Classifiers and Machine Learning Techniques for Image Processing and Computer Vision

Anderson Rocha, Siome Goldenstein

Institute of Computing
University of Campinas (Unicamp)
13084–851, Campinas, SP – Brazil
{anderson.rocha, siome}@ic.unicamp.br

***Abstract*—We propose the use of classifiers and machine learning techniques to extract useful information from data sets (e.g., images) to solve important problems in Image Processing and Computer Vision. We are interested in: two and multi-class image categorization, hidden messages detection, discrimination among natural and forged images, authentication, and multi-classification.**

***Keywords*-Machine Learning Techniques; Digital Forensics; Steganalysis; Feature Fusion; Classifier Fusion; Multi-class Classification; Image Categorization.**

I. Introduction

In the digital age, information reaches us at remarkable speed and the amount of data it brings is unprecedented. In the hope of understanding such flood of information, data mining and machine learning approaches are required. In this scenario, one of the problems we often face is data categorization (classification).

Data categorization solutions need to provide approaches that lead to the proper understanding of the scene context under investigation in order to allow correct decisions (e.g., pointing out the class of an object).

Notwithstanding, defining the features and nuances that we would like to select is not an easy task and machine learning as well as pattern recognition techniques can be valuable tools.

In this PhD work, we devise and deploy classifiers and machine learning techniques to extract useful information from data sets (e.g., images), relevant similarity measures among examples, or even semantical features in a data set to solve important problems in Image Processing and Computer Vision.

We are particularly interested in: two and multi-class image categorization, hidden messages detection, discrimination among natural and forged images, authentication, and multi-classification.

To start with, we introduce a technique for image forensics analysis and image classification in categories such as *indoors*, *outdoors*, *computer generated*, and *art works*.

From this multi-class classification, we face some important questions: how to solve a multi-class problem in order to combine, for instance, several different features such as color, texture, shape, and silhouette without worrying about the pre-processing and normalization of the combined feature vector? How to take advantage of different classifiers, each one custom tailored to a specific set of classes in confusion? To cope with most of these problems, we present a feature and classifier fusion technique based on combinations of binary classifiers. We also validate our solution with a real application for automatic produce classification.

Finally, we address another interesting problem: how to combine powerful binary classifiers in the multi-class scenario more effectively? How to boost their efficiency? In this context, we present a solution that boosts the efficiency and effectiveness of multi-class from binary techniques.

II. Contributions — Image Categorization and Digital Media Forensics

To the human eye, changes in the values of the least significant bit (LSBs) of images are imperceptible, making them an ideal place for hiding information without perceptual change in the cover object. We have found out that small perturbations in this LSB channel, while imperceptible to humans, are statistically detectable in the context of image analysis [1].

In this work, we introduce the Progressive Randomization (PR) image meta-description approach for *Image Categorization* and *Digital Media Forensics* in the context of *Hidden Messages Detection*. The Progressive Randomization approach captures the differences between broad-image classes using the statistical artifacts inserted during a controlled perturbation process.

A. Hidden messages detection

Digital Steganalysis is a categorization problem in which we want to distinguish between *non-stego* or *cover objects*, those that do not contain a hidden message, and *stego-objects*, those that contain a hidden message.

Recently, *Steganography*, the art of hidden communications, has received a lot of attention around the world mainly because its potential applications: identification of subcomponents within a data set, captioning, time-stamping, and tamper-proofing (demonstration that original contents have not been altered) [2]. Unfortunately, not all applications are harmless, and there are strong indications that Steganography has been used to spread child pornography pictures on the internet [3]. Robust algorithms to detect the very existence of hidden messages in digital contents can help further forensic and police work.

The Progressive Randomization method we propose in this work allows us to detect LSB-based hidden messages in digital images with lossless compression (e.g., PNGs). Furthermore, the technique is able to identify which kinds of images are more sensitive to the hidden messages embedding as well as which kinds of images are more suitable to these operations.

B. Image Categorization

Image Categorization is the body of techniques that distinguish between image classes, pointing out the global semantic type of an image. Here, we want to distinguish the class of an image (e.g., *Indoors* from *Outdoors*). One possible scenario for a consumer application is to group a photo album, automatically, according to classes.

