alessandrogiusti@idsia.ch

Davide Migliore
davide@idsia.ch

Juergen Schmidhuber
juergen@idsia.ch

IDSIA, SUPSI and University of Lugano
Switzerland

www.idsia.ch

Abstract

Texture classification is used for many vision systems; in this paper we focus on improving the reliability of the classification through the so-called imprecise (or credal) classifiers, which suspend the judgment on the doubtful instances by returning a set of classes instead of a single class. Our view is that on critical instances it is more sensible to return a reliable set of classes rather than an unreliable single class. We compare the traditional naive Bayes classifier (NBC) against its imprecise counterpart, the naive credal classifier (NCC); we consider a standard classification dataset, when the problem is made progressively harder by introducing different image degradations or by providing smaller training sets. Experiments show that on the instances for which NCC returns more classes, NBC issues in fact unreliable classifications; the indeterminate classifications of NCC preserve reliability but at the same time also convey significant information, reducing the set of possible classes (on most critical instances) from 24 to some 2-3.

1 Introduction

Texture classification is used in a large number of fields, ranging from industrial applications to remote sensing and biomedical engineering. The problem is traditionally solved by first computing texture descriptors (i.e. feature vectors) from image data and then performing supervised or unsupervised classification.

In this paper we adopt the Local Binary Patterns descriptor [1], and focus on the classification problem, showing that the reliability of classification can be improved by suspending the judgment, namely returning a set of classes rather than a single class, on the instances whose classification is most doubtful. We call *indeterminate* the classifications containing more than one class.

Recently, there has been in machine learning increasing attention towards classifiers able to return more classes on doubtful instances: see for instance the Naive Credal Classifier (NCC) [2], which is an extension towards robustness of the well-known naive Bayes and

conformal predictions [13]. We focus in particular on NCC; it shares part of the assumptions of naive Bayes but, being based on imprecise probability, it returns more classes on the doubtful instances. The theory of imprecise probability can be seen as a generalizing the traditional Bayesian theory towards robustness [14]; it provides theoretical reasons and practical algorithms to work with a *set* of probability distributions rather than with a single probability distribution. Therefore, classifiers based on imprecise probability (*credal classifiers*) computes, when classifying an instance, a *set* of probability distributions; if the most probable class cannot be consistently identified across all distributions, the credal classifier returns more classes. In general, imprecise probability leads to weaker but more robust conclusions than traditional Bayesian probability.

A conservative approach to classification, which suspends the judgment on the doubtful instances, can generally constitute an advantage for machine vision applications. For instance, assuming that surface defects are categorized for taking appropriate countermeasures, an indeterminate classification containing two classes may trigger a machining able to fix either of the two possible problems, or call for manual evaluation if no such treatment exists.

By comparing naive Bayes and NCC, we show that indeterminate classifications can improve the reliability of classification of textures. In particular, NBC is almost randomly guessing among the classes returned by NCC, when NCC returns an indeterminate classification; we see as more sensible, on the critical instances, returning a reliable set of categories rather than a single but unreliable category. The ability to return indeterminate classifications is particularly useful when the classification problem is challenging: in practical applications this is often the case, possibly because of the actual appearance of the different texture classes, or due to several factors such as limited size of the training set, small, noisy or defocused images. We thoroughly compare NBC and NCC in texture classification, applying to the standard OUTEX [15] dataset various disturbances, namely a) reduction of the training set size; b) crop; c) noise; d) blur.

2 Previous Work

A large amount of literature deals with texture recognition. In particular, several different descriptors have been proposed, including Haralick features [16], Gabor filters [17] and histograms of texture elements (textons) [18, 19, 20]. In this paper we consider Local Binary Patterns [21], which is one of the most used approaches in practical applications [22]; it has the advantage of simple implementation and fast performance, and has been successfully used in a wide variety of different fields.

Many extensions to the basic LBP methodology have been developed, in order to increase its discrimination abilities. As shown in [23], more powerful descriptors can be obtained by concatenating multiple LBP feature vectors, and possibly by jointly considering local variance measures $\text{VAR}_{P,R}$, which also summarize useful texture information, or by taking into account multiple radii for achieving scale invariance. Instead, in this paper we are considering the relatively simple case of the $\text{LBP}_{P=16,R=2}^{\text{riu}2}$ rotation-invariant mapping (see following Section), which provides a manageable but still powerful descriptor.

