Automatic blur detection for meta-data extraction in content-based retrieval context

Jérôme da Rugna, Hubert Konik

Automatic blur detection for meta-data extraction in content-based retrieval context

Jérôme Da Rugna [a] and Hubert Konik[a]

[a]Université Jean Monnet
Laboratoire LIGIV EA 3070, 10 rue Barrouin, Saint-Étienne, France

ABSTRACT

During the last few years, image by content retrieval is the aim of many studies. A lot of systems were introduced in order to achieve image indexation. One of the most common method is to compute a segmentation and to extract different parameters from regions. However, this segmentation step is based on low level knowledge, without taking into account simple perceptual aspects of images, like the blur. When a photographer decides to focus only on some objects in a scene, he certainly considers very differently these objects from the rest of the scene. It does not represent the same amount of information. The blurry regions may generally be considered as the context and not as the information container by image retrieval tools. Our idea is then to focus the comparison between images by restricting our study only on the non blurry regions, using then these meta data. Our aim is to introduce different features and a machine learning approach in order to reach blur identification in scene images.

Keywords: content-based image retrieval, meta data, blur detector, hierarchical process, machine learning

1. INTRODUCTION

Among visual information systems, query by content is a very common model developed for content-based image retrieval. Nevertheless, even if this research thematic has expanded rapidly these last few years, many problems have put a break on its real concrete use. In fact, one of the most common well-known problem is the so-called semantic gap between the true capacity of such tools and the real hope of the potential users. Actually, it seems quite illusive to reach the semantic aspect by only using the image of a scene. For example, an image of a rising sun conveys other semantics notions : holidays, romanticism and so on... and these notions are out of reach by only query image by content.

Nevertheless, keeping that problematic in mind, some meta data can be extracted directly from the image in some precise applications. For example, considering classical holiday photographies or portraits, the blurry regions represent generally the background, otherwise speaking regions of no interest for the viewer. On the contrary, the non blurry regions can be normally seen as the interest part of the global image, at least for the one who took the picture... Objectively, if a photographer decides to focus only on some objects in an image, he certainly considers that the rest of the image is not the prime interest. Let consider the examples presented in figure 1.

Considering figure 1, it seems logical to limit the content of the image only on the sharp regions. For example, the crowd appearing in the images 1(a) and 1(c) is not interesting from the photographer point of view. The focus is centered on the car, the flower and on the football player.

So, the goal of our study is to propose a tool that extract directly from the image the blur regions in order to restrict the future similarity research only on the rest of the image.

This paper is organized as follows. Section 2 discusses the image segmentation methods we employ. In section 3, we present some parameters used to discriminate blurry against non-blurry regions. The experimental step is discussed in section 4 along with some evaluations we have done to show the first results of our approach. We conclude the paper in section 5.

Further author information: (Send correspondence to Jérôme Da Rugna)
Jérôme Da Rugna: E-mail: darugna@ligiv.org, Telephone: (33) 4 77 91 57 50

(a) (b) (c)

Figure 1: Some blur images.

2. SEGMENTATION PROCESS

First of all, our idea is to segment automatically the original image in order to discriminate blurry regions from the others. This step is an essential low-level one that consists in partitionning the image in visually distinct and homogeneous regions. Objectively, none method seems to be more robust in a general context, each of them being more adapted among the content of the image[1] . Moreover, with the diversity of homogeneity predicates, the segmentation is still the aim of many studies[2] .

Moreover, our objective is to detect the blurry regions in an image. Then, even if the blurry notion is difficult to apprehend and seems to be extremely relative, we can assume that a blurry region is less influenced by low pass filtering than other regions. To that effect, we have chosen two different approaches in order to segment each image : a pyramidal approach and a watershed one. We have proposed an objective and quantitative study of segmentation methods for content-based image retrieval, in an indexation oriented new approach. Results on a large collection of test images are presented[3] .

