Multi-Scale Spectral Residual Analysis to Speed up Image Object Detection

Grimaldo Silva, Leizer Schnitman, Luciano Oliveira
Programme of Post-graduation in Mechatronics
Intelligent Vision Research Laboratory, UFBA
{jgrimaldo, leizer, lrebouca}@ufba.br

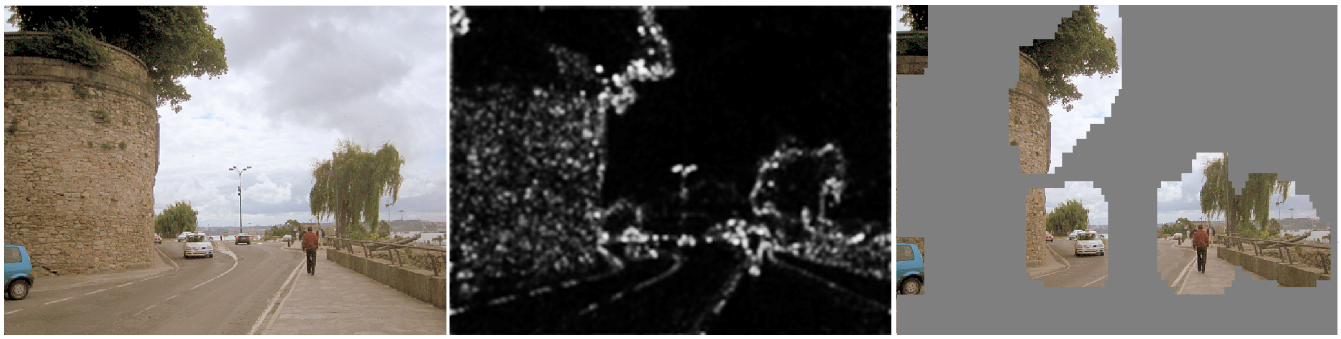

Fig. 1. From left to right: original image, saliency map, candidate regions in the saliency map. A very usual approach to search for an image object is sliding window, which performs a dense search in image space. By using a multi-scale saliency map, we are able to tease out image regions which are likely unnecessary for object search when sliding image windows. After that, a detector can be attached to only selected regions, allowing faster object detectors.

***Abstract*—Accuracy in image object detection has been usually achieved at the expense of much computational load. Therefore a trade-off between detection performance and fast execution commonly represents the ultimate goal of an object detector in real life applications. In this present work, we propose a novel method toward that goal. The proposed method was grounded on a multi-scale spectral residual (MSR) analysis for saliency detection. Compared to a regular sliding window search over the images, in our experiments, MSR was able to reduce by 75% (in average) the number of windows to be evaluated by an object detector. The proposed method was thoroughly evaluated over a subset of LabelMe dataset (person images), improving detection performance in most cases.**

***Keywords*-multi-scale spectral residue, saliency, person detection**

I. Introduction

Image object localization has been reaching remarkable results in real life applications. However, the more accurate is the method, the heavier it is with respect to computational cost. Achieving the best trade-off between detection performance and computational cost usually represents a challenging task. Indeed, in many practical situations, object detection requires on-the-fly execution in order to be feasible in practice. Among these time-critical tasks, there are: perception for driver assistance [1], video traffic analysis [2] and surveillance systems [3]. If we still consider the current availability of high resolution images, which demands additional processing time, the mentioned trade-off presents an even bigger challenge.

To cope with the aforementioned trade-off problem, many methods have been proposed. Zhu et al. [4] and Viola and Jones [5] have developed rejection cascades, reducing the time required to detect non-objects. These works were based on the so called sliding window search. Toward methods to avoid or to reduce the overhead of a dense search, saliency detectors have demonstrated promising results. As saliency detectors are able to locate regions of interest in images, they can be used in a broad spectrum of applications – from thumbnail generation [6] to semantic colorization [7]. Examples of such saliency methods are found in [8], which uses statistical properties of natural scenes to select regions of interest, and also in [9] based on the computation of saliency inspired on the pre-attentive phase of human visual system, responsible for drawing attention to specific parts of the visual stimuli.

The positive traits of saliency methods on search space reduction allowed Ip et al. [10] to make a saliency analysis in very large images in order to assist human visualization by means of possible regions of interest (ROI). ROI are found through a difference of Gaussians at multiple image scales[1]. Likewise, Rutishauser et al. [11] proposed an object recognition (among grocery items) based on the saliency method found in [9] and a scale invariant feature transform (SIFT) keypoint detector [12]. First, the saliency method is applied to determine the most likely areas to have an object; instead of thresholding the saliency map generated in the first

[1]Throughout the text, the words 'octave' and 'scale' are used interchangeably.

step, a region growing segmentation defines the best object hypothesis; at the end, image object silhouette is delineated by means of the keypoints detected over the salient areas. Feng et al. [13] address the problem of object detection using a sliding window over an image, specifying each window saliency as the cost of composing it with remaining parts of the image; therefore the image is segmented into regions based on similarity; the difference between regions is calculated over LAB histograms and spatial distances; these features are then used to select the most differentiated windows which hopefully present the most salient objects.