Given a descriptor, one may apply different methods for classification; for instance, Nearest-Neighbor, Support Vector Machines (SVM) [24] or boosting [25]; however, but it is common to focus the attention on the effectiveness of the descriptor rather than on the capabilities of the classifier. In this paper, we concentrate on the latter, and show the advantages

of using classifiers based on imprecise probabilities.

3 Texture Feature Extraction: LBP

For a given image, the feature vector is computed as the histogram of the $LBP_{P=16,R=2}^{riu2}$ codes associated to its pixels. For a given pixel p , such a code is defined by considering a $P = 16$ neighbors placed in a circular chain centered on p , with a radius $R = 2$ pixels, and starting with the neighbor displaced by 2 pixels on the right; the gray level of the 12 neighbors with non-integer coordinates is computed by bilinear interpolation of adjacent pixels. The LBP code for pixel p is therefore computed by comparing its gray level to the gray level of its neighbors. The resulting feature vector is 18-dimensional, which corresponds to 17 histogram bins representing uniform rotation-invariant codes, plus 1 bin where nonuniform patterns are accumulated.

We use such feature vectors to solve a supervised texture classification problem. We consider a dataset of images representing a number of different textures; a feature vector is computed for each image in the dataset, and associated with the known class label. A classifier is then trained on a subset of the dataset (training set), and its accuracy evaluated on the remaining part of the dataset (testing set).

3.1 Effect of image degradations

Literature on texture classification is mostly concerned on advocating new descriptors which provide improved accuracy over standard data sets, or on describing applications for texture classification algorithms. In practical applications, imaging conditions are often suboptimal, and various conditions often concur to make the classification problem harder.

In this paper we consider several such conditions (see Figure 1 and evaluate how they affect the proposed classifier; in particular, we consider:

- texture classification on small image patches (“*crop*”), which occurs when the image area depicting the texture to be classified is limited in size.
- effect of *noise* on texture classification; noisy images are often acquired from microscopes in biomedical settings (i.e. photon-limited imaging), or when extremely short exposure times are needed for imaging fast-moving subjects, as in many industrial applications. The effect of noise on texture classification is particularly marked, as the LBP descriptor relies on point-wise comparisons of gray levels of neighboring pixels.
- effect of *blur*, often affecting images from microscopes or macroscopes, which have a very shallow depth of field.
- limited size of the training set, either because ground truth (i.e. class labels) may be difficult or expensive to obtain, or because few instances are available, possibly only for some of the classes (i.e. rare defects).

4 Bayesian probability and Imprecise Probability

A traditional criticism of Bayesian methods is the need for specifying a prior distribution. In fact, prior information is generally difficult to quantify; moreover one often prefers to let the

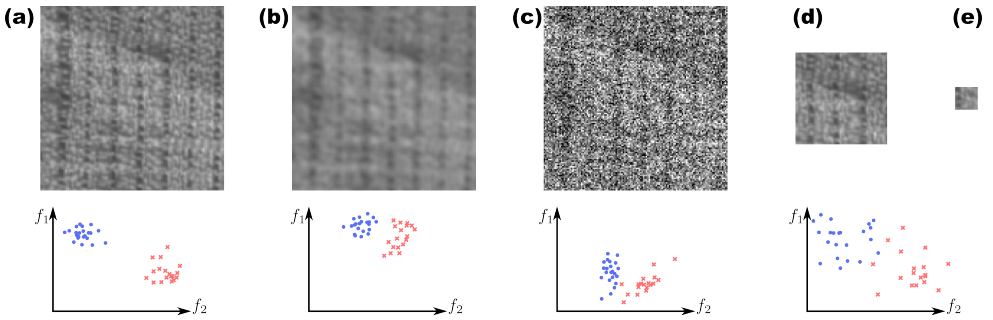


Figure 1: Some of the considered degradations affecting the classification problem: original image (a), blur ($\sigma = 4$ pixels (b)), noise ($\text{snr}=0.12$ (c)), and smaller size of the available patch (crop = $1/4$ (d); crop = $1/64$ (e)). Second row illustrates effect on feature vectors for instances in two classes, by visualizing the projection of feature vectors in two dimensions obtained by PCA; the instances are well separable in the degraded images; blur and noise shift the balance of features: in particular, noise biases $\text{LBP}_{P=16,R=2}^{\text{riu}2}$ histograms towards bin 18 (representing nonuniform patterns), whereas blur biases histograms towards the central bins, corresponding to uniform patterns. Smaller image patches (“crop”) increase the spread of feature vectors around the cluster centers, whose position is unchanged; this is explained as the normalized histogram averages data from smaller samples (i.e. fewer pixels), thus increasing variance of feature vectors without affecting their expected value.