We have sought do devise and deploy a technique able to combine relevant information shared by common examples in large-scale image databases in order to categorize them into two classes such as: *natural* vs. *computer generated images* [4], [5], *outdoors* vs. *indoors* [6], [7], and *artworks* vs. *natural images* [8]. In addition, we have tested the system to categorize images into $n = 4$ classes in a multi-class scenario with the categories: natural outdoor images, natural indoor images, artwork, and computer generated images.

Our approach captures the statistical properties of the analyzed classes and seeks for variations in such properties.

C. Progressive Randomization (PR)

Here, we introduce the Progressive Randomization image meta-description approach for *Image Categorization* and *Steganalysis*. PR captures the differences between broad-image classes using the statistical artifacts inserted during a controlled perturbation process. The most important observation is that different image classes do have distinct behavior when undergoing successive controlled perturbations.

The method consists of four stages: (1) the randomization process; (2) the selection of feature regions; (3) the statistical descriptors analysis; and (4) the invariance transformation.

In the randomization process, we progressively perturb the LSB value of a selected number of pixels. Perturbations of different intensities can be carried out. Each one will result a new perturbed image.

With the region selection, we select image regions of interest. For each perturbed image, we select some regions of interest and, for each one, we use statistical descriptors to characterize it.

If we want to evaluate only the relative variations across the perturbations, we perform a normalization with respect to the values in the input image, the one that does not have any perturbation. This amounts to the invariance step.

Pixel perturbation: Given an input image I of $|I|$ pixels, we define the LSB pixel perturbation $T(I,p)$ as the process of substitution of the LSBs of a random set of pixels S of length $p \times |I|$. The replacement process uses a bit sequence B comprising independent draws from a Bernoulli distributed random variable, $Pr(\{\mathbf{x} = 0\}) = Pr(\{\mathbf{x} = 1\}) = \frac{1}{2}$. In this case, p represents the percentage of pixels of the $|I|$ available pixels in the image we want to select.

Consider a pixel $px_i \in S$ and one associated bit $b_i \in B$ such that $\mathcal{L}(px_i) \leftarrow b_i$ for all $px_i \in S$. In this case, $\mathcal{L}(px_i)$ is the LSB of pixel px_i. Figure 1 depicts one perturbation example using the bits $B = 1110$.

The randomization process: Given an input image I, the randomization process consists of successive perturbations $T(I, P_1), \ldots, T(I, P_n)$ over the pixel LSBs in I, where $P_i \in P$ stands for the relative sizes of the selected amount of pixels in S.

The $T(I, P_i)$ transformations are controlled perturbations of different percentages (weights) over the available LSBs. In our base implementation, we have used $n = 6$ where $P = \{1\%, 5\%, 10\%, 25\%, 50\%, 75\%\}$. The process yields n images which only differ in their LSB channel used in the perturbations.

Region selection and statistical description: Given an input image I, we use r regions with size $l \times l$ pixels to make the further statistical characterization process more localized. The region selection can be as simple as the selection of static overlapping regions as well as can use more sophisticated filters accounting for image regions with high richness of details (c.f. [9]).

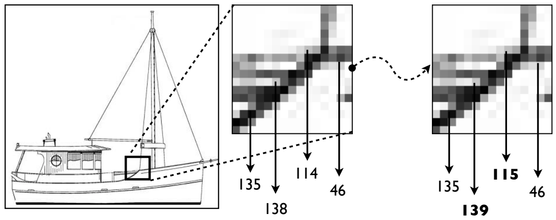
The LSB perturbations change the content of a selected set of pixels and yield localized changes in the statistics of such pixels. A pixel with L bits spans 2^L possible values and represents 2^{L-1} classes of invariance if we consider only the LSB possible changes. We call this classes pairs of values (PoV). When we disturb all the LSBs available in a set S with a sequence of bits B, the 0/1 distribution of values in a given PoV will be the same as in B. We use the statistical descriptors χ^2 (chi-squared test) [10] and U_T (Ueli Maurer Universal test) [11] to analyze such changes.

