2010 IEEE International Conference on Robotics and Automation
Anchorage Convention District
May 3-8, 2010, Anchorage, Alaska, USA

Independent Component Analysis and Bayes' Theorem for Robotics and Automation

Richard E. Hudson and Wyatt S. Newman, *Senior Member, IEEE*

Electrical Engineering and Computer Science Department
Case Western Reserve University
Cleveland, OH 44106, USA
{richard.hudson & wyatt.newman}@case.edu

Abstract **–Independent Component Analysis (ICA) provides a pragmatic means to perform pattern classification using Bayes' Theorem. Use of ICA with Bayes' Theorem is reviewed and illustrated with examples from classification of images. It is described how ICA with Bayes can create a pattern-classification system that is trainable merely by presenting examples. A specific algorithmic approach is advocated, and demonstrations of its versatility and ease of use show how this technique offers promise for industrial applications.**

Index Terms – pattern classification, image processing, ICA, industrial inspection, visual inspection, Bayes' Theorem.

I. Introduction

The ability to classify images into groups is frequently useful, as in industrial inspection or character recognition. Often, it is useful to classify parts of an image into separate regions, such as "grass" and "road". Since its introduction (see [7]), Independent Component Analysis (ICA) has been the focus of much research. The property of statistical independence reduces the complexity of probabilistic computations, making it suitable for use with Bayes' Theorem (see [15]). Furthermore, an ICA transform generates weights with a Gaussian probability distribution, which further simplifies probability calculations. This paper describes the use of ICA and Bayes' Theorem for image classification and presents illustrative examples, including application to industrial inspection.

One criticism of existing visual industrial inspection methods is their inability to easily adapt to changing circumstances, for example, producing a new color of a product. This fear of high ongoing costs in terms of robustness and additional expert involvement is a barrier to more widespread use of automated inspection. Use of ICA/Bayes holds promise of offering the ability to retrain an inspection system solely by presenting examples of pre-classified training images. Such a system would be suitable for shop-floor application, as illustrated herein by a specific industrial application: Quality Control inspection of cellulose kitchen sponges.

While ICA/Bayes is not novel in the research literature, there are virtually no industrial applications reported. This presentation aims to spotlight the opportunity for application of ICA/Bayes for industrial applications, specifically because of its versatility, its performance, and its compatibility with a means for simple training by example. While a specialized algorithm within any narrow application domain will inevitably outperform a general-purpose algorithm, a more versatile algorithm will have greater applicability to industry. Further, systems that can be used by shop-floor personnel are more desirable than those that require expert interventions to code or tune. For these reasons, ICA/Bayes pattern classification should be exploited in industry.

A tutorial review of ICA/Bayes is presented here, followed by three application examples. Importantly, the same algorithm is applied to all three domains, and training is performed by presenting the system with examples. Ease of use and generality are emphasized.

II. Background on Independent Component Analysis and Bayes' Theorem

A. Independent Component Analysis

Independent Component Analysis (ICA) is a linear transform, similar to Principal Component Analysis (PCA) (See [14]). In PCA, the objective is to find an orthogonal set of basis vectors to decompose a pattern and to order those vectors by their relative importance, thus offering a means for a lower-order approximate representation. In contrast, the goal with ICA is to find a set of basis vectors (not necessarily orthogonal) that are chosen to span the pattern space while minimizing mutual information between any two basis vectors.

A linear ICA decomposition can be expressed as:

$$\mathbf{y} = \mathbf{M}\mathbf{x} \tag{1}$$

where **x** is the original image, **M** is the ICA transform, and **y** is the representation of **x** in the new ICA space. By definition, an ICA transform **M** results in **y** having the following key property:

$$P(\mathbf{y}) = P(y_1) \cdot P(y_2) \cdots P(y_N) \tag{2}$$

where the notation *P(a)* indicates the probability of result *a* occurring, y_i is component *i* of vector **y** and *N* is the number of components in **y**. Furthermore, component values of **y** have (approximately) Gaussian probability distributions with, by ICA convention, variance 1. That is,

$$P(\mathbf{y}_i) = N(\mathbf{y}_i, 1) \tag{3}$$

where the notation *N(a,b)* indicates a Gaussian probability curve with mean *a* and variance *b*.