data “speak by themselves” without introducing prior beliefs, which might be subjective. A simple solution is to set a *uniform* prior, which assigns a priori the same probability to all the outcomes regarded as possible in the problem under consideration. The uniform prior looks non-informative but does not properly model a condition of ignorance a priori, as clarified in the following example, inspired to [15].

Let us consider a bag containing blue marbles and red marbles; no drawings have been made from the bag. Which is the probability of getting a blue marble in the next draw? According to the uniform prior, one assigns to both colors probability 0.5 of appearing in the next draw; this might look like a non-informative statement, but underlies the very strong assumption that the bag contains an equal number of red and blue marbles. In the subjective interpretation of probability, it means that one is disposed to pay an amount of money 0.5 to enter a gamble whose reward is 1 if the prediction is correct and 0 if the prediction is wrong; according to the uniform prior, one is equally disposed to bet the same amount of money (0.5) on both colors.

Yet, we cannot exclude the bag to be severely biased towards one color or another; in fact, if we are ignorant about the content of the bag, the only reliable statement is the proportion of red (or blue) marbles to be comprised between 0 and 1. Using imprecise probability [15], the situation can be better modelled by a *set* of prior distributions, letting vary the prior probability of blue between a *lower probability* (\underline{P}) of 0 and an *upper probability* (\overline{P}) of 1, and doing the same for red. This is called a *vacuous* prior model. The subjective interpretation of imprecise probability [15] is that \underline{P} is the *maximum* amount of money that one is disposed to pay to enter the above gamble. Since both blue and red have $\underline{P} = 0$, imprecise probability advises to *not* bet at all. If one is ignorant, this is clearly a more sensible option than being equally disposed to bet on both colors.

If the variable about which to made inference is discrete, as in the case of classification, the prior is usually chosen to be a Dirichlet. The mathematical tool which manages a set of priors is the Imprecise Dirichlet Model (IDM) [15].

To clarify the difference between Bayesian estimate and IDM, we show the posterior probabilities in the case of the bag of marbles. Let us consider a Dirichlet prior parameterized as $Dir(\alpha_{blue}, \alpha_{red})$; the parameter $\alpha := \alpha_{red} + \alpha_{blue}$ is the *equivalent sample size* and can be thought of as a number of *hidden draws*. Moreover, α_{blue} and α_{red} can be considered as the percentage of blue and red marbles extracted in the hidden draws. The uniform prior is typically defined by $\alpha=1, \alpha_{blue} = \alpha_{red} = 0.5$.

Assuming to have observed n draws, within which the blue and the red marble have been drawn n_{blue} and n_{red} times, the posterior probability of red is for instance:

$$P(red|\boldsymbol{\alpha}, \mathbf{n}) = \frac{n_{red} + \alpha_{red}}{n + \alpha}$$

where $\boldsymbol{\alpha}$ is the vector containing α, α_{red} and α_{blue} and \mathbf{n} is the vector containing n, n_{red} and n_{blue} .

Under the IDM, α_{blue} and α_{red} vary between 0 and 1, thus modelling a condition of prior-ignorance. The posterior upper probability of blue is obtained for $\alpha_{blue} = 1$, which is equivalent to assume all the hidden draws to be blue; the posterior lower probability of blue is obtained for $\alpha_{blue} = 0$, which is equivalent to assume all the hidden draws to be *not blue*:

$$\underline{P}(red|\boldsymbol{\alpha}, \mathbf{n}) = \frac{n_{red}}{n + \alpha} \quad \bar{P}(red|\boldsymbol{\alpha}, \mathbf{n}) = \frac{n_{red} + \alpha}{n + \alpha}$$

Learning with imprecise probability can be seen as updating the set of priors with the likelihood, yielding a *set* of posteriors (*generalized Bayes' rule*). The posterior probability of a certain event lies therefore within an interval, rather than being a fixed value as in traditional probability. On large data sets, imprecise probability would take the same decisions as the traditional Bayesian approach; in fact, in this case the choice of the prior is not influential. Yet, on small data sets, imprecise probability leads to peculiar behaviors; for instance, classifiers based on imprecise probabilities (*credal classifiers*) return more classes when the most probable class cannot be consistently identified within the set of posteriors.

