Traditional Saliency Reloaded: A Good Old Model in New Shape

Simone Frintrop, Thomas Werner, and Germán M. García
Institute of Computer Science III
Rheinische Friedrich-Wilhelms-Universität Bonn, Germany
frintrop@iai.uni-bonn.de

Abstract

In this paper, we show that the seminal, biologically-inspired saliency model by Itti et al. [21] is still competitive with current state-of-the-art methods for salient object segmentation if some important adaptions are made. We show which changes are necessary to achieve high performance, with special emphasis on the scale-space: we introduce a twin pyramid for computing Difference-of-Gaussians, which enables a flexible center-surround ratio. The resulting system, called VOCUS2, is elegant and coherent in structure, fast, and computes saliency at the pixel level. It is not only suitable for images with few objects, but also for complex scenes as captured by mobile devices. Furthermore, we integrate the saliency system into an object proposal generation framework to obtain segment-based saliency maps and boost the results for salient object segmentation. We show that our system achieves state-of-the-art performance on a large collection of benchmark data.

1. Introduction

More than 15 years ago, Laurent Itti and colleagues published one of the first computational attention and saliency systems: the iNVT (iLab Neuromorphic Vision Toolkit) [21]. It was designed to simulate the mechanisms of human visual attention and thus to find regions of interest in images. The approach is considered to be the origin of computational saliency systems and has been cited more than 6000 times. Since then, the interest in the computer vision community in saliency computation has increased strongly, since it is a valuable method for tasks such as thumbnailing, retargeting, or object recognition. Several databases for evaluating systems have been created and many new sophisticated saliency systems are presented every year (see [5, 7] for surveys). Many systems show now impressive results with precise saliency maps: obviously there has been a huge progress in the field. But do these new methods compute saliency in a better, or just in a different way? In other words, did the original Itti model miss essential ideas to compute saliency that newer approaches include?

In this paper, we show that the traditional structure of saliency models based on multi-scale Difference-of-Gaussians is still competitive with current salient object detection methods. We present the new saliency system VOCUS2 that follows in its basic structure the Itti-Koch model [21]: feature channels are computed in parallel, pyramids enable a multi-scale computation, contrasts are computed by Difference-of-Gaussians. Since the original model was designed to simulate eye movements, we need some adaptations to achieve high performance for salient object segmentation. Especially important is the scale-space structure (we use a new twin pyramid) and the center-surround ratio, which has turned out to be the most crucial parameter of saliency systems. Like the approach of Itti, the resulting system has a simple and elegant structure which follows concepts from human perception. Furthermore, it is applicable not only to web images with few prominent objects but also especially well suited for applications on mobile systems since it is fast and does not require center or background priors for good performance.

Our system produces pixel-precise saliency maps, instead of segment-based ones. We believe that this is an important quality of a general saliency method, since pixels actually do have different saliency. Even if they have exactly the same color value, they might be perceived differently, due to their neighborhood: a pixel that differs strongly from its neighbors, e.g., at an object border, is considered more salient than one surrounded by similar pixels. However, for some tasks, segment-based saliency maps are beneficial, especially if precise object boundaries are required. To obtain these, we show how to integrate the presented saliency method into an object proposal generation framework [30]. The resulting object proposals are then combined to form a segment-based saliency map, which gives especially good results on several benchmark datasets for salient object segmentation.

The contribution of this paper is thus not a new concept for saliency, but to show that the traditional, biologically-inspired concept is still valid and obtains competitive results for salient object segmentation if adapted accordingly. We believe that this insight is very valuable for the computer vision community and related fields since it shows what is really essential for computing saliency: a measure of a center-surround contrast on different scales. Since the computation of this contrast with DoG filters is quick and simple, this method is well suited for many applications, especially for those requiring real-time performance.

2. Background and Related Work

Saliency computation has its roots in psychological theories about human attention, for example the feature integration theory (FIT) [34]. This theory states that in the human brain, several features are processed in parallel in different brain areas, and the location of these features are collected in one "master map of locations". From this map, "attention" selects the current region of interest. This map is similar to what is nowadays called "saliency map", and there is strong evidence that such a map exists in the brain, presumably in V1 [41].