Invariance: In some situations, it is necessary to use an image-invariant feature vector. For that, we normalize all descriptors values with regard to their values in the input image (the one with no perturbation)

$$F = \{f_e\}_{e=1\ldots n\times r\times m} = \left\{\frac{d_{ijk}}{d_{0jk}}\right\}_{i=0\ldots n, j=1\ldots r, k=1\ldots m.} \tag{1}$$

where d denotes a descriptor $1 \leq k \leq m$ of a region $1 \leq j \leq r$ of an image $0 \leq i \leq n$ and F is the final feature vector created for the image I.

The need for invariance depends on the application. For instance, it is necessary for Steganalysis but harmful for Image Categorization. In Steganalysis, we want to differentiate images that do not contain hidden messages from those that contain hidden messages, and the image class is



Binary Representation			
135	1000 0111	**114**	0111 0010
138	1000 1010	**46**	0010 1110

Figure 1. LSB perturbation example using the bits $B = 1110$.

not important. On the other hand, in Image Categorization, the descriptor values are important to improve the class differentiation (c.f. [9]).

III. Contributions — Feature and Classifier Fusion

In some situations, multi-class categorization problems are so complex that simple characterization techniques are not adequate to solve them. In order to deal with these complex problems, feature and classifier fusion techniques may become mandatory.

In spite of the fact that feature-level combination is not straightforward for multi-class problems, for binary problems this is a simple task. In this scenario, it is possible to combine different classifiers and features by using classic rules such as `and`, `or`, `max`, `sum`, or `min` [12]. For multi-class problems, this is more difficult given that one feature might point out to an outcome class C_i and another feature might result the outcome class C_j, and even another one could result C_k. With many different resulting classes for the same input example, it becomes difficult to define a consistent policy to combine the selected features.

In this context, one approach sometimes used is to combine the feature vectors for different features into a single and big feature vector. Although quite effective for some problems, this approach can also yield unexpected classification results when not properly prepared. First, in order to create the combined feature vector, we need to tackle the different nature of each feature. Some can be well conditioned such as continuous and bounded variables, others can be ill-conditioned for this combination such as categorical ones. In addition, some variables can be continuous and unbounded. To put everything together, we need a well-suited normalization. However, this normalization is not always possible or, sometimes, it leads to undesirable properties in the new generated feature vector such as equally weighting all the feature coefficients, a property that in general we do not want.

When combining feature vectors this way, eventually we would need to cope with the curse of dimensionality. Given that we add new features, we increase the number of dimensions which then might require more training data.

Finally, if we want to add a new feature, we might need to redesign the normalization in order to deal with most of these problems. In Section III-A, we present a simple and effective way to feature and classifier fusion that cope with most of the previously discussed concerns.

A. Proposed solution

To deal with most of the aforementioned problems, we propose a unified approach that can combine many *features* and *classifiers*, requires less training, and is more adequate to some problems than a naïve method, where all features are simply concatenated and fed independently to each classification algorithm.

To accomplish our objective, we propose to cope with the multi-class problem as a set of binary problems. In this context, we define a *class binarization* as a mapping of a multi-class problem onto two-class problems (divide-and-conquer) and the subsequent combination of their outcomes to derive the multi-class prediction. We refer to the binary classifiers as *base learners*. Class binarization has been used in the literature to extend naturally binary classifiers to multi-class and SVM is one example of this [13]. However, to our knowledge, this approach was not used before for classifier and feature fusion.

In order to understand the class binarization, consider a toy example problem with three classes. In this case, a simple binarization consists in training three base learners, each one for two classes at a time. In this sense, we need $O(N^2)$ binary classifiers, where N is the number of classes.

We train the ij^{th} binary classifier using the patterns of class i as positive and the patterns of class j as negative examples. To obtain the final outcome we just calculate the minimum distance of the generated vector (binary outcomes) to a binary pattern representing each class.

Consider again a toy example with three classes as we show in Figure 2. In this example, we have the classes: *Triangles* $\triangle$, *Circles* $\bigcirc$, and *Squares* $\Box$. Clearly, one first feature we can use to categorize elements of these classes can be based on shape. As we can see, we can also use texture and color properties. To solve this problem, we train binary classifiers differentiating two classes at a time, such as $\triangle \times \bigcirc$, $\triangle \times \Box$, and $\bigcirc \times \Box$. Also, we give each one of the available classes a unique identifier (e.g., $\triangle = \langle +1, +1, 0 \rangle$).