5 Naive Bayes and Naive Credal Classifier

The naive Bayes classifier (NBC) is a simple yet effective classifier, which *naively* assumes the stochastic independence of the features given the class; despite this simplistic assumption, it is however generally competitive, especially on small and medium sized data sets. Because of the naive assumption, NBC tends to be too confident in its predictions, assigning often an unrealistic high probability to the most probable class [9].

However, NBC and Bayesian classifiers in general can sometimes issue *prior-dependent* classifications, i.e., the most probable class sometimes changes under different priors. This might be acceptable if it has been possible to carefully elicit the prior modelling domain knowledge, which is however uncommon; thus, prior-dependent classification are generally unreliable.

The naive credal classifier (NCC) [2] addresses this problem by specifying a set of priors through the IDM. More in particular, it allows the marginal prior probability of each class to

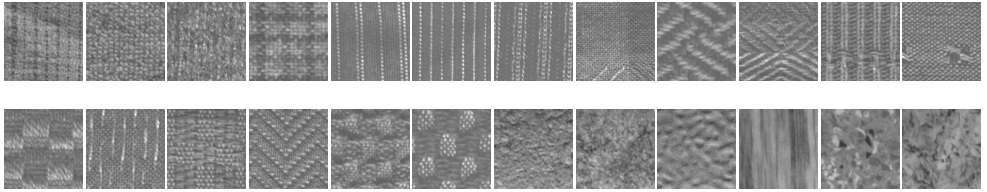


Figure 2: Exemplars of the 24 classes in the considered dataset.

vary between 0 and 1 and the conditional prior probability of each value of the features given each class to identically vary between 0 and 1. Overall, NCC adopts a set of joint priors over features and classes; in fact, it considers *all* the Dirichlet priors which are admissible by NBC. The set of priors is then turned into a set of posteriors by the generalized Bayes' rule, once the likelihood has been computed from the data \mathbf{d} ; at this point, NCC is trained.

Assuming to have k features, NCC classifies an instance when the values $\mathbf{f} = \{f_1, \dots, f_k\}$ of the features are known. While traditional classifiers return the most probable class, obtained after conditioning on both \mathbf{d} and \mathbf{f} in the posterior, NCC identifies the *non-dominated* classes; this is achieved by running pairwise comparison between the classes. Let us consider an example with two classes c_1 and c_2 ; if the set of posteriors contains a distribution such that $P(c_1|\mathbf{d}, \mathbf{f}) > P(c_2|\mathbf{d}, \mathbf{f})$ but also a distribution such that $P(c_2|\mathbf{d}, \mathbf{f}) > P(c_1|\mathbf{d}, \mathbf{f})$, both c_1 and c_2 are non dominated; instead, if $P(c_1|\mathbf{d}, \mathbf{f}) > P(c_2|\mathbf{d}, \mathbf{f})$ on every posterior, c_1 *dominates* c_2 . Therefore there can be one or multiple non-dominated classes, yielding respectively *determinate* or *indeterminate* classifications. When there is a single non-dominated class, this corresponds to the class identified by NBC; if instead there are more non-dominated classes, they contain the class returned by NBC and the classification issued by NBC is prior-dependent. On prior-dependent instances, NCC returns a less informative but more robust answers than NBC. Indeterminate classifications are less frequent on large data sets, because the choice of the prior becomes less influential as more data are available; yet, prior-dependent instances can be present even on large data sets, as shown in Section 6.

Exhaustive experiments [2] have shown that NBC is generally unreliable on the instances indeterminately classified by NCC; in fact, the performance of NBC is constituted by a mix of good accuracy on the *non* prior-dependent instances and much lower accuracy on the prior-dependent ones. Instead, NCC generally preserves its reliability also on prior-dependent instances thanks to indeterminate classifications. Note that indeterminate classification can in fact be informative, if for instance few classes are returned out of many possible ones.

An open source implementation of NCC is available [3].

6 Experimental Results

We first evaluate the classifiers on the original data set; then, we perform a set of experiments, to assess the sensitivity of their performance to (a) the size of the training set, (b) noise, (c) cropping and (d) blur. To evaluate the classifiers, we perform 10 runs of 10-folds cross-validation; during cross-validation, we discretize numerical features through supervised discretization [2]. We did not perform feature selection, as our aim is comparing NBC and NCC, rather than fine tuning the classifiers for maximum performance.