First of all, if a blurry region is less influenced by low pass filtering, it seems logical to test a pyramidal approach. The basic idea of the pyramidal structure is to produce a stack of interrelated images with progressively reduced resolution. Following the Burt's[4] approach, each level is obtained using a low pass filtering and a subsampling at the same time. Taking into account both spatial and color information, we construct this color pyramid[5] in a gamma-corrected RGB space, where the mixing is additive. The main principle of our algorithm is a top-down seeds propagation, from the lower resolutions of the image to the higher ones.

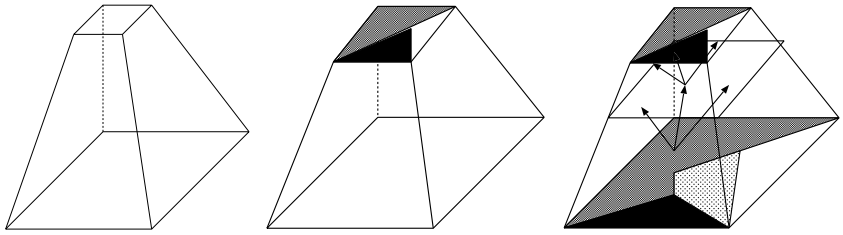
The following steps are then followed, illustrated in figure 2:

- Definition of seeds. A nonparametric technique is used at the top level of the pyramid in order to delineate arbitrarly shaped clusters in it. The computational module of the technique is the mean shift pattern recognition procedure[6] or the k-means classical one. At this step, each seed receives a different label and the previous moments are computed on each region in this manner extracted.

- Propagation through the parent-child spatial relationship between the image elements of two adjacent layers until the base of the pyramid, that is to say the original image. In this kind of pyramid, each element can belong to different parents in the upper level, the choice is made to link it to the most similar region. More precisely, a labelization is created at each level and an element can belong to 4 potential fathers, that belong to different regions. The winner is the one that is the most similar to the current element. Moreover, when the similarity is inadequate, a new region, via a new label, is then created.

Secondly, the differences between blurry and non blurry regions seem to be in the edge detection. And morphological segmentation techniques are of particular interest, through more precisely the watershed[7] algorithm. Conventional operators generally produce in fact many local minima due to noise or quantization errors. So we have implemented an improved method[8] that addresses a catchment basins merging algorithm developed to automate the segmentation of images. The proposed merging algorithm employs regional criteria to merge the non-significant minima.

Figure 3 presents some results of these two methods on mixed (blurry and non-blurry) images. Generally speaking, the watershed method gives rise to less regions. Nevertheless, it happens that both algorithms merge blurry and non blurry regions. In our approach, we expect that a “non blurry” verdict will be returned.



(a) Pyramid (b) Seeds (c) Top-down process

(d) Image (e) Top level (f) Segmentation

Figure 2: Top-down pyramidal segmentation.

3. PROPOSED DESCRIPTORS

Considering the image “well-segmented” and robust enough, at least from a blur point of view, the problem now is to propose some descriptors that are able to classify into different classes. Objectively, each descriptor will be relatively powerful and a cooperative method will be introduced. So, we first developped a large data set before realizing the learning step.

As we previous said, our basic idea is that blurry regions are more invariant according to low pass filtering. Moreover, it seems to be difficult to quantify blurry regions with absolute parameters. On the contrary, only relative evolutions between two treatments are able to be characteristic.