For a given set of vectors **X**, **M** is not guaranteed to exist, but if it does, it is unique to within sign differences and reordering of components. That is, if ICA is run twice on a collection of vectors **X**, and ICA transforms $\mathbf{M}_1$ and $\mathbf{M}_2$ are found, then the corresponding transformations:

$$\mathbf{p} = \mathbf{M}_1\mathbf{x} \tag{4}$$

and

$$\mathbf{q} = \mathbf{M}_2\mathbf{x} \tag{5}$$

result in vectors **p** and **q** that are equivalent, but for possible sign changes or component reordering. As a further constraint, ICA can only find **M** if no more than one component of the vectors of **X** has a Gaussian probability distribution. (See [9].)

Various algorithms for finding ICA decompositions exist; an excellent overview of various methods (and their justifications) can be found in [10]. The present work uses the fastICA method found in [8].

B. ICA in image processing

ICA requires vectors as inputs. As images are inherently two-dimensional arrays of data, a first step is defining how to convert from the one to the other. As it turns out, because order of components is not important for ICA (as it seeks statistical independence among all components), as long as the conversion from image space to vector space and back is consistent, component ordering is irrelevant.

There are, however, still multiple ways to apply ICA to image processing tasks. In particular, the question comes up of how much of an image to present to ICA for processing. In some literature, such as [1, 5], an image of size MxN is treated as a single whole, fed entirely into ICA as a single vector of length MN. The resulting independent components are also of length MN, and the resultant transform is size MNxMN.

One contrasting method, used in [2, 4, 6], is to divide the image into small regions, referred to here as *thumbnails*, of LxW pixels, where L<M and W<N. Typical values of L and W would be 10 to 20. These thumbnails are each treated as a strung-out vector of length LW. (This process of generating vectors from images will be referred to as *vectorizing.*) Classification of the collection of thumbnails may be further analyzed at subsequent hierarchical layers of processing in the resulting reduced-order space. Experimentally, we have found that the choice of thumbnail size can affect performance results, and that an optimal size may exist.

C. Bayes' Theorem

Bayes' Theorem is a rigorous probability formula, the proof of which can be found in [12]. Bayes' Theorem may be stated as

$$P(a \mid b) = \frac{P(b \mid a) \cdot P(a)}{P(b)} \tag{6}$$

where *P(a|b)* is the probability of *a* given *b*. *P(a)* is often referred to as the *a-priori* probability of *a*, the likelihood of *a* before we know whether or not *b* occurred. Use of Bayes' Theorem is often problematic because of the cost in terms of both time and information requirements. Notably, joint probability distributions are typically unknown, and their use in Bayes' formula is computationally unwieldy. However, in ICA space, the statistical independence properties of ICA components makes it practical to invoke Bayes' Theorem.

III. A Method for Using ICA and Bayes' Theorem for Image Classification

Returning to the problem statement, assume we wish to analyze an image (of MxN pixels, which may be a thumbnail of a larger image) by assigning it to one out of the P groups from group set G. These groups are defined according to the problem task at hand. For example, a character recognition task might have each group be associated with a particular letter of the alphabet, for a total of 27 groups, the letters plus "none". The proposed method is to first use an ICA transform to convert the image into a representation space where probabilities are easy to determine, then use Bayes' Theorem to find to which group an image is most likely to belong. This method is divided into a training part, which is done once, followed by the testing part, which is done for each image to be classified. The training steps are as follows:

A. Training

1) Assume availability of a set of training images that are representative of the problem domain. (In [11], for example, scenes from nature are chosen as the problem domain.) Each of the groups within set G should be represented. Vectorize these images (see section II B, above) and then run ICA (i.e. derive a set of statistically independent basis vectors) on the entire set. This will create an ICA transform for that particular image set. Assuming ICA converges, (see II.A, above,) the resulting basis vectors will satisfy the properties of statistical independence and Gaussian component probability distribution. An assumption in subsequent use is that these ICA properties hold for "related" images drawn from the same domain.

2) Next, manually create groups of member images. These need not be the same as used in part 1. For

example, the group “grass” would consist of thumbnails showing a WxH region of grass.