The data set contains 4500 instances, 18 features and 24 classes (see Figure 2); NBC

achieves a satisfactory accuracy of about 92% (for a comparison, SVM achieves 92.5%). However, thanks to NCC we can separately analyze what happens on the instances which are respectively prior-dependent and non prior-dependent. In particular, the performance of NBC is constituted by a mix of reliable classifications on the instances which are *not* prior-dependent and unreliable classifications on the prior-dependent ones. When the instance is *not* prior-dependent, NBC and NCC return the same class achieving accuracy 94%; when instead the instance is prior-dependent, NBC drops to 56% accuracy while NCC remains around 85% accuracy, returning on average 2.5 classes. The accuracy of NCC on the instances determinately classified is measured as the % of instances where the set of classes contains the correct one. That is why we need also to report the number of classes returned by NCC to fully characterize its performance. However, we can see that indeterminate classification are nevertheless informative; they reduce the number of classes from 24 to 2-3 in most cases, for instance. Prior-dependent instances amount to some 5%.

To understand how prior-dependent instances are possible despite the large data set, we have to consider that the estimate of the conditional probability of each feature given the class requires to estimate some 240 parameters (24 classes * 10, where 10 is roughly the number of intervals in which the features gets discretized), each corresponding to a specific combination of value of the class and of the feature; as some of such combinations rarely appear in the data set, their estimate can be indeed sensitive on the prior which is set.

However, with NBC (and with Bayesian classifiers in general) there is no way of understanding whether a certain instance is prior-dependent. An approach which could be implemented trying to mimic the behavior of NCC is to set a *rejection option* on NBC, outputting more classes if the probability of the class does not exceed a certain threshold. Yet, as discussed in Section 5, NBC generally estimates an extremely high posterior probability for the returned class; most often, such probability exceeds 95%, thus making ineffective the rejection option. Instead, NCC detects some 2% of prior-dependent instances, even among those classified by NBC with probability higher than 95%. In fact, an instance is less likely to be prior-dependent as the probability computed by NBC for the most probable increases; but such association is not deterministic [□]: some non-prior dependent instances are classified by NBC with a relatively small probability for the most probable class, while some prior-dependent ones are instead classified by NBC with high probability. Therefore, the prior-dependency analysis performed by NCC is much more sophisticated than the rejection option.

We have then performed a series of experiments, comparing NBC and NCC at varying size of the training set (Fig.3, third row). We have generated data sets of size comprised between 50 and 1000, by down-sampling the original data set. The down-sample has been performed in a *stratified* way, namely the proportion of the classes is equal between the original data set and the down-sampled ones. For the sample sizes 50 and 100, we built 20 data sets each; for the sample sizes 100, 250, 500 and 1000 we built 5 data sets each. On each data set, we run cross-validation; we then averaged the results obtained on data sets of the same size. At each size of the training set, NBC reliably classifies the instances which are not prior-dependent, but its accuracy sharply decreases on the prior-dependent ones, as shown in Fig.3(g). Thanks to indeterminate classifications, NCC is instead much more accurate than NBC on the prior-dependent instances, as can be seen by comparing Fig.3(g) and Fig.3(h). Moreover, NCC becomes steadily more determinate with increasing size of training set, as shown in Fig.3(i); for instance, if the data set increases from 50 to 100, the proportion of indeterminate classifications decreases from 79% to 33%; at the same time, the average number of non-dominated classes decreases from 13 to 7.