On the reduction of image search space, Lampert et al. [14] propose the use of a branch-and-bound optimization applied on the score of the classifier, which is used to separate input space. The method was called Efficient Subwindow Search (ESS). The target function is subjected to maximize the classification score whereas minimizing the number of windows evaluated by a detector. In its original form, that method only detects one object per image, but it can be modified to search for multiple objects. ESS effectively reduces the number of evaluated windows over the image in contrast to regular sliding window based detectors [5], [15], [16].

Following all these ideas, the multi-scale spectral residue (MSR) analysis aims to speed up sliding window-based object detection by spectral residual analysis on multiple scales. Our method relies on a sliding window approach based on the image saliency with the goal of assigning a score to each window before object detection stage (see Fig.). Although Feng et al. [13] also assign a saliency score to each window, our approach has some important differences. MSR computes an image-wise saliency following the rationale in [8], in a more flexible way, allowing saliency detection in the original image aspect ratio. Additionally, we explore properties of the frequency domain to extract interesting regions in contrast to the use of spatial properties such as composability of segments. MSR differs from ESS in the requirements and methodology. ESS avoids a dense detector search by using an optimization method that requires a linear classifier and local image descriptors such as [12]. MSR does not impose such constraints, and can be used on most sliding window based detectors by relying solely on an object saliency. Our approach also avoids assumptions about an object shape to reduce the search space, as such, it does not attempt to segment an object based on salient locations, as in Rutishauser et al. [11]; instead, MSR indicates regions of interest and relies on a classifier for actual object detection and localization. Recent solutions of rejection cascades [4] [5] in a sliding window search can easily be integrated to MSR. This latter can be combined with MSR in order to achieve faster processing time.

This work is structured as follows: an overview of saliency detection methods is given in Section II. Section III describes MSR and a methodology to evaluate the impact of window selection on detector performance. In Section IV, the MSR is compared to other saliency methods, and its runtime and detection performance are measured over a person dataset. Finally, overall conclusions are drawn in Section V.

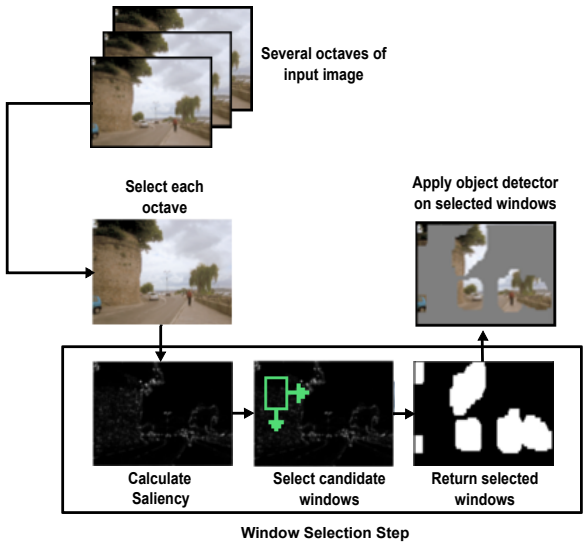

Fig. 2. Overview of MSR. For each octave of the original image, the saliency map is computed, and a sliding window is applied on the saliency map. Candidate windows are selected according to their scores given by a quality function. Finally, an object detector is applied only in the candidate windows.

Contributions: Our contribution resides in a novel method, called MSR, with the aim of achieving a better trade-off between the number of windows selected to be evaluated by a detector, and the number of miss detections. MSR has demonstrated an average reduction of 75% of windows to be evaluated, while keeping or improving detection performance.

A. Proposed method at a glance

When performing a dense search for an object, only a small subset of the image might contain objects. However, sliding window based detectors are only able to provide image object localization after running a classification function over each window on multiple orientations and octaves (scales). For that, the use of a full-fledged object detector implies an expensive operation, requiring preprocessing, feature extraction and classification. In order to reduce the number of windows which will be evaluated by a detector, we propose a bottom-up saliency approach to select windows of interest before running the detector in each window. Although MSR has been motivated by [8], it was conceived to overcome some limitations of that method when used on uncontrolled scenes. These improvements are listed below:

1) resizing each image octave by a constant resizing factor – 15% of its size, instead of making assumptions about object scale by using a fixed image size for saliency detection. This change allows search of salient objects at multiple scales;
2) choice of threshold k for region selection is not dependent on each image saliency map, but on a constant global value based on a trade-off between selected regions and false negatives (FN) in the classification. In [8], the threshold is calculated as $k = 3 \cdot E(S(x))$, or three times the mean saliency map $S(x)$ intensity. How-

ever, this latter formulation incorrectly regards objects in cluttered images (many objects) as non-salient.

3) saliency quality in a region is calculated from a window-wise saliency mean, instead of using pixel values individually as in [8], allowing detection of entire objects even when their saliency is non-uniform along its length.