When we receive an input example to classify, let's say a triangle-shaped one, as we show in the picture, we first employ the binary classifiers to verify if the input example is a triangle or a circle based on shape, texture and color features. Each classifier will give us a binary response. Let's

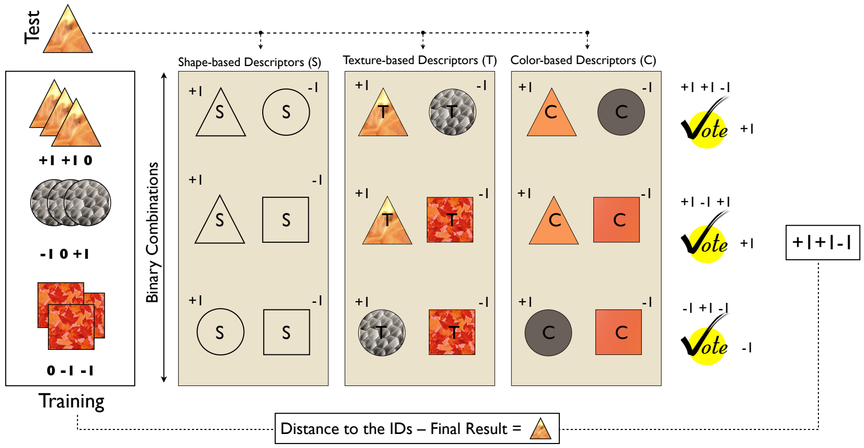


Figure 2. Toy example for feature and classifier fusion.

say we obtain the votes $\langle +1, +1, -1\rangle$ for the binary classifier $\triangle \times \bigcirc$. Thereafter, we can use majority voting and select one response (+1 in this case, or $\triangle$). Then we repeat the procedure and test if the input example is a triangle or a square, again for each one of the considered features. Finally, after performing the last test, we end up with a binary vector. Finally, we calculate the minimum distance from this binary vector to each one of the class unique IDs. In this example, the final answer is given by the minimum distance of

$$\min \mathcal{D}(\langle 1, 1, -1\rangle, \{\langle 1, 1, 0\rangle, \langle -1, 0, 1\rangle, \langle 0, -1, -1\rangle\}. \quad (2)$$

One aspect of this approach is that it requires more storage given that once we train the binary classifiers we need to store their parameters. However, many classifiers in our daily-basis employ some sort of class binarization (e.g., SVMs) and are considered fast.

As we present in Section IV, we can reduce storage requirements and speed up the classification itself by selecting classes that are in confusion and designing specific binary classifiers to separate them.

In summary, the presented approach has the following advantages:

1) By combining independent features, we have more confidence in a resulting outcome given that it is calculated from the agreement of more than one single feature. Hence, we have a simple error-correcting mechanism that can withstand some misclassifications;
2) If we find some classes that are in confusion and are driving down the classification results, we can design special and well-tuned binary classifiers and features to separate them;
3) We can easily point out whether or not a given feature is indeed helping the classification. This is not straightforward with normal binding in a big feature vector;
4) The addition of new classes only require training for the new binary classifiers;
5) The addition of new features is simple and just requires partial training;
6) As we do not increase the size of any feature vector, we are less prone to the curse of dimensionality not requiring more training examples when adding more features.

Finally, we have validated the system addressing a fruit-and-vegetable categorization task in a produce distribution center. The approach has yielded good accuracy results when considering a 15-class classification scenario (c.f. [14]).

IV. Contributions — Multi-class from Binary Classifiers

Recently, there has been a lot of success in the development of effective binary classifiers. Although many statistical classification techniques do have natural multi-class extensions, some, such as SVMs, do not. Therefore, it is important to know how to map the multi-class problem into a set of simpler binary classification problems. One of the most common approaches for multi-class from binary is known as class binarization. A *class binarization* is a mapping of a multi-class problem onto several two-class problems (divide-and-conquer) and the subsequent combination of their outcomes to derive the multi-class prediction [13], [15], [16]. We refer to the binary classifiers as *base learners* or *dichotomies*.