3) For each group, perform the ICA transform on each member and find $P(\mathbf{y} \mid \mathbf{x} \subset G_p)$. A variety of methods to do this exist, as found in [3, 13]. For this paper, each component of **y** within any group is assumed to have Gaussian probability density. Thus each group can be characterized by identifying vector $\overline{\mathbf{y}}$ containing the mean of the *j* training data vectors, $\mathbf{g}_j$

$$\overline{\mathbf{y}} = \frac{1}{S}\sum_{j=1}^{S}\mathbf{g}_j \tag{7}$$

where S is the number of training samples. The variance of each group is given by:

$$\text{var}(\mathbf{y}_i) = \frac{1}{S}\sum_{i=1}^{S}(\mathbf{y}_i - \overline{\mathbf{y}}_i)^2 \tag{8}$$

If the ICA members of any one group do not have approximately Gaussian distribution, this may often be addressed by subdividing that group into subgroups whose members form a more Gaussian distribution.

If the group membership distribution is truly representative of actual population densities, then an *a-priori* estimate of group membership probability ($P(\mathbf{x} \subset G_p)$) can be estimated from the training data classifications. If the training data does not provide credible *a-priori* categorization probabilities, then a default *a-priori* assumption is that an image is equally likely to be a member of each of the category groups.

This concludes the training process. Next is described how to exploit ICA decomposition in Bayes’ Theorem to classify patterns.

B. ICA-based classification

Given a means to decompose patterns in terms of statistically independent vectors, one can easily utilize such decomposition in Bayes’ Theorem to perform pattern classification, as follows.

1) For each pattern to be classified, represent this pattern in terms of ICA decomposition. This can be performed on the entire collection of **X** (vectorized) images as a single matrix multiply,

$$\mathbf{Y} = \mathbf{MX}, \tag{9}$$

resulting in **Y**, the transformed images. Each column vector within **Y** is the ICA-space representation of the corresponding column vector in **X**.

For each $\mathbf{y} \subset \mathbf{Y}$, determine the probability that its corresponding **x** is in group G_p.

$$P(\mathbf{x} \subset G_p \mid \mathbf{y}) = \frac{P(\mathbf{y} \mid \mathbf{x} \subset G_p) \cdot P(\mathbf{x} \subset G_p)}{P(\mathbf{y})} \tag{10}$$

2) Repeat the above computation for each $p \leq P$, the number of groups. Evaluation of (10) is typically problematic. However, the value of ICA decomposition, for which each component is statistically independent of all other components, is that (10) becomes computable. Using (2), (10) can be written as

$$P(\mathbf{x} \subset G_p \mid \mathbf{y}) = P(y_1 \mid \mathbf{x} \subset G_p) \cdot P(y_2 \mid \mathbf{x} \subset G_p) \cdots P(y_n \mid \mathbf{x} \subset G_p) \cdot \frac{P(\mathbf{x} \subset G_p)}{P(y_1) \cdot P(y_2) \cdots P(y_n)}. \tag{11}$$

Image **x** most likely belongs to the group with the highest probability $P(\mathbf{x} \subset G_p|\mathbf{y})$.

IV. Applications

Use of ICA and Bayes’ Theorem for image classification is demonstrated in three examples: a simplified character-recognition task, scene preprocessing in the context of mobile robots, and an industrial inspection task.

A. ICA for character recognition

An example pattern-classification application is illustrated here using a simplified character-recognition task. Theintent here is to illustrate the use and versatility of ICA-based pattern classification. (We note that, within the particular domain of OCR, specialized algorithms would be expected to outperform ICA. However, it is irrelevant in this illustration that the patterns of interest happen to be characters). The present example consists of recognizing whether a given pattern is A, B, C, D, E or blank. The letters are fixed in font (10 point Arial/Western as created in MS Paint) and have known location and (zero) rotation within a 20x20 image. Salt and pepper noise was then added, where some fraction (between 10% and 30%) of the pixels, chosen at random, were replaced with either a black or a white pixel, with equal probability of each. The algorithm was trained using full-sized thumbnails (20x20) on correctly-classified training data for each of the groups. The algorithm presumed no *a priori* information regarding knowledge of characters, but was merely trained on the examples.