Instead of relying on the object detector to choose the most likely image region to contain an object (just after obtaining the saliency map), windows are slid over an integral saliency space. This latter step corresponds to computing the integral image of the pixels in the saliency space in the same way as Viola and Jones [5]. After that, a quality function $f(\cdot)$ is applied at each window w, providing a score. The score of a given window is calculated using the mean of its saliency intensity, and a window is selected if its score is greater than or equal to a threshold k. A higher k selects smaller number of windows, while potentially missing more true positive (TP) detections in the further steps of the method. Conversely, as value of k gets lower, MSR approaches to a method based on regular sliding window search. An overview of MSR mechanism for window selection is summarized in Fig. 2.

II. Overview of saliency detection approaches

An object draws more attention when it has a strong contrast in relation to its neighbourhood, objects such as traffic signs or a stop light were created to explore this property in order to be perceived faster than surrounding objects. While an attention mechanism can help a person focus on specific objects in a scene, in a similar way, an algorithm capable of detecting salient objects in images must search for characteristics such as visual uniqueness, rarity and unpredictability [17]. This is so in order to correctly highlight image regions which demand extra attention. Following these ideas, we briefly summarize some of saliency detectors:

***Itti's method (IT)**:* Among the first salient methods, a biologically inspired approach was developed by Itti et al. [9]. In that approach, saliency of a given pixel is calculated based on its uniqueness in relation to local surroundings. Uniqueness is defined on the analysis of color, intensity and orientation over multiple scales. After that, these features are then normalized and combined in a way where channels with larger contrasts are preferred.

***Graph based (GB) visual saliency**:* Similarly to Itti, Harel et al. [18] form activation maps from particular feature channels, and normalize them to better highlight salient regions.

***Frequency tuned (FT) saliency region detection**:* Instead of using local information to define the saliency, Achanta et al. [19] define saliency of a pixel as its distance from the image pixel mean on LAB space, formally represented as

$$S_a(x, y) = ||\mathbf{I}_\pi - \mathbf{I}(x, y)||_2 \,, \tag{1}$$

where $\mathbf{I}_\pi$ is the mean image feature vector, $\mathbf{I}(x, y)$ is the original pixel value, $||\cdot||_2$ represents an L_2 norm where each pixel is a feature vector of type $[L, a, b]$.

***Luminance contrast (LC)**:* Also using global contrast, Zhai and Shah [20] developed a method for pixel-level saliency detection using the contrast of a pixel with respect to the others in a scene. It is given by

$$S_z(I_k) = \sum_{\forall I_i \in I} ||I_k - I_i|| \,, \tag{2}$$

where I_i and I_k are pixels in the image and $||\cdot||$ represents the Euclidean distance.

***Spectral residual (SR)**:* Similar to global methods, frequency based approaches also explore properties of the entire image. Hou and Zhang [8] used these properties based on $1/f$'s law, which states that an ensemble of images on the Fourier Spectrum obeys the distribution

$$E\{A(f)\} \propto 1/f \,, \tag{3}$$

where $A(f)$ is the amplitude averaged over orientations, and f is a given spectrum in the frequency domain. Whilst objects do not follow properties of natural scenes, detection of potential salient points is based on finding statistical singularities on the spectrum of an image. These singularities are called spectral residues.

III. Pruning windows by multi-scale spectral residue

Saliency detectors are able to associate a degree of local or global uniqueness for each image pixel (or group of pixels). This information is useful to help pruning undesired windows. In this regard, during a search for objects via sliding windows, the capability to choose whether a detector will evaluate a particular window or ignore it (based on its object likelihood) can bring benefits to speed up the classification task in further steps.

Saliency detectors face additional complexities when dealing with uncontrolled scenes, such as variations in object (color, size, illumination and noise). Particularly, it is noteworthy that spectral residual (SR) analysis [8] is susceptible to those factors when selecting image ROI, since an object may have intense intra-variability. To avoid that, in MSR, saliency is measured in a per-window basis, and the saliency of a window is defined as the mean intensity of its salient pixels, enabling higher resilience to variability of salient pixels.

Another limitation of SR in the context of aiding object detectors is its threshold for region selection, defined as $k = 3 * E(S(x))$ where $E(S(x))$ denotes the mean value of the saliency map and k the threshold. Such scheme expects that images have but a small number of salient regions. If it is not the case, that method potentially excludes important objects because of the high lower bound. Given that situation, we define the threshold k as a constant value throughout the entire collection of images, representing an average trade-off between the number of selected windows and false negatives (FN) caused by window selection.

From the aforementioned improvements, summarized on Fig. 3, the underlying concepts required for multi-scale analysis have been conceived in Section III-A.