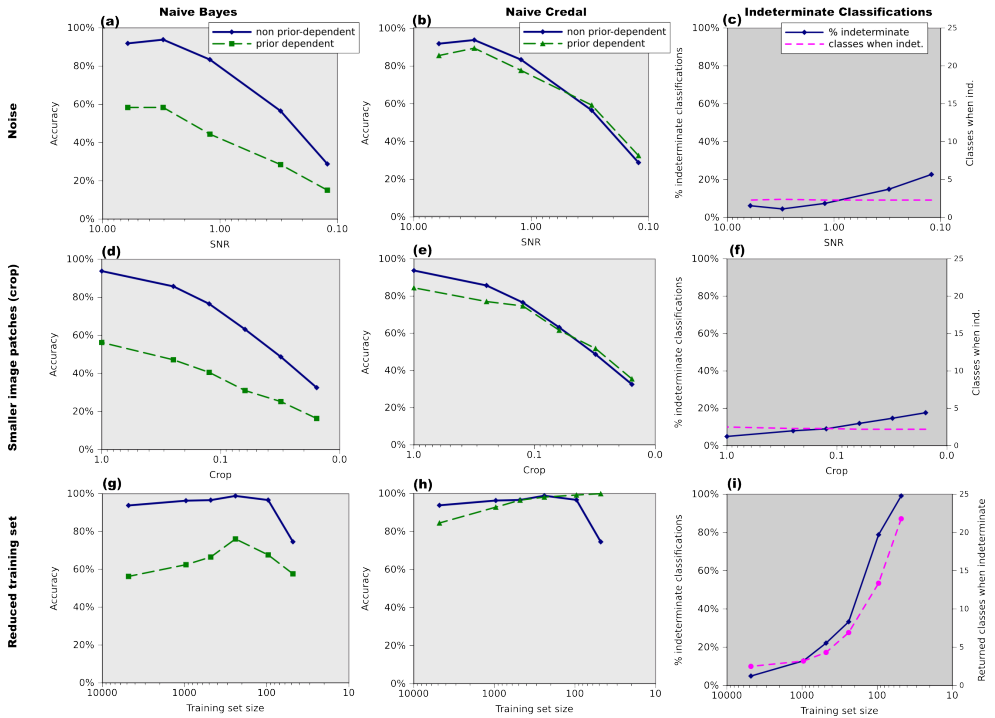


Figure 3: Experimental Results on the OUTEX dataset. First and second columns show the accuracy of NBC and NCC, respectively, on instances which are prior dependent (light dashed) and non prior-dependent (dark). Third column shows the percentage of indeterminate classifications (dark) and the average number of classes returned by NCC when indeterminate (light dashed). First row shows the effect of increasing noise, expressed as decreasing SNR in log scale. Second row shows the effect of smaller image patches for classification, where 1 represents the original 128x128 pixel images. Third row shows the effect of smaller training sets.

We then assessed sensitivity to cropping (Fig.3, second row) and noise (Fig.3, first row), by applying to the full data set respectively a crop ratio varying between 1 (full 128x128 pixel image patches) and 0.016 (1/64 of the original area, i.e. 16x16 image patches, see Figure 1) or a SNR ratio ranging from 6 (no added noise on the OUTEX images) and 0.12 (noise with a variance of 0.05, with images in the [0-1] intensity range, and a signal standard deviation of 0.11 – i.e. an extremely challenging problem). Both experiments confirm that NBC undergoes a sharp drop of accuracy on the prior-dependent instances, as shown in Fig.3(a) and 3(d); on the contrary, NCC remains reliable on prior-dependent instances thanks to indeterminate classifications, as shown in Fig.3(b) and 3(e). Of course, the overall performance of both NBC and NCC is deteriorated by low crop ratios or high noise levels; for instance, when the crop ratio decreases from 1 to 0.016, the accuracy of NBC and NCC on the *non*-prior dependent instances decreases from 93% to 32%, as shown in Fig.3(e); when the SNR decreases from 6 to 0.12, the accuracy of NBC and NCC on the *non*-prior dependent instances decreases respectively from 92% to 28%, see Fig.3(b). Interestingly, although NCC is not designed to become more indeterminate with noisy data, this is what actually

happens; for instance, the proportion of indeterminate classification rises from 10% to 18% when the cropping ratio is reduced from 0.125 to 0.016, or from 8% to 23% when the SNR decreases from 6 to 0.12. A possible explanation is that NBC computes less extreme probabilities under cropping or noise and, as already explained, instances classified with lower probability by NBC are more likely to be prior-dependent.

As final experiment, we assessed sensitivity to blur, which we varied between 0 (no blur) and 6 (Gaussian blur with $\sigma = 6$ pixels). Both NBC and NCC turned out to be quite insensitive to blur: for instance, the average accuracy of NBC decreased from 92% to 88% varying the blur from 0 to 6; at any level of blur the proportion of prior-dependent instances is constantly around 5%. Also the remaining indicators (accuracy of NBC and NCC on the instances which are prior-dependent and not prior-dependent) show only minor fluctuations with blur.