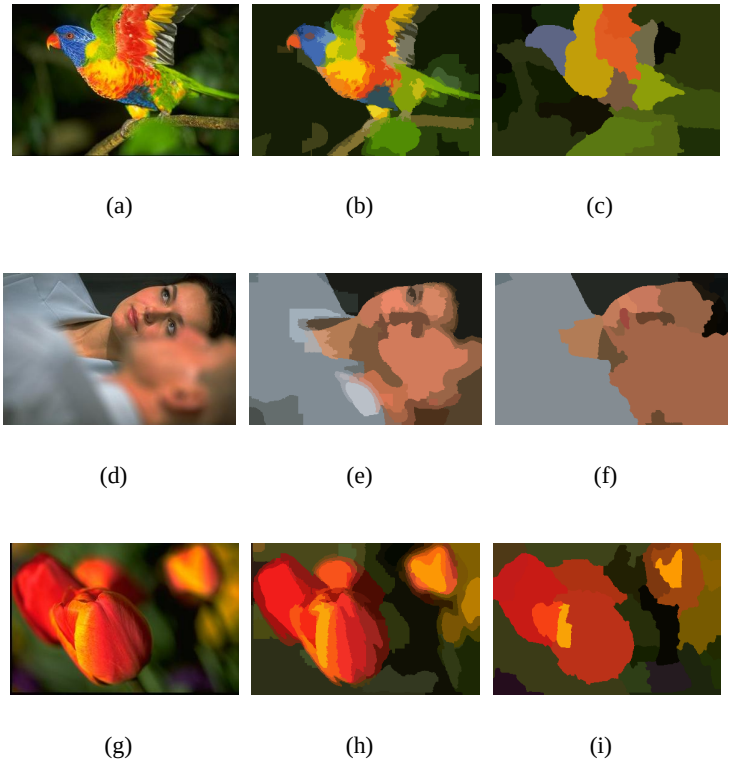
First of all, different parameters are associated to each segmented region. Then, a reference segmentation is always obtained on the original image, using one of the methods presented below.

Our descriptors are then classified into three families :

- Evolution of classical moments.
- Evolution of textural descriptors.
- Evolution of high frequencies.

3.1. Statistical moments

First of all, the first four classical moments are computed on the original image and different low-pass filtered versions (from a 3×3 to a 9×9 one).



(a) (b) (c)

(d) (e) (f)

(g) (h) (i)

Figure 3: Some segmentations using pyramidal and watershed methods (original image on the left).

For each filtered image is notably associated

$$\frac{\mathcal{M}_n^F - \mathcal{M}_n^o}{\mathcal{M}_n^o}$$

where $\mathcal{M}_n^o$ and $\mathcal{M}_n^F$ are respectively statistical n-order moments of the original image and the filtered one.

3.2. Textural descriptors

Many studies about texture can be found in the literature, from the co-occurrence matrices to the wavelet transforms. Among these options, we have retained run-length matrices[9] and its color application[10] . More particularly, these parameters suppose to first realize a quantization of the image. So, during our study, we have implemented two principal methods: one based on mean shift[6] and the other based on k-means.

Objectively, the results are different but seem to have a stable performance, very important fact in our aim. That is to say that non blurry regions are well discriminated.

Let precise the run-lengths used parameters. This matrix is of size $C \times T$ where C is the number of quantized colors and T is the maximum length in the considered direction. More precisely, a matrix is computed for each region defined by the segmentation process. Let then note $L^{\Theta}[i, j]$ the number of sections of color i and length j in the direction Θ, that is to say that j pixels of entry i in the quantization image are adjacent in the direction Θ. The most characteristic directions are in natural images 0 and 90 degrees, but we compute each direction from 0 to 315 degrees with a 45 degrees step. Among the most discriminant parameters, we have retained the following ones:

- Weight of short sections : $\frac{1}{\sum_{i=1}^{C}\sum_{j=1}^{T} L^{\Theta}[i,j]} \sum_{i=1}^{C}\sum_{j=1}^{T} \frac{L^{\Theta}[i,j]}{j^2}$
- Weight of long sections : $\frac{1}{\sum_{i=1}^{C}\sum_{j=1}^{T} L^{\Theta}[i,j]} \sum_{i=1}^{C}\sum_{j=1}^{T} L^{\Theta}[i,j] \times j^2$
- Inhomogeneity : $\frac{1}{\sum_{i=1}^{C}\sum_{j=1}^{T} L^{\Theta}[i,j]} \sum_{i=1}^{C} \left(\sum_{j=1}^{T} L^{\Theta}[i,j] \right)^2$

Each parameter is computed with $\Theta = 0$, $\Theta = 90$ and the minimum and maximum values are retained too, that is to say $\min_{\Theta \in [0..315]}$ and $\max_{\Theta \in [0..315]}$. As previously described, the evolution of each parameter computed on the original image and different filtered versions are added, so that each parameter is no more absolute but relative, improving normally its capability to discriminate blurry to non blurry regions. Objectively, the weight of short sections for example seems to be more influenced by the filtering.