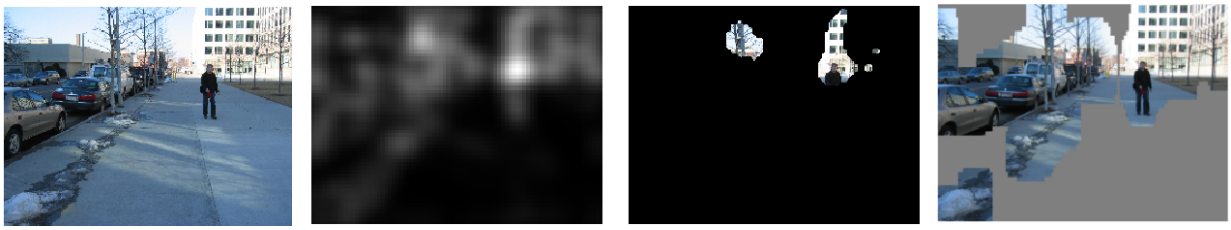

Fig. 3. Comparison between SR and MSR. From left to right: the original image, SR saliency map, region selection using SR formulation in original image, and MSR window selection at a particular octave.

(a) SR with image at 15% of its size

(b) SR with image at 7% of its size

Fig. 4. Differences in saliency at multiple scales. In 4a, SR was calculated in 15% of the original image size, generating strong reactions on mostly small objects; in 4b, using 7% of the original image size, bigger objects were also selected. The image reduction examples demonstrate how the image size influences on the scale of saliency detection, which will be tuned to best select objects in a given octave.

A. Multi-scaling the spectral residue

Most saliency methods are able to detect objects of different sizes. Methods such as [9] and [18] make direct use of feature analysis at multiple image scales to achieve that result. In contrast, SR searches objects at a single scale, which is specified based on a estimation of common object sizes over normal visual conditions [8]. For that, SR cannot be used in an uncontrolled multi-scale environment, as the saliency detector will not search for objects at the same scale as the object detector. Because of that, it was necessary to establish a connection between the search scale of the object detector and the saliency detector.

The scale of salient objects in SR is implicitly defined by the image size. Therefore smaller objects are more salient on bigger images, because the smaller an image gets, the bigger are the objects that become salient, as depicted on Fig. 4. In this case, searching for salient objects with various sizes has a strong relation to how a sliding-window based object detector searches for bigger objects in an image using a fixed size window. This search is accomplished by resizing an image at a fixed compound rate, such as $I_{i+1} = R(I_i, s)$, where R represents the resize function, I_i denotes the i-th image octave and s the resizing factor; the detector thus slides the detection window over each octave i.

As we focus on detection of saliency and objects within the same search scale, using a fixed-size window, we may conclude that from a particular octave I_i, there is a constant resizing factor β capable of adjusting the two detectors to the same scale. Given a value of β, saliency detection will be executed on each octave i over a reduced image, $R(I_i, \beta)$, with its color histogram normalized. This histogram normalization is applied to increase object contrast, enhancing the overall saliency of the object against the scene. Another practical use of further resizing the image using β is to reduce the computational load of saliency calculation. Defining a specific value for β will depend on factors such as: object of interest, scale of search and saliency detector. A β value of 0.15 was chosen based on experimental data. The choice of this value is discussed in detail in Section III-C.

After obtaining the image octaves, and consequently the generated saliency maps for each octave, a quality value $f(w)$ for each window w was calculated from its mean saliency intensity. To speed up mean computation, the quality value $f(w)$ is calculated after computing the integral image of the saliency map (having then mean calculation with constant time complexity).

B. Determining the quality function threshold

Proper evaluation of window selection impact on performance was done by means of an analysis of the window selection rate (WSR) and saliency false negative rate (SFNR). WSR denotes the number of windows selected for further processing, while SFNR represents how many objects the detector failed to recognize after MSR pruning.

Both WSR and SFNR depend on a threshold k which represents a minimum score for a window to be selected for actual object detection. Thus, given that W is the set of all windows generated from sliding on the entire collection of images at every scale and M the set of all objects of interest from this same collection of images, we can calculate the trade-off between $\mathrm{WSR_k}$ and $\mathrm{SFNR_k}$ in a five-step process. First, we define the set of selected windows S_k as

$$\mathrm{S_k} = \{w \in W \mid f(w) \geq k\}\,, \tag{4}$$

where $f(w)$ is the quality value of a window w and k is the threshold for window selection. Given S_k, it is possible to calculate the window selection rate with

$$\mathrm{WSR_k} = \frac{n(\mathrm{S_k})}{n(W)}\,, \tag{5}$$

where $n(\cdot)$ denotes cardinality of a set. To calculate the $\mathrm{SFNR_k}$ one should enumerate for each object $j \in M$ the number of windows in which the object was correctly matched, given by

$$\mathrm{C_{k,j}} = \{w \in S_k \mid o(w) = j\}\,, \tag{6}$$

where $o(w)$ is a function that, in case an object exists at window w, and this is correctly classified by a detector, returns the matched object from set M; otherwise $o(w)$ returns any element $\notin$ M. From that, its trivial to find the set of objects detected, F_k, defined as

$$F_k = \{j \in M \mid n(C_{k,j}) \geq 1\}\,. \tag{7}$$

Finally, in order to calculate how many miss detections were caused by the saliency (SFNR), we use

$$\mathrm{SFNR_k} = \frac{n(F_{k_{\min}}) - n(F_k)}{n(F_{k_{\min}})}\,, \tag{8}$$

where $k_{\min}$ is the minimum threshold value, which guarantees $S_{k_{\min}} = W$. Thus, to generate a full trade-off curve, this process is repeated for each $k \in K$ where K is the set of unique window scores.