There are many possible approaches to reduce multi-class to binary classification problems. We can classify such approaches into three broad groups: (1) *One-vs-All* (OVA), (2) *One-vs-One* (OVO), and (3) *Error Correcting Output Codes* (ECOC). Also, the multi-class decomposition into binary problems usually contains three main parts: (1) the ECOC matrix creation; (2) the choice of the base learner; and (3) the decoding strategy.

Focusing on the creation of the ECOC matrix and on the decoding strategy, in this work, we introduce a new Bayesian multi-class from binary reduction method. First, we introduce the concept of correlations among outcomes of binary classifiers, and then we present a principled way to find groups of highly correlated base learners. With that in hand, we can learn the proper joint probabilities that allow us to predict the class. Finally, we present two additional strategies: one to reduce the number of required base learners in the multi-class classification and the other to find new dichotomies (classes to be compared using one binary classifier) in order to replace the less discriminative ones.

In order to classify an input example, we use a team of binary classifiers $\mathcal{T}$. We call $\mathcal{O}_\mathcal{T}$ a realization (set of outcomes) of the classifiers in the team $\mathcal{T}$. Each element in $\mathcal{T}$ is a base learner (binary classifier) and produces an outcome $\in \{-1, +1\}$.

Given an input element x to classify, a realization $\mathcal{O}_\mathcal{T}$ contains the information to determine the class of x. In other words, $P(y = c_i|x) = P(y = c_i|\mathcal{O}_\mathcal{T})$.

However, we do not have the probability $P(y = c_i \mid \mathcal{O}_\mathcal{T})$. From Bayes theorem,

$$\begin{aligned} P(y &= c_i|\mathcal{O}_\mathcal{T}) = \frac{P(\mathcal{O}_\mathcal{T}|y = c_i)P(y = c_i)}{P(\mathcal{O}_\mathcal{T})} \\ &\propto P(\mathcal{O}_\mathcal{T}|y = c_i)P(y = c_i) \end{aligned} \quad (3)$$

$P(\mathcal{O}_\mathcal{T})$ is just a normalizing factor and it is ruled out.

Previous approaches have solved the above model by considering the independence of the dichotomies in the team $\mathcal{T}$ [15]. If we consider independence among all dichotomies, the model in Equation 3 becomes

$$P(y = c_i|\mathcal{O}_\mathcal{T}) \propto \prod_{t \in \mathcal{T}} P(\mathcal{O}^t_\mathcal{T}|y = c_i)P(y = c_i), \quad (4)$$

and the class of the input x is given by

$$cl(x) = \arg\max_i \prod_{t \in \mathcal{T}} P(\mathcal{O}^t_\mathcal{T}|y = c_i)P(y = c_i). \quad (5)$$

Although the independence assumption simplifies the model, it comes with limitations and it is not the best choice in all cases. In our work, we relax the assumption of independence among all binary classifiers. When two of these dichotomies have a lot in common, it would be unwise to deal with their results as independent random variables (RVs). In our approach, we find groups of highly correlated classifiers and represent their outcomes as dependent RVs, using a single *conditional probability table* (CPT) as an underlying distribution model. Each group then has its own CPT, and we combine the groups as if they are independent from each other — to avoid a dimensionality explosion. Our technique is a Bayesian-Network-inspired approach for RV estimation. We decide the RV that represents the class based on the RVs that represent the outcomes of the dichotomies.

We model the multi-class classification problem conditioned to groups of affine dichotomies $\mathcal{G}_\mathcal{D}$. The model in Equation 3 becomes

$$P(y = c_i|\mathcal{O}_\mathcal{T}, \mathcal{G}_\mathcal{D}) \propto P(\mathcal{O}_\mathcal{T}, \mathcal{G}_\mathcal{D}|y = c_i)P(y = c_i). \quad (6)$$

We assume independence only among the groups of binary classifiers that are considered highly correlated $g_i \in \mathcal{G}_\mathcal{D}$. Therefore, the class of an input x is given by

$$cl(x) = \arg\max_j \prod_{g_i \in \mathcal{G}_\mathcal{D}} P(\mathcal{O}^{g_i}_\mathcal{T}, g_i|y = c_j)P(y = c_j). \quad (7)$$

To find the groups of highly correlated classifiers $\mathcal{G}_\mathcal{D}$, we define a relationship matrix $\mathcal{A}$ among the classifiers. The relationship matrix measures how correlated are the outcomes of two binary classifiers when classifying a set of training examples X (c.f. [17]).