In the same way, we have previously noticed the influence of the quantization step. In order to ponderate it, we have computed on each region the only number of quantized colors and the evolution through the filtering versions, assuming the fact that blurry regions are less textured. Logically, the blurry regions must be less influenced by the filtering too. These parameters are noted "number of quantized regions" and "evolution of quantized regions".

3.3. High frequencies descriptors

Image enhancement and restoration of noised and blurred images methods are common procedures intented to process an image so that the resulting processed image is more suitable. Among the large collections of methods[11–13] , we have retained the Sapiro's approach[14] , notably because of its color application. The method consists in an anisotropic diffusion algorithm. Our idea is that the difference between the original and the restored image will be more sensitive in the blurred regions.

Then, we compute first statistical moments of the differences image on each region. The two first moments are noticed to be the more characteristic.

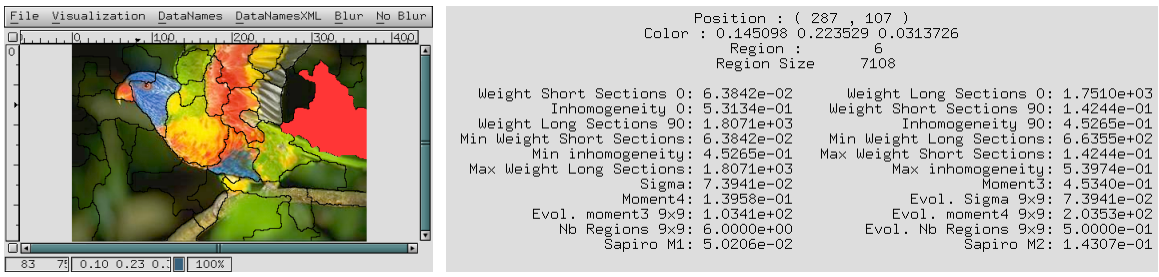
Moreover, the Fourier transform seems to be well-adapted in order to dissociate the blurry regions from the others. So, we have implemented a high-pass filtering in the Fourier domain and we have computed the first statistical moments in the same way as previously described.

4. EXPERIMENTAL STEP

4.1. The training set

Now the images are automatically segmented, the problem is to discriminate the blurry regions from the others. In order to realize this decomposition, we have first introduced an experimental reference database with many images where several blurry regions appear. These images have been choiced for their diversity according to the context. The figure 1 presents some examples, where different blur models coexist. For example, considering the sport car image, blurry regions with the crowd appear because of the movement. On the other hand the focus is only made on the flower.

More precisely, we have developped an interactive tool that present to the user the segmentation result on each image and his assignment is to classify manually each region: blurry or non-blurry. Figure 4 presents the interface of this learning step. Automatically, all the parameters presented below are computed on each region and retained.



(a) Blur Module

(b) Regions informations

Figure 4: Blur module manual selection interface.

4.2. Classifiers

Given this set of observations we need at this point to introduce learners able to classify automatically the regions. Classification by examples is one of the main machine learning problem. It is well known, there is no general rule to guide how to choose learner for a specific task[15] . It is natural to pre-select a set of classifiers in way to wonder which approach is the most adapted to our goal: blur classification. Let us describe in some words the different supervised learners hand-picked*.