C. Parameter choice

The proper choice of value for β will change according to the scale and characteristics of a given object. For person detection, the best value for β was found to be 0.15. This was achieved over the LabelMe [21] dataset for persons (see Section IV-A for more detail). Figure 5 shows the trade-off of WSR and SFNR for different values of β.

A possible consideration is to use the parameter β only in the original image (full resolution). It would save processing time dedicated for calculation of the saliency at each scale. However, multi-scale methods had dominant superior performance in our tests, as can be noted in Fig. 6.

Henceforth, to facilitate result analysis, we focus on the operating points of 20% and 30% of WSR. The choice of these operating points intends to evaluate a preferable runtime performance (20% of WSR) in spite of detection performance, or to keep detection performance (30% of WSR) with acceptable speed gains.

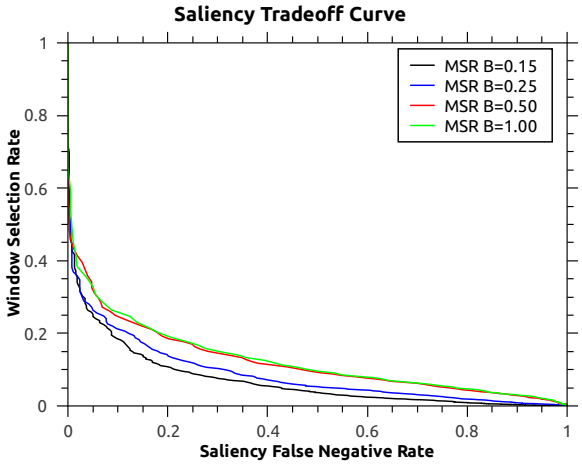

Fig. 5. Trade-off curve for person detection using different β values. When the curve is closer to the origin it is better.

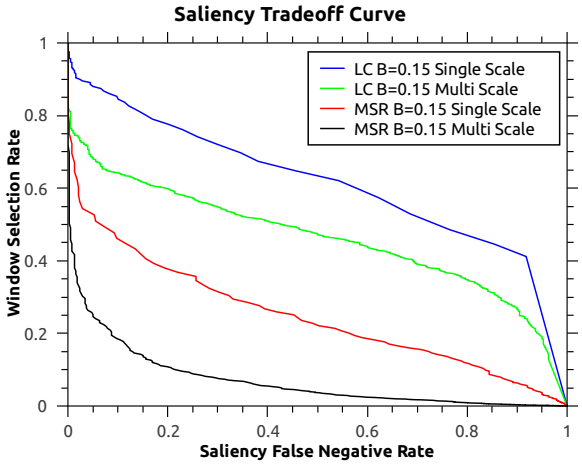

Fig. 6. Comparison between multi-scale analysis and using the same saliency map for all scales. Methods presented are MSR and LC [20]. When the curve is closer to the origin it is better.

IV. Experimental Evaluation

A. Methodology

Evaluation of MSR was accomplished by a four-step analysis: (i) comparing the detection performance considering several saliency detection methods using the same sliding window parametrization in a multi-scale analysis; (ii) analysing MSR scalability with respect to detection, i.e., how it behaves on different image resolutions; (iii) how MSR affects a detector receiver operating characteristic (ROC) curve with respect to a regular sliding window, and, finally, (iv) impact on detection runtime speed with different parameters.

To standardize comparisons, a set of 330 images was extracted from the LabelMe [21] dataset. Image sizes range from 320 by 240 to 2592 by 1944. Additionally, the dataset encompasses several environments, including city, snow, forest and river, where each scene contains at least one person.

For all analyses using the aforementioned dataset, a com-

TABLE I
SFNR FOR EACH METHOD AT 20% OF WSR

Method	β 0.15	0.25	0.50	1.00
MSR	08.73%	11.38%	17.99%	17.99%
LC	93.92%	93.92%	94.18%	94.43%
FT	87.83%	82.54%	76.46%	77.25%
GB	-	-	-	47.09%
IT	-	-	-	38.89%

bination of histogram of oriented gradients (HOG) [16] and Support Vector Machine (SVM) was used as the method to classify persons. The rationale of using HOG/SVM was not only because it is a state-of-the-art detector, but also to facilitate comparison with other future search reduction methods, since its source code is publicly available. Our HOG/SVM detector was trained using a person dataset distinct from the one created with images from LabelMe. Addtionally, for the sliding window, the detector was set up with window size of 64 by 128 pixels, a stride of 8 pixels horizontal-wise, and 16 pixels vertical-wise, and image resizing rate of 0.96, for each octave.

It is noteworthy that, for analysis (i), two state-of-the-art saliency methods have not been included – [22] and [23]. The former, because the saliency detection is concentrated mostly on images with a single and clear salient object; the latter, because of its very slow runtime speed. In (ii), we examine how MSR performance changes over different image resolutions and how each image octave contributes to its results. This experiment is important to considering recent increases in availability of high resolution images. In (iii), we built a ROC curve to show the effects of MSR on a person detector at different WSR configurations in comparison to a normal sliding window. Finally, in (iv), we evaluate if the number of windows discarded before detection is sufficient to compensate for the additional processing required by MSR.