After calculating the relationship matrix $\mathcal{A}$, we use a clustering technique to find correlated binary classifiers in $\mathcal{A}$. In addition, we use a procedure to identify the less important binary classifiers within each group of highly correlated binary classifiers and eliminate them (*Shrinking*). For that, we calculate the cumulative entropy of each group testing one binary classifier each time. Those yielding the lowest information gain are tagged as less important to the group.

The elimination of some binary classifiers takes us to the process of finding other binary classifiers potentially interesting. This is the next stage of our approach named *Augmenting*. In this stage, we look for new candidate binary classifiers to replace those ones eliminated in the stage of *Shrinking*. For that, we analyze the resulting training confusion matrix and represent the classes as a graph where the nodes are classes' identifiers and the edges are the confusion between the classes. From the graph we are able to create a hierarchy of classes in confusion and split, using normalized cuts, the nodes (c.f. [17]).

V. Results and Conclusions

In this PhD work, we have used several machine learning and classification techniques to extract useful information within data sets. We have showed that such information are valuable and can be exploited to solve a variety of Image Processing and Computer Vision related problems. This PhD work has been done in three years and has resulted[1] in:

- Three patent deposits [18]–[20];
- Three international journal papers [9], [14], [21] with impact factors of 1.27, 2.22 and 9.92, respectively;
- One national journal paper [2];
- Four international conference papers [1], [17], [22], [23];
- One national conference paper [24].

In summary, we have introduced a meta-description approach named Progressive Randomization for image categorization and digital media forensics in the context of hidden messages detection. The image categories we have considered were *indoors*, *outdoors*, *computer generated images* and *artworks*.

The introduced technique detects hidden messages of medium size (e.g., $\cong 25\%$ of the channel capacity) with accuracy above 90%. Messages of this size could be the

[1] For further reference, please visit `http://www.ic.unicamp.br/~rocha/wtd-sibgrapi-2010/`

amount of information used by a criminal to spread 80-100-*kilobyte* child porn images hidden within unsuspicious $1,280 \times 1,024$-pixel wallpapers. In addition, the presented technique is able to separate natural and synthetic images, for instance, with an accuracy above 90%.

During the work, we also have discovered that a multi-class classification problem could be solved in a more principled way using smart combinations of features and classifiers. The combination of features and classifiers can be more effective than the traditional focus on a universal descriptor that solves the whole problem.

Given that each class of interest has certain peculiarities, finding a unique feature that captures all the class properties is a complex problem. In this sense, we have designed a multi-class feature and classifier fusion technique using class binarization methods.

We have validated the approach in a real application for semi-automatic produce classification. For this particular application, we have created a database with thousands of images acquired in local produce distribution center. The devised technique achieves an accuracy of $\cong 98\%$ for the top result and over 99% when we contemplate the top two results considering shape, appearance, texture, and color information.

Finally, we have introduced the concept of correlations among the outcomes of binary classifiers to solve multi-class problems using smaller pieces of such problems (divide-and-conquer). The combination of small binary classifiers can be very effective when solving a complex multi-class problem. The experiments have shown that the introduced technique can solve an 1,000-class categorization problem with just a few, but powerful, binary classifiers.

Acknowledgments

The authors thank the São Paulo Research Foundation (FAPESP) for its support on this project (Grant 05/58103-3).

References

[1] A. Rocha and S. Goldenstein, “Progressive randomization for steganalysis,” in *IEEE MMSP*, Victoria, Canada, 2006, pp. 314–319.

[2] ——, “Steganography and steganalysis in digital multimedia: Hype or hallelujah?” *RITA*, vol. 15, no. 1, pp. 83–110, 2008.