To test performance, conditions ranged from 0 to 100% of the pixels being corrupted with noise, tested in 5% increments of 36 examples each. This task was then given to the algorithm and to 10 test subjects. The test images are shown in Fig. 1, and the classification results are summarized in Fig. 2.

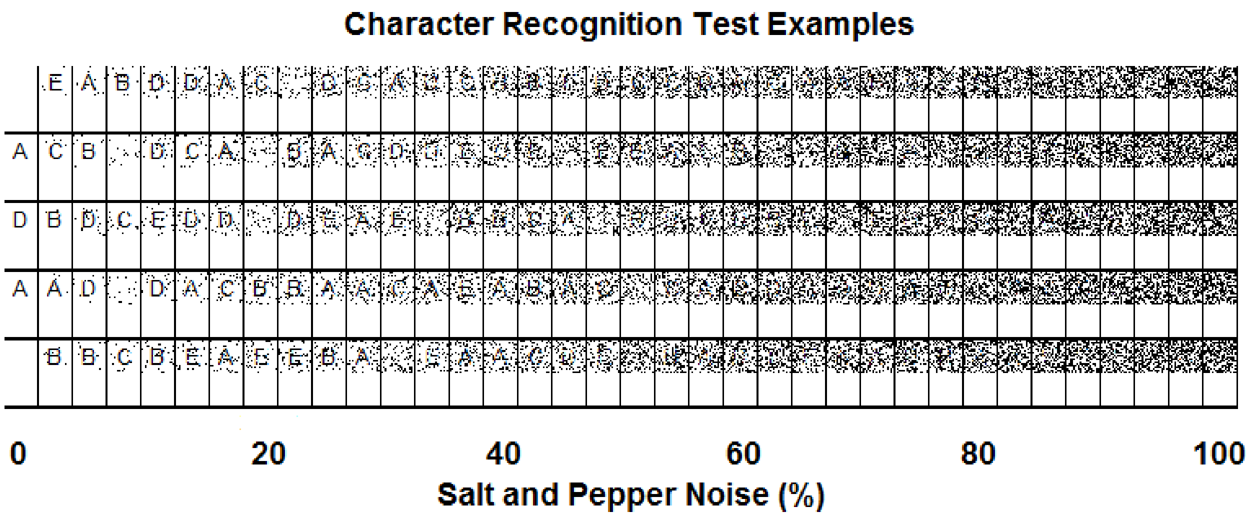

Figure 1: Examples of the fixed-location characters with noise added

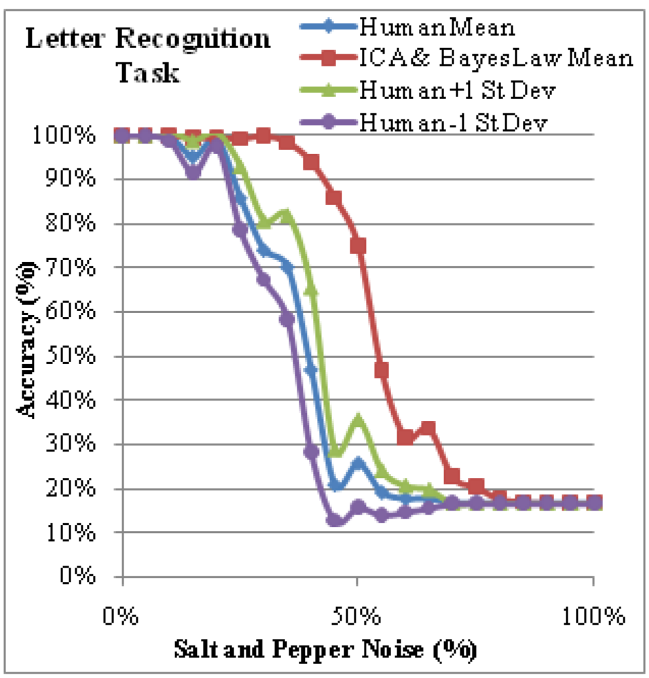

Figure 2: Comparison between ICA-Bayes classification algorithm and human performance on an example character recognition task. Also graphed is human performance +1 and -1 standard deviation.