- Decision Tree
  In way to realize classification, the first expecting classifier is a rules based one. We intend rules as, for example, " IF feature LESS THAN x THEN properties". Many techniques are able to extract rules from a classified distribution set, like rules induction algorithm (CN2) or decision tree methods. A decision tree is an approximation of a discrete value. It can be viewed as a partitioning of the observed space. A leaf represents a space partition of similar objects, and thus are considered to belong in the same class. The classification of an unknown item is realized by finding the leaf corresponding to this item. A rules based classification is quite the only intelligible method : an expert is able to understand a rule. The choice of C4.5[16] instead of ID-3 or CART is guided by the effective compromise between efficient results, rules size and learning time cost. Figure 5 shows a decision tree computed by C4.5 algorithm on [Blurry / Non-Blurry] regions classification problem, using the SIPINA software[17] .

- Artificial Neural Network
  We have applied a classical back-propagation neural network to classify instances, using Weka[18] library. The different parameters of the neural approach(Hidden layers, activation function,...) were set manually after several tests, in order to obtain the best accuracy and to avoid over-fitting.

- Support Vector Machine
  SVM[19] is a modern generation learning system based on recent advances in statistical learning theory. SVM is really efficient in many real-world problems, especially in the case of non-separable data. We used the SvmLight library[20] in way to compute the instance classification. Instead of Artificial Neural Network or Decision tree, SVM does not suffer from over-fitting.

- Boosting and bagging Approaches
  Boosting[21] and Bagging[22] are general methods for improving the accuracy of any given learning algorithm. Boosting can be viewed as a combination of several classifiers, produced by the same learner method, based on different conjecture on the features space. Bagging is a "bootstrap" method by training each classifier on a random redistribution of the training set. We have retained in this study the bagging method as defined in[23] . Nevertheless, the boosting (Ada-Boosting) approach gives no convincing results.

*During this work, we experienced other learners, like Naive Bayes, but these classifi ers were not effi cient, or based on a too close approach from an already choosen one

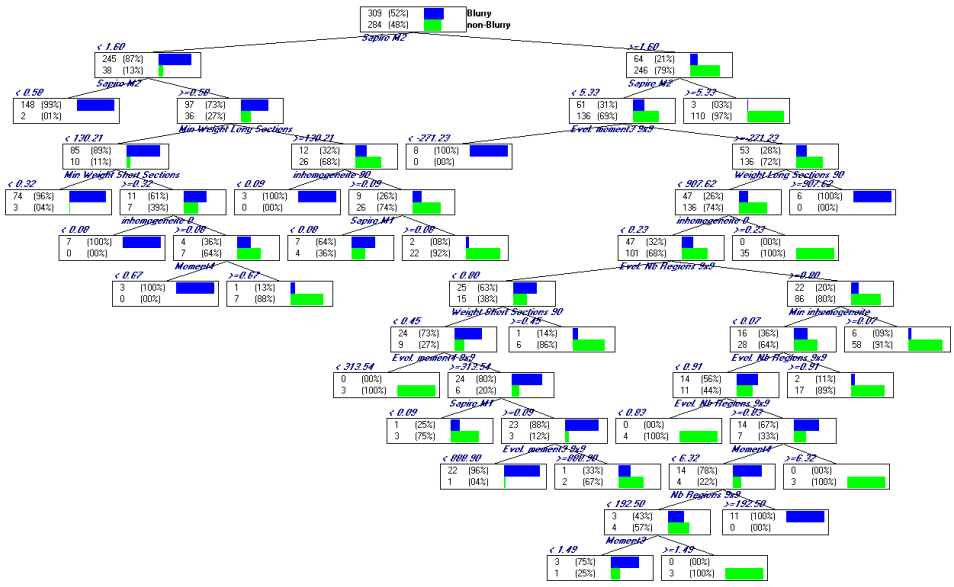


Figure 5: C4.5 output example

4.3. Evaluation

4.3.1. Effectiveness measures

To evaluate the ability of these classifiers in our context, it needs to introduce several measures of effectiveness. Let N the number of elements to classify, f_i the classifier class and a_i the real class ot the i-th element. Blur is considered as a true instance.