[3] S. Morris, “The future of netcrime now: Part 1 – threats and challenges,” Home Office Crime and Policing Group, Washington DC, USA, Tech. Rep. 62/04, 2004.

[4] S. Lyu and H. Farid, “How realistic is photorealistic?” *IEEE TSP*, vol. 53, no. 2, pp. 845–850, 2005.

[5] S. Dehnie, T. Sencar, and N. Memon, “Identification of computer generated and digital camera images for digital image forensics,” in *ICIP*. Atlanta, USA: IEEE, 2006.

[6] A. Payne and S. Singh, “Indoor vs. outdoor scene classification in digital photographs,” *Pattern Recognition*, vol. 38, no. 10, pp. 1533–1545, 2005.

[7] N. Serrano, A. Savakis, and J. Luo, “A computationally efficient approach to indoor/outdoor scene classification,” in *ICPR*. IAPR, 2002, pp. 146–149.

[8] F. Cutzu, R. Hammoud, and A. Leykin, “Distinguishing paintings from photographs,” *Elsevier CVIU*, vol. 100, no. 3, pp. 249–273, March 2005.

[9] A. Rocha and S. Goldenstein, “Progressive randomization: Seeing the unseen,” *Computer Vision and Image Understanding (CVIU)*, vol. 114, no. 3, pp. 349–362, March 2010.

[10] A. Westfeld and A. Pfitzmann, “Attacks on steganographic systems,” in *IHW*. Dresden, Germany: Springer, 1999, pp. 61–76.

[11] U. Maurer, “A universal statistical test for random bit generators,” *Intl. Journal of Cryptology*, vol. 5, no. 2, pp. 89–105, February 1992.

[12] C. M. Bishop, *Pattern Recognition and Machine Learning*, 1st ed. Springer, 2006.

[13] T. G. Dietterich and G. Bakiri, “Solving multi-class learning problems via error-correcting output codes,” *JAIR*, vol. 2, no. 1, pp. 263–286, January 1996.

[14] A. Rocha, D. Hauagge, J. Wainer, and S. Goldenstein, “Automatic fruit and vegetable classification from images,” *COMPAG*, vol. 70, no. 1, pp. 96–104, January 2010.

[15] A. Passerini, M. Pontil, and P. Frasconi, “New results on error correcting output codes of kernel machines,” *IEEE TNN*, vol. 15, no. 1, pp. 45–54, Jan 2004.

[16] N. Pedrajas and D. Boyer, “Improving multi-class pattern recognition by the combination of two strategies,” *IEEE TPAMI*, vol. 28, no. 6, pp. 1001–1006, Jun 2006.

[17] A. Rocha and S. Goldenstein, “From binary to multi-class: divide to conquer,” in *INSTIC VISAPP*, 2009, pp. 323–330.

[18] ——, “Processo e equipamento de randomização progressiva para análise e raciocínio em multimídia,” INPI, Patente PI0605994-5, September 29th 2006.

[19] ——, “Progressive randomization process and equipment for multimedia analysis and reasoning,” WIPO, Patent PCT/BR2007/000156, June 15th 2008.

[20] A. Rocha, D. C. Hauagge, J. Wainer, and S. Goldenstein, “Método e sistema para reconhecimento e classificação de objetos em ambientes semi-controlados,” PCT, Patent PCT/BR2010/00073, March 3rd 2010.

[21] A. Rocha, W. Scheirer, T. E. Boult, and S. Goldenstein, “Vision of the unseen: Current trends and challenges in digital image and video forensics,” *ACM CSUR*, 2011, in Press.

[22] A. Rocha, W. Scheirer, S. Goldenstein, and T. E. Boult, “The Unseen Challenge Data Sets,” in *IEEE CVPR WVU*. Anchorage, USA: IEEE, 2008, pp. 176–183.

[23] A. Rocha and S. Goldenstein, “PR: More than Meets the Eye,” in *IEEE ICCV*, 2007, pp. 1–8.

[24] A. Rocha, D. Hauagge, J. Wainer, and S. Goldenstein, “Automatic produce classification from images using color, texture and appearance cues,” in *SIBGRAPI*, 2008, pp. 3–10.