As shown in Fig 2, the ICA/Bayes algorithm dramatically outperformed humans in this pattern-recognition task. While alternative algorithms may perform comparably, it is important to note the generality and the ease of use of the ICA/Bayes approach. No *a priori* knowledge was required, and user interaction with the pattern-classification system only required presenting the system with training examples.

B. ICA for Scene Preprocessing

A second illustration of the algorithm is a mobile-robotics application. The challenge is to identify key features in the environment. In particular, it is desired to identify white lines that bound where the robot can and cannot drive. The ICA-Bayes algorithm was evaluated in terms of four categories- (“white line”, “barrel”, “light grass” and “dark grass”). Results are shown in Fig.3.

Training was performed by manually classifying thumbnails within example images as belonging to one of the available classifications. For this purpose, 200 (7x7 pixel) thumbnails were manually classified from 12 (640x480 pixel) scenes. These thumbnails were used to identify ICA components. Scenes were then analyzed in terms of thumbnails, and each thumbnail was assigned to one of the pre-established groups. The result of classifying each thumbnail can be interpreted as a coarse image with 49x fewer pixels and which is suitable for subsequent higher-level processing (e.g. identification of continuous lines).

With conventional image-processing techniques, substantial tuning by trial and error would be required to obtain sufficiently robust recognition of what constitutes part of a white line. Simple thresholding is inadequate, color matching is unreliable, and edge detection is too noisy to identify white-line membership. In contrast, the ICA/Bayes classification approach implicitly incorporates all linear image processing options as well as use of color. Importantly, no expertise is required in using this algorithm. It is only necessary for the user to create training examples (in this case, using “Paint” to select example regions of line, grass and barrel).

The results shown benefitted from subdividing the category of “grass” into two sub-categories—light grass and dark grass. In the present case, this was done manually. In continuing work, identification of useful subcategories is being performed automatically by recognition of non-Gaussian distributions of training examples within a category. Such automation would be important in the creation of a system intended for use by non-experts.

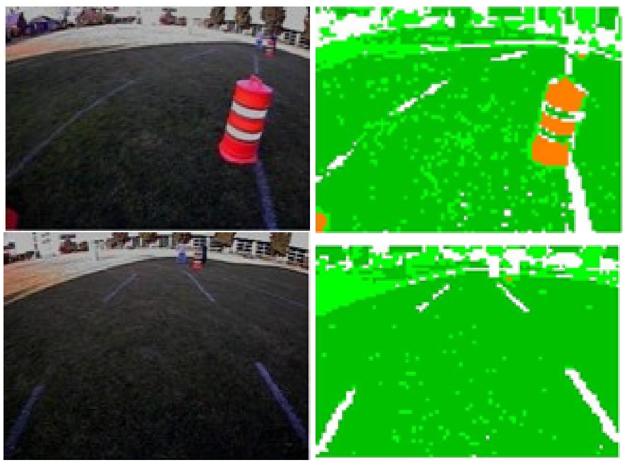

Fig 3: original scenes (left) and resulting coarse-scene classifications (right) for mobile-robotics application.

C. *ICA for Industrial Inspection*

A third example shows the potential for application of the present algorithm to industrial inspection. In this example, the task is to classify cellulose kitchen sponges as either “first quality” or “other”. This classification is to be performed in real time, in-line with the production line.

According to the manufacturer, this task has not been automated in the past due to a variety of related complications. One, the color of sponge changes frequently, so the inspection system must be able to adapt to this. This can occur both from run to run (variation in dye quality), but also when color is intentionally changed (for example, from blue to yellow.) Two, individual sponges are textured, with both a repetitive texture and random hole sizes and locations. This means individual examples of “first quality” are not consistent in appearance, something that thwarts many vision algorithms. Third, the variety of failure modes is huge and poorly defined. For example, some batches might have “rawness” while others might have “white cotton spots.” Adaptability in the vision system is important to catch these continuously changing circumstances.

One common category is that the sponge has regions in which it is “undercooked” in the manufacturing process. Training data was obtained by acquiring images (752x480 pixels) from a variety of sponges that had been pre-sorted as either first quality or flawed due to “raw” regions. Full images were subdivided into thumbnails (20x20 pixels), and these thumbnails were manually classified as “good” or “raw.” These pre-sorted thumbnails comprised the training set.