The earliest computational attention models have been built based on FIT, that means they compute several feature maps, usually with Difference-of-Gaussian and Gabor filters, and fuse them to a saliency map. In chronological order, systems based on this structure are [25, 31, 21, 12, 35, 16]. These models have been mainly designed to simulate eye movements, and it was recently shown that they still obtain good results for such tasks [8]. However, on benchmarks for salient object detection and segmentation these models usually perform less well and it has been claimed that other methods are required for such tasks.[1]

During the last decade, many saliency systems have been presented that are less biologically inspired and that were mainly designed for salient object segmentation. There are approaches that are based on the spectral analysis of images [17, 33], models that base on information theory [9, 23, 24], Bayesian theory [20, 40], or on decision theory [15, 14], and those that use machine learning techniques to learn a combination of features [27, 3]. Because of the overwhelming number of different approaches, it is hard to keep an overview and to see the differences, and, more importantly, the similarities of the methods. When looking more closely at the methods, it reveals that the underlying method that exists in basically all saliency systems is a contrast computation. Since salient items are salient because of their difference to the surrounding, computing such a contrast is an essential step.

The cognitive models based on FIT compute the center-surround contrast usually by Difference-of-Gaussian (DoG) filters since these model best the ganglion cells in the human retina. Also other approaches such as the Bayesian surprise model [20] or the decision-theoretic model of [15] use DoG filters to compute contrasts. Some methods compute the contrast not based on pixels but on patches [37, 6] or on superpixels [32, 38, 42]. Instead of computing local contrasts, some approaches compute global contrasts by considering the whole image as surrounding region [1, 9].

While it has turned out during the last decade that there are numerous ways to compute saliency (since there are numerous ways to compute contrasts), it is less clear which of these methods is favorable. The usefulness of a system for an application will depend also on other aspects than AUC values: is the system robust enough to deal with data different from the benchmarks? Is the system comprehensible so that it is possible to quickly understand how parameters affect the performance? And last but not least: is the approach fast enough to be applicable for real-time applications?

We believe that the traditional FIT-based structure fulfills these requirements: it is simple and elegant in structure, easy to implement, there are few parameters which are easily adaptable, and the system can be implemented to be very fast. We show in this paper how a FIT-based approach can be implemented to achieve state-of-the-art performance for salient object segmentation and that the resulting system is thus suitable for many applications.

3. Saliency Computation with VOCUS2

In this section, we introduce our new implementation[2] of the traditional FIT-based saliency approach by Itti et al. [21] and outline in detail which adaptions have been made to obtain state-of-the-art performance for salient object segmentation. Our saliency system is called VOCUS2, indicating that it is a successor of our previous VOCUS system [12]. VOCUS had a bottom-up and a top-down part, and the bottom-up part was also based on FIT. We are currently working on a similar top-down part for VOCUS2.

Sec. 3.1 describes the basic saliency system, producing pixel-precise saliency maps. In Sec. 3.2, we combine the method with an optional location prior and in Sec. 3.3 with a proposal detection method to obtain segment-based maps.

3.1. Basic Saliency System

Fig. 1 shows an overview of our new saliency system VOCUS2. In a nutshell, the system works as follows. The input image is converted into an opponent-color space with channels for intensity, red-green, and blue-yellow colors. For each channel, we compute two image pyramids (one center and one surround pyramid), on which center-

[1] Discussion at the CVPR 2013 tutorial "A Crash Course on Visual Saliency Modeling: Behavioral Findings and Computational Models"