- Overvall efficiency of the classifier : $Accuracy = \frac{|\{i;f_i=a_i\}|}{N}$
- Precision of Blurry classification : $Precision = \frac{|\{i;f_i=a_i \& a_i=true\}|}{|\{i;f_i=true\}|}$
- Recall of Blurry classification : $Recall = \frac{|\{i;f_i=a_i \& a_i=true\}|}{|\{i;a_i=true\}|}$
- False Positive rate : $FP = \frac{|\{i;f_i=true \& a_i=false\}|}{|\{i;a_i=false\}|}$

We can note that the False Positive rate is really important in our case. It measures the percentile of non-blurry regions classify as blurry. If we consider that blurry regions contain less information than non-blurry ones, we expect the classifier to not assign to "blurry verdict" regions with effective information. Our goal would be, in fact, to get the best FP rate possible, even if the $Recall$ rate does not afford the awaited results.

4.4. Cross Validation

To estimate the final efficiency of a method, we wonder an estimation method with bias, low variance and handling over-fitting. Between the two well-known accuracy estimation methods, we choose cross-validation[24] instead of bootstrap. Cross-Validation is intelligible, simple to implement and, in many case, is the more effficient. We have selected a 10-Fold cross-validation.

4.5. Results

These presented results are computed of the same image training set. It includes more than 40 various images with blurry and non-blurry regions. The training set size is about 600 regions.

The tables 1[†] and 2 present results in two different cases :

- All features bench: The learner method works on every feature, without weighted approach. The real vector dimension is about 80.
- Selected features: The learner method works on a specific set of features. This one is produced by selecting the most correlated features with the blurry/non-blurry class. The real vector dimension is about 10.

Excluding SVM, the first observation is that all methods give similar results. The FP rate is arround 10%, which is, in fact, an interesting result. As shown in figure 6, where the blurry regions are black painted, the detection is visually efficient, as False Positive regions are non-blurry regions, but are not regions containing a fist-full of informations. These images are obtained with C4.5 rules, obtained without bagging (not best possible learner) using selected features. Naturally, these images were not included in the training set. C4.5 is, in our context, the best compromise between efficient results, time cost, and, it is important, give intelligible rules to the expert. We present in figure 6, the original image, the image with black painted blurry regions, and, in third column, black painted blurry region where the accuracy of the region leaf (C4.5 tree) reach a threshold, 90% in this instance. Using this option, called "Min accuracy", the FP rate is, as expected, really low.

The poor results of SVM were not expected, we cannot explain it, we can only suppose this approach is not adapted to our data. However, the least quality considering all features is explained, in the case of C4.5 and ANN, by the over-fitting problem: we need, of course, to weight parameters and to delete one-to-one correlated features.

Method	C4.5	Bagging C4.5	ANN	SVM
Accuracy	80.7%	88.3%	85.1%	68.8%
Precision	82.1%	89.1%	87.5%	81.6%
Recall	82.8%	89.7%	89.3%	51.7%
FP	18%	10.5%	19%	14.0%

Table 1: 10-fold Cross-Validation : All features.

Method	C4.5	Bagging C4.5	ANN	Bagging ANN	SVM
Accuracy	84.3%	88.0%	85.3%	88.4%	61.0%
Precision	86.7%	88.64%	87.8%	89.5%	80.2%
Recall	82.5%	88.35%	84.4%	88.0%	38.2%
FP	13.1%	9.8%	12.12%	10.2%	14.1%

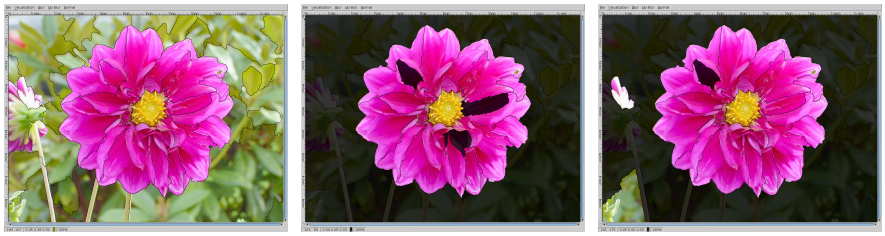
Table 2: 10-fold Cross-Validation : Selected features.