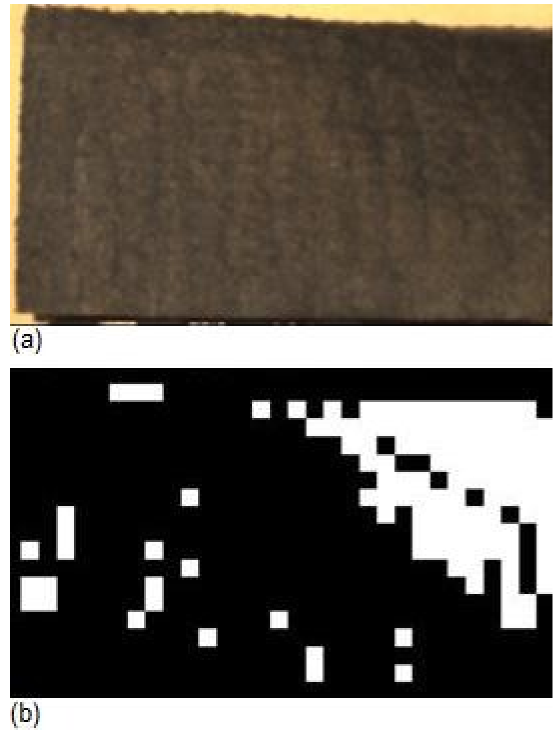

Figure 4: ICA-Bayes performance on the Sponge Inspection Task. (a) shows the original image, (b) shows the Two-Group result (White- Bad Sponge, Black-Other

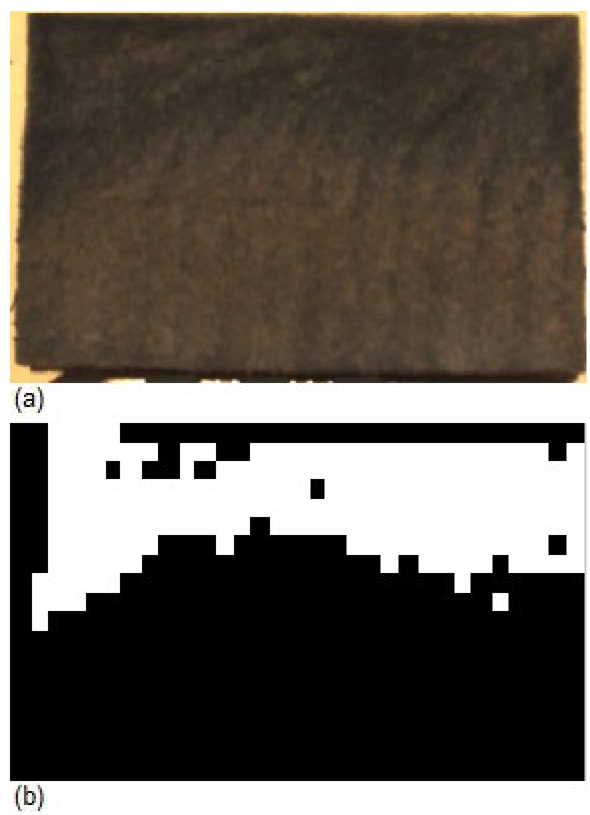

Figure 5: ICA-Bayes performance on the Sponge Inspection Task. (a) shows the original image, (b) shows the Two-Group result (White- Bad Sponge, Black-Other

After training, new images were introduced and the algorithm categorized thumbnails within these images as "good" or "raw." Figures 4 and 5 show representative images and the corresponding sub-image classifications into categories of "good" (black) or "raw" (white).

For the sponge-inspection task, most-likely classification of thumbnails into categories does not, in itself, optimize the inspection solution. An additional consideration is the cost associated with a particular type of error (false positive vs. false negative). A convenient means to incorporate consideration of misclassification penalty weights is to alter the *a-priori* probability. Instead of using the *a-priori* probability of "member of group", we can instead define an *a-priori* probability of "other", such that a cost metric is optimized. This can be determined by defining a cost function and then finding the *a-priori* value that minimizes that function over the training set of sponges.