Note that the experiments were performed by applying the degradations both to the training and testing sets; this explains the robustness to blur, as the classifier learns to capture the large-scale intensity variations in the textures (*macrotecture*), as the high-frequency components are canceled by blurring. If the classifier is trained on the original images, the performance on blurred samples quickly drops, as the feature vectors themselves are heavily affected by blur – although somewhat predictably.

7 Conclusions

In this paper we considered the application of the NCC imprecise classifier on a standard texture classification problem. Experimental results show how the imprecise classifier successfully recognizes difficult instances to be classified, thus returning a set of possible classes, ultimately providing a more robust behavior than the traditional NBC. We believe that many applications using texture classification might benefit from such behavior.

As the present approach did not require any special adaptation of the texture descriptor, it can be effortlessly adapted to the vision tasks including a classification step, such as face recognition, gesture recognition, key-point following using random ferns, and many others.

Future experiments might be conducted with more advanced credal classifiers, no longer based on the naive assumption.

Acknowledgments

Work partially supported by the Swiss NSF grant n. 200021-118071/1 and funded through the EU Project IM-CLeVeR (Intrinsically Motivated Cumulative Learning Versatile Robots), Contract number: FP7-ICT-IP-231722.

References

- [1] Barbara Caputo, Eric Hayman, and P. Mallikarjuna. Class-specific material categorisation. *Computer Vision, IEEE International Conference on*, 2:1597–1604, 2005. ISSN 1550-5499.
- [2] G. Corani and M. Zaffalon. Learning Reliable Classifiers from Small or Incomplete

- Data Sets: the Naive Credal Classifier 2. *Journal of Machine Learning Research*, 9: 581–621, 2008.
- [3] G. Corani and M. Zaffalon. JNCC2: The Java Implementation Of Naive Credal Classifier 2. *Journal of Machine Learning Research*, 9:2695–2698, 2008.
- [4] U. M. Fayyad and K. B. Irani. Multi-interval Discretization of Continuous-valued Attributes for Classification Learning. In *Proc. of the 13th International Joint Conference on Artificial Intelligence*, pages 1022–1027, San Francisco, CA, 1993. Morgan Kaufmann.
- [5] D.J. Hand and K. Yu. Idiot’s Bayes-Not So Stupid After All? *International Statistical Review*, 69(3):385–398, 2001.
- [6] R.M. Haralick, K. Shanmugam, and I.H. Dinstein. Textural features for image classification. *IEEE Transactions on systems, man and cybernetics*, 3(6):610–621, 1973.
- [7] A. Hoogs, R. Collins, R. Kaucic, and J. Mundy. A common set of perceptual observables for grouping, figure-ground discrimination, and texture classification. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 458–474, 2003.
- [8] Thomas Leung and Jitendra Malik. Representing and recognizing the visual appearance of materials using three-dimensional textons. *Int. J. Comput. Vision*, 43(1):29–44, 2001. ISSN 0920-5691. doi: <http://dx.doi.org/10.1023/A:1011126920638>.
- [9] Timo Ojala, Topi Maenpaa, Matti Pietikainen, Jaakko Viertola, Juha Kylliäinen, and Sami Huovinen. Outex - new framework for empirical evaluation of texture analysis algorithms. pages 701–706, 2002.
- [10] Timo Ojala, Matti Pietikainen, and Topi Maenpaa. Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. pages 701–706, 2002.
- [11] M. Pietikainen. Image analysis with local binary patterns. *Image Analysis*, pages 115–118, 2005.
- [12] Ruiz, L. A. and Fdez-Sarria, A. and Recio, J. A. Texture feature extraction for classification of remote sensing data using wavelet decomposition: a comparative study. pages 1682–1750, 2004.
- [13] G. Shafer and V. Vovk. A tutorial on conformal prediction. *The Journal of Machine Learning Research*, 9:371–421, 2008.
- [14] M. Varma and A. Zisserman. A statistical approach to texture classification from single images. *International Journal of Computer Vision*, 62(1):61–81, 2005.
- [15] P. Walley. Inferences from multinomial data: learning about a bag of marbles. *J. R. Statist. Soc. B*, 58(1):3–57, 1996.
- [16] G. Zhang, X. Huang, S. Li, Y. Wang, and X. Wu. Boosting local binary pattern (lbp)-based face recognition. *Advances in biometric person authentication*, pages 179–186, 2004.