B. Comparison of saliency methods in a multi-scale structure

A comparison of MSR against other state-of-the-art methods in the same multi-scale structure is presented in Fig. 7. As the results represent only the best configuration of each method, a more detailed information is organized on Table I and II.

In these experiments, both IT [9] and GB [18] were only evaluated using the original octave size, with no β scaling, as these methods already perform analysis at multiple scales internally. We also did not compare MSR with the original SR [8] since the latter one was designed to operate on a single image size.

The results indicate that MSR achieved superior performance on almost the entire trade-off curve. In addition, SFNR at 20% and 30% of WSR were at least ten times better when compared to other methods. Yet, at 50% of WSR, the SFNR is close to zero (less than 0.3%), which indicates that a detector could process images twice as fast with a negligible loss in TP.

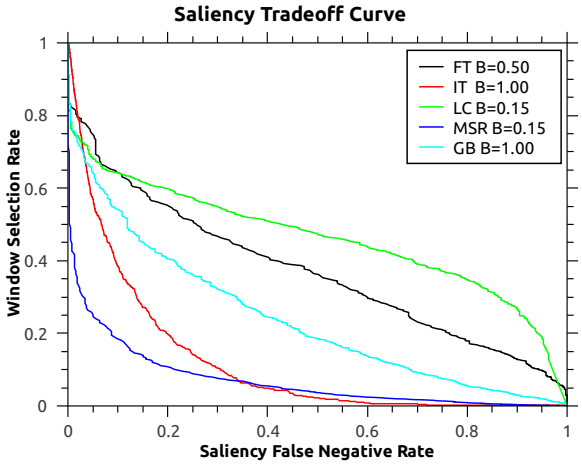

Fig. 7. Comparison of best results from different saliency methods applied to guide multi-scale detectors, the methods are MSR, FT [19], GB [18], IT [9], LC [20]. The false negative rate represents only objects that the detector would have matched if a regular sliding window approach had been used. When the curve is closer to the origin it is better.

TABLE II
SFNR FOR EACH METHOD AT 30% OF WSR

Method	β 0.15	0.25	0.50	1.00
MSR	02.91%	02.64%	06.08%	05.55%
LC	86.24%	86.24%	85.98%	87.04%
FT	70.37%	64.29%	59.79%	62.69%
GB	-	-	-	33.86%
IT	-	-	-	23.81%

The MSR and IT methods had the best overall trade-off between WSR and SFNR. Both SR and IT were among the worst on recent evaluation of general purpose saliency detection found in [17]. Some possible causes of this discrepancy are:

1) saliency detection is done in a single scale in [17], while in our tests the saliency was recalculated at every octave. This provided a better performance for most methods;
2) differences in scene selection. The dataset used in [17] was gathered by [19] with images containing mostly uncluttered objects and natural background. In our tests, images were extracted from LabelMe [21], wherein the images contain a wide range of locations and varying degrees of clutter;
3) little background information on some images (large objects).

C. Scalability

We compare how well MSR can select image windows at different starting image resolutions in Fig. 8. From this information, we can conclude that increasing image size allows for an even better trade-off between SFNR and WSR.

To further confirm the scalability of MSR on larger resolutions, we compare its ability to eliminate windows at a fixed threshold in several octaves in Fig. 9, showing that

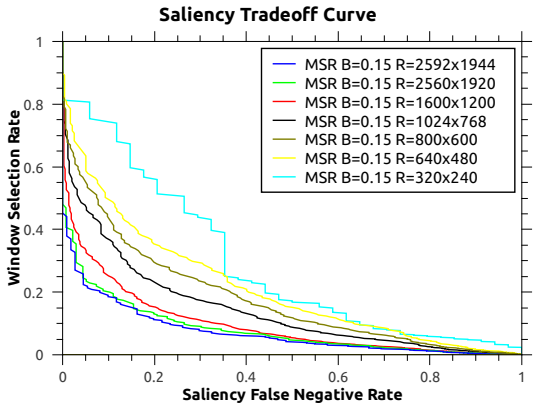

Fig. 8. Trade-off between WSR and SFNR at different starting resolutions. Aspect ratio is kept by approximating the image resolution to the closest image size. When the curve is closer to the origin it is better.