With a training set of 800 sponges (434 sellable, 366 other), using this method resulted in overall sponge classification accuracy of 95% for "sellable" sponge and 95% "other" sponge. This outperformed the human accuracy levels of 93% "sellable" and 87% "other".

V. Summary and Conclusions

This paper describes the application of ICA and Bayes' Theorem to image classification tasks. This method was illustrated in three problem domains, each of which showed good performance. In particular, in both the synthetic problem domain of simple character recognition and the industrial domain of sponge-manufacturing inspection, the ICA-Bayes' Theorem algorithm was shown to outperform humans. Additionally, potential for application to image processing for mobile robotics was demonstrated.

In utilizing this algorithm, one should appreciate where the primary effort lies. Computationally, identifying ICA basis vectors can be slow. However, suitable algorithms exist, and their application can be automated within a user-friendly system. Further, it should be appreciated that the computationally-intensive steps occur in preprocessing. Subsequent use of an identified ICA transform is quite fast, and thus it is suitable for real-time pattern classification.

Training the system can be tedious, since it is necessary to provide a sufficiently rich set of pre-categorized examples. However, many pattern-classification and industrial-inspection applications are not defined in terms of analytic specifications; often, classifications are inferred only by examples. In such cases, training examples can be obtained by domain experts (e.g. shop-floor inspectors), and subsequent use of the ICA/Bayes classification system would not require intervention by an expert programmer. This attribute makes the ICA/Bayes technique attractive for future shop-floor use.

References

[1] Beckmann, C. F., & Smith, S. M. (2004). Probablistic Independent Component Analysis for Functional Magnetic Resonance Imaging. *IEEE Transactions of Medical Imaging , 23*, 137-152.

[2] Bell, A. J., & Sejnowski, T. J. (1997). The 'Independent Components' of Natural Scenes are Natural Filters. *Vision Research , 37*, 3327-3338.

[3] Bishop, C. M. (2006). *Pattern Recognition and Machine Learning.* Springer.

[4] Celik, T., & Tjahjadi, T. (2009). Multiscale texture classification using dual-tree complex wavelet transform. *Pattern Recognition Letters , 30*, 331-339.

[5] Choi, S. (2006). Differential learning algorithms for decorrelation and independent component analysis. *Neural Networks , 19*, 1558-1567.

[6] Christoyianni, I., Koutras, A., Dermatas, E., & Kokkinakis, G. (2002). Computer aided diagnosis of breast cancer in digitized mammograms. *Computertized Medical Imaging and Graphics , 26*, 309-319.

[7] Comon, P. (1994). Independent Component Analysis, A new concept? *Signal Processing , 36*, 287-314.

[8] Hyvarinen, A. (1999). Fast and Robust Fixed-Point Algorithms for Independent Component Analysis. *IEEE Transactions on Neural Networks*, *10*, pp. 626-634.

[9] Hyvarinen, A. (1999). Survey on Independent Component Analysis. *Neural Computing Surveys , 2*, 94-128.

[10] Hyvarinen, A., & Oja, E. (2000). Independent Component Analysis: Algorithms and Applications. *Neural Networks , 13* (4-5), 411-430.

[11] Karklin, Y., & Lewicki, M. S. (2003). Learning higher-order structures in natural images. *Network:Computation in Neural Systems , 14*, 483-499.

[12] Mack, D. R. (1993). Reverend Bayes' Useful Contribution. *IEEE Potentials , 12* (1), 41-42.

[13] Rasmussen, C. E. (2000). The Infinite Gaussian Mixture Model. *Advances in Neural Information Processing Systems , 12*, 554-560.

[14] Shlens, J. (2005, December 10). *A Tutorial on Principal Component Analysis*, 2. Retrieved September 14, 2009, from www.snl.salk.edu/~chlens/notes.html

[15] Vitria, J., Bressan, M., & Radeva, P. (2007). Bayesian Classification of Cork Stoppers Using Class-Conditional Independent Component Analysis. *IEEE Transactions on Systems, Man and Cybernetics- Part C: Applications and Reviews*, *37.*