5. CONCLUSION AND FUTURE WORKS

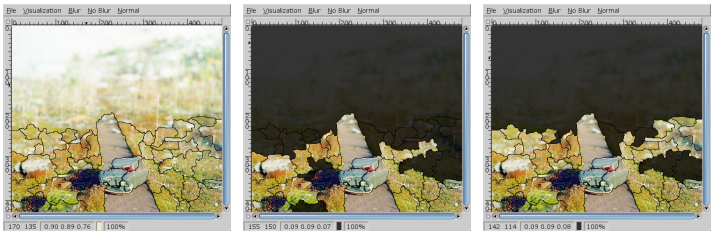
We have presented feature sets for achieving a blur discrimination in order to be used in a content-based retrieval tool. The first experiments indicate that previous introduced decision rules permit to reach our goal quite accurately. Nevertheless, this is still an early step towards a real detect use in a searching tool. The next steps will be for instance to add wavelets descriptors and spatial relationship (through emergency definition) in order to develop not only a binary verdict but now a continuum from absolutely not to certainly blurry.

Our aim is then to integrate these kind of meta data directly in the graphics format in order to use it during the content-based retrieval request. For example, it must offer the possibility to ponderate differently the blurry regions from the others. Regarding the inductive semantics included in these kinds of meta-data, we surely be able to improve existant tools.

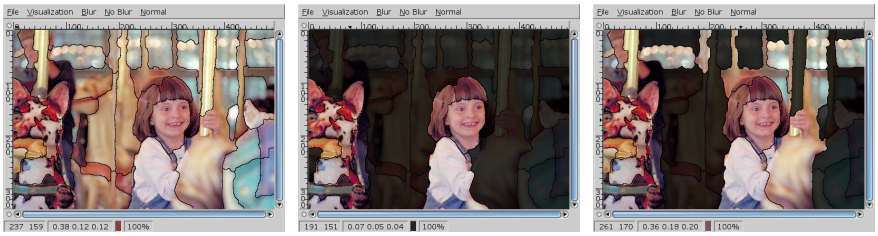
[†]Baggging ANN is not presented in All features bench, as the time cost for this approach in this case is too important.



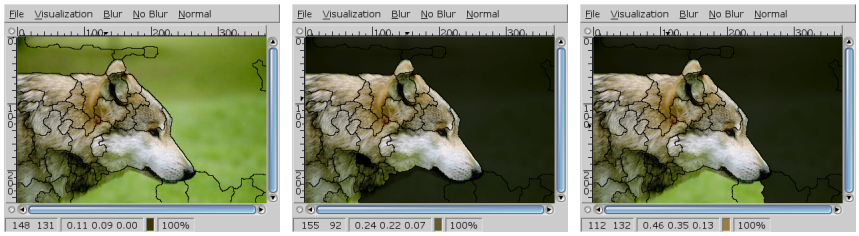
(a) Original Image (b) Blurry Regions (c) Blurry Regions with min accuracy



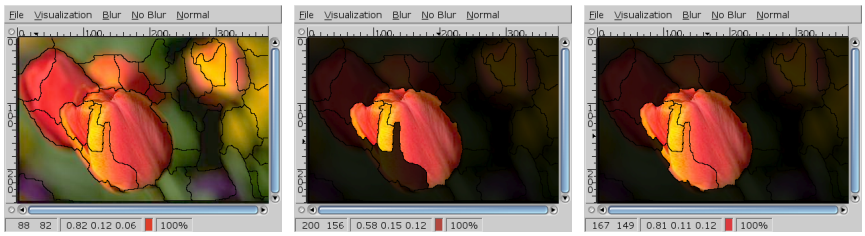
(d) (e) (f)



(g) (h) (i)



(j) (k) (l)



(m) (n) (o)

Figure 6: Some blur extraction examples.