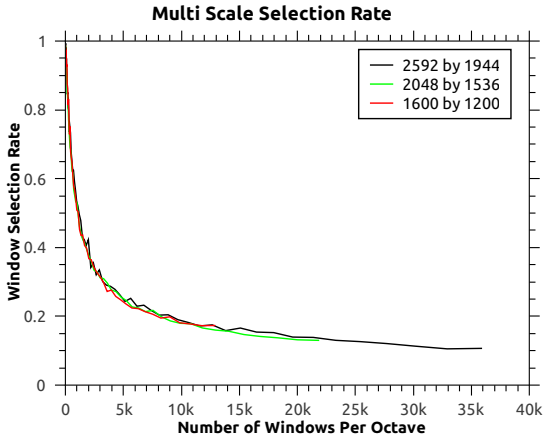

Fig. 9. Relation between number of windows at each image size and number of windows selected for the detector. An operating point was selected at 20% of WSR (see Fig. 7 for reference).

larger size images contribute for better WSR. In this test, the number of windows in each octave is calculated with $1 + [(w_i - w_w)/s_h] * [(h_i - h_w)/s_v]$, where w_i and h_i are the image width and height, respectively, w_w and h_w are the window width and height, s_h is the horizontal stride and s_v is the vertical stride.

D. Detection performance

Comparison between an object detector with and without MSR is presented on Fig. 10. In the tests, MSR at 20% of WSR provided greater TPR than regular sliding window within the range of 0 and 1.48 of FPPI. At 30% of WSR and within its range of 0 and 1.98 of FPPI, our method also obtained larger TPR than a regular sliding window approach. The maximum TPR of a regular sliding window is 0.71, while for MSR at 30% of WSR the maximum is 0.69. Even though the difference was small to match the actual maximum TPR of a regular sliding window, MSR operated at least on 50% of WSR, which still represents a twice as fast image processing with only a negligible performance loss (less than 0.3%).

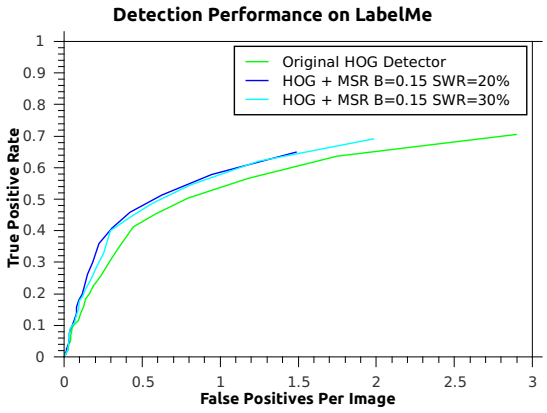

Fig. 10. ROC curve showing differences between person detection performance using a regular sliding window and MSR. Both methods use the same HOG+SVM detector.

Some examples of positive and negative results at 30% of WSR can be found, respectively, on Fig. 11 and Fig. 12.

E. Runtime performance

In order to evaluate MSR runtime speed, a comparison was performed with the traditional sliding window HOG detector. We summarized the results on Table III. Time was calculated as the proportion of the total detection time for a specific WSR value of a regular sliding window execution.

Expected gain, considering elimination of 80% and 70% of windows to be classified, was 5x and 3.3x. However, the results demonstrated that for both 19.9% and 29.6% of SWR[2], the actual runtime speed gain was smaller then 4.8x and 3.2x. This indicates that MSR window selection mechanism imposed only a small processing overhead for each window, which was compensated by the large number of windows discarded.

TABLE III
RUNTIME SPEED PROPORTION FOR EACH METHOD

Method	WSR	Total Time Proportion	Avg. Time Proportion Per Window
Regular Slide	100%	1.0000	1.0000
MSR $\beta = 0.15$	19.9%	0.1932	0.1996
MSR $\beta = 0.15$	29.6%	0.2852	0.2994

V. Conclusion

This work presented a method to speed up sliding window-based object detectors by multi-scale spectral residual analysis, named MSR. This way, MSR avoids using a full-fledged object detector on windows unlikely to contain objects, speeding up detection. In our experiments, MSR was able to provide better or similar detection performance, and faster detection with scalability to increasing image resolutions. Furthermore, our

[2]The closest thresholds to 20% and 30% of SWR, respectively. Equivalent to 80% and 70% of window elimination.

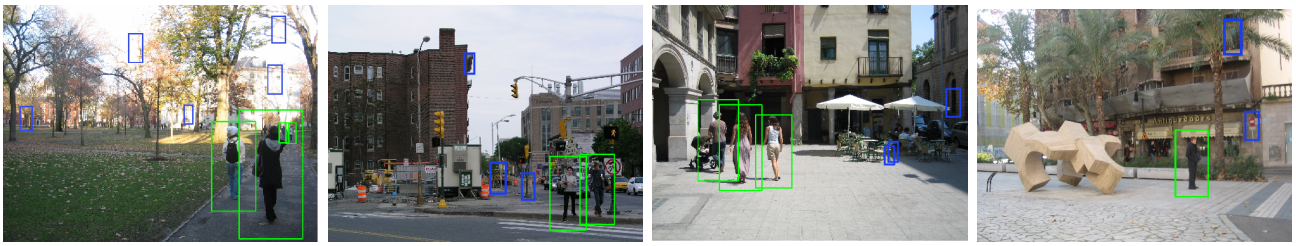

Fig. 11. MSR positive results at 30% of WSR after non-max suppression. Blue rectangles indicate avoided false positives (improving performance); TP are marked with green.

Fig. 12. MSR negative results at 30% of WSR after non-max suppression. Yellow rectangles indicate FN caused by MSR (affecting performance); blue rectangles indicate avoided false positives (improving performance); TP are marked with green, while red rectangles are FP.

choice for spectral residual analysis has demonstrated comparatively better results on the task of faster object detection than other state-of-the-art saliency methods. Although the initial goal was of faster execution, we plan to modify MSR to take object-specific spectral information into account in order to improve even more detection performance.

Acknowledgment

Grimaldo was supported with a scholarship by Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES)

References

[1] M. Enzweiler and D. Gavrila, "Monocular pedestrian detection: Survey and experiments," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 31, no. 12, pp. 2179 –2195, 2009.

[2] V. Kastrinaki, M. Zervakis, and K. Kalaitzakis, "A survey of video processing techniques for traffic applications," *Image and Vision Computing*, vol. 21, no. 4, pp. 359–381, 2003.

[3] W. Hu, T. Tan, L. Wang, and S. Maybank, "A survey on visual surveillance of object motion and behaviors," *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, vol. 34, no. 3, pp. 334–352, 2004.

[4] Q. Zhu, S. Avidan, M. Yeh, and K. Cheng, "Fast Human Detection Using a Cascade of Histograms of Oriented Gradients," *IEEE International Conference on Computer Vision and Pattern Recognition*, pp. 1491–1498, 2006.

[5] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," *IEEE International Conference on Computer Vision and Pattern Recognition*, pp. 511–518, 2001.

[6] X. Hou and L. Zhang, "Thumbnail generation based on global saliency," *Advances in Cognitive Neurodynamics*, pp. 999–1003, 2008.

[7] A. Y. S. Chia, S. Zhuo, R. K. Gupta, Y. W. Tai, S. Y. Cho, P. Tan, and S. Lin, "Semantic colorization with internet images," *ACM Transactions on Graphics*, vol. 30, no. 6, p. 156, 2011.

[8] X. Hou and L. Zhang, "Saliency detection: a spectral residual approach," in *IEEE International Conference on Computer Vision and Pattern Recognition*, 2007, pp. 1–8.

[9] L. Itti, C. Koch, and E. Niebur, "A model of saliency-based visual attention for rapid scene analysis," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 20, pp. 1254–1259, 1998.

[10] C. Y. Ip and A. Varshney, "Saliency-assisted navigation of very large landscape images," *IEEE Transactions on Visualization and Computer Graphics*, vol. 17, no. 12, pp. 1737–1746, 2011.

[11] U. Rutishauser, D. Walther, C. Koch, and P. Perona, "Is bottom-up attention useful for object recognition?" in *IEEE International Conference on Computer Vision and Pattern Recognition*, vol. 2, 2004, pp. 37–44.

[12] D. Lowe, "Object recognition from local scale-invariant features," *IEEE International Conference on Computer Vision*, pp. 1150–1157, 1999.

[13] J. Feng, Y. Wei, L. Tao, C. Zhang, and J. Sun, "Salient object detection by composition," in *IEEE International Conference on Computer Vision*, 2011, pp. 1028–1035.

[14] C. H. Lampert, M. B. Blaschko, and T. Hofmann, "Beyond sliding windows: object localization by efficient subwindow search," *IEEE International Conference on Computer Vision and Pattern Recognition*, pp. 1–8, 2008.

[15] P. F. Felzenszwalb, D. McAllester, and D. Ramanan, "A discriminatively trained, multiscale, deformable part model," in *IEEE International Conference on Computer Vision and Pattern Recognition*, 2008, pp. 1–8.

[16] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *IEEE International Conference on Computer Vision and Pattern Recognition*, vol. 1, 2005, pp. 886–893.

[17] M. M. Cheng, G. X. Zhang, N. J. Mitra, X. Huang, and S. M. Hu, "Global contrast based salient region detection," in *IEEE International Conference on Computer Vision and Pattern Recognition*, 2011, pp. 409–416.

[18] J. Harel, C. Koch, and P. Perona, "Graph-based visual saliency," *Advances in neural information processing systems*, vol. 19, p. 545, 2007.

[19] R. Achanta, S. Hemami, F. Estrada, and S. Süsstrunk, "Frequency-tuned salient region detection," in *IEEE International Conference on Computer Vision and Pattern Recognition*, 2009, pp. 1597–1604.

[20] Z. Zhai and M. Shah, "Visual Attention Detection in Video Sequences Using Spatiotemporal Cues Categories and Subject Descriptors," in *ACM International Conference on Multimedia*, 2006, pp. 815–824.

[21] B. C. Russell, A. Torralba, K. P. Murphy, and W. T. Freeman, "LabelMe: a database and web-based tool for image annotation," *International Journal of Computer Vision*, vol. 77, no. 1, pp. 157–173, 2008.

[22] M. M. Cheng, G. X. Zhang, N. J. Mitra, X. Huang, and S. M. Hu, "Global contrast based salient region detection," in *IEEE International Conference on Computer Vision and Pattern Recognition*, 2011, pp. 409–416.

[23] S. Goferman, L. Zelnik-Manor, and A. Tal, "Context-aware saliency detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp. 2376–2383, 2011.