REFERENCES

1. J. P. Cocquerez and S. Philipp, *Analyse d'images: filtrage et segmentation*, Masson, 1995.
2. N. R. Pal and S. K. Pal, "A review on image segmentation techniques," *Pattern Recognition* **26**, pp. 1277–1294, 1993.
3. J. Da Rugna and H. Konik, "Comparison of different segmentation algorithms for indexation oriented approach," in *RFIA*, 2004. to be published.
4. P. J. Burt, "Fast filter algorithms for image processing," *Computer Graphics and Image Processing* **16**, pp. 20–51, 1981.
5. H. Konik, V. Lozano, and B. Laget, "Color pyramids for image processing," *Journal of Imaging Science and Technology* **40**, pp. 535–542, 1996.
6. D. Comaniciu and P. Meer, "Mean shift: a robust approach toward feature space analysis," *IEEE Transactions on Pattern Analysis and Machine Intelligence* **24**, pp. 1–18, 2002.
7. L. Vincent and P. Soille, "Watersheds in digital spaces: an efficient algorithm based on immersion simulations," in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **13**, pp. 583–598, 1991.
8. M. C. D. Andrade, G. Bertrand, and A. A. Araújo, "Segmentation of microscopic images by flooding simulation: a catchment basins merging algorithm," in *Proceedings of SPIE Symposium on Electronic Imaging*, **3026**, pp. 164–175, 1997.
9. M. Galloway, "Texture analyss using gray-level run length," in *Computer graphics and image processing*, **4**, pp. 172–199, 1974.
10. C. Vertan, M. Ciuc, V. Buzuloiu, and C. Fernandez-Maloigne, "Compact color-texture run-length description for ornamental stones recognition and indexing," in *The Hyperion Scientific Journal*, **3A**, pp. 69–75, 2002.
11. J. A. S. Centeno and V. Haertel, "An adaptive image enhancement algorithm," in *Pattern Recognition*, **30**, pp. 1183–1189, 1997.
12. K. Deguchi, T. Izumitani, and H. Hontani, "Detection and enhancement of line structures in an image by anisotropic diffusion," in *Pattern Recognition Letters*, **23**, pp. 1399–1405, 2002.
13. L. M. Kennedy and M. Basu, "Application of projection pursuit learning to boundary detection and deblurring in images," in *Pattern Recognition*, **33**, pp. 2019–2031, 2000.
14. G. Sapiro and D. L. Ringach, "Anisotropic diffusion of multivalued images to color filtering," in *IEEE Transactions on Image Processing*, **5**, pp. 1582–1585, 1996.
15. T. M. Mitchell, *Machine Learning*, McGraw-Hill, New York, 1997.
16. J. R. Quinlan, "Improved use of continuous attributes in c4.5," *Journal of Artificial Intelligence Research* **4**, pp. 77–90, 1996.
17. R. Rakotomalala, "Sipina software." http://eric.univ-lyon2.fr/ ricco/sipina.html.
18. S. Garner, "Weka: The waikato environment for knowledge analysis," 1995.
19. V. Vapnik, *The Nature of Statistical Learning Theory*, NY, Springer-Verlag, 1995.
20. T. Joachims, "Making large-scale svm learning practical. advances in kernel methods - support vector learning," 1999.
21. R. E. Schapire, "A brief introduction to boosting," in *IJCAI*, pp. 1401–1406, 1999.
22. J. R. Quinlan, "Bagging, boosting, and c4.5," *Thirteenth National Conference on Artificial Intelligence - Eighth Innovative Applications of Artificial Intelligence Conference* , pp. 725–730, August 1996. AAAI Press / MIT Press.
23. L. Breiman, "Bagging predictors," *Machine Learning* **24**(2), pp. 123–140, 1996.
24. R. Kohavi, "A study of cross-validation and bootstrap for accuracy estimation and model selection," in *IJCAI*, pp. 1137–1145, 1995.