[2] Code: http://www.iai.uni-bonn.de/∼frintrop/vocus2.html

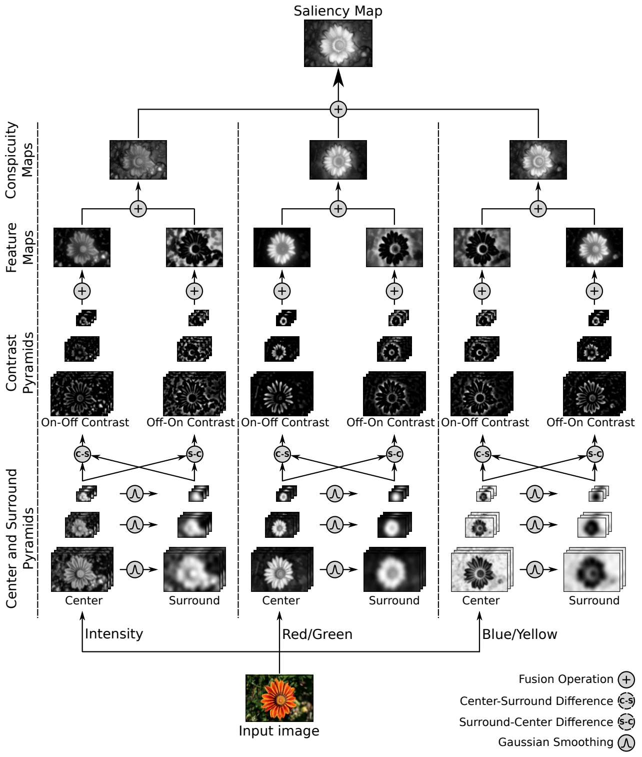

Figure 1. Overview of our saliency system VOCUS2

surround contrast is computed. Finally, the scales and feature channels are fused to one saliency map. This overall structure is in correspondence to FIT and similar to other FIT-based systems such as iNVT [21], VOCUS [12], or STB [35]. The details of the methods and the differences to the iNVT will be outlined in the following. Table 1 summarizes the main differences.

3.1.1 Feature Channels

As in most other saliency systems, we base our computations mainly on intensity and color features. This is in correspondence to human perception, since color is one of the basic features that guide visual attention [36]. We implemented also an orientation channel based on Gabor filters, but it turned out that it is less useful for salient object segmentation. The reason is that it assigns high saliency values to object edges, which makes the segmentation of objects difficult. Therefore, this feature is not used in the following; it is however likely that it is useful for other applications such as modeling eye fixations.

The color computation is performed in an opponent color space, which corresponds to the opponent theory of human perception [19]. This theory states that there are three opponent channels in the human visual system: red versus green, blue versus yellow, and black versus white. We have experimented with the LAB color space, but obtained better results with the simple color space used in [24]: the intensity channel is obtained by I $= \left(\frac{R+G+B}{3}\right)$, and the two color channels by $RG = R - G$ and $BY = B - \frac{R+G}{2}$. We can treat now all three channels I, RG, and BY equally to determine feature-specific saliencies.

3.1.2 Scale Space with Twin Pyramids

The saliency computation is embedded in a scale-space structure on which multi-scale center-surround contrasts are computed. However, our scale space differs from the iNVT in two ways: first, instead of a simple Gaussian pyramid, we use a more sophisticated scale-space with scales and octaves as in [28]. Second, instead of subtracting layers of the pyramid to obtain Difference-of-Gaussian (DoG) contrasts, we build twin pyramids, consisting of a center pyramid $\boldsymbol{C} = (C_0, ..., C_k)$ and a surround pyramid $\boldsymbol{S} = (S_0, ..., S_k)$ (we use $k = 5$). Here, every center image C_i has a corresponding surround image S_i, which is obtained from C_i by smoothing with a sigma σ_x that corresponds to the desired center-surround ratio ($\sigma_x = \sqrt{\sigma_s^2 - \sigma_c^2}$, where σ_c is the value used to obtain the center image C_i and σ_s is the effective smoothing factor for the surround image S_i). This has the advantage that we are not restricted to center-surround ratios given by the pyramid but can vary this value flexibly. Since the center-surround ratio has turned out to be the most crucial parameter in a saliency system, this ability to fine-tune the system is of special importance and justifies the slightly higher computational effort.

To motivate our modified scale-space, let us briefly describe the original approach in iNVT and its limitations. In iNVT, the center-surround contrast is computed by subtracting images from different layers of the pyramid $L(c, s) = |L_c \ominus L_{c+\delta}|$, where L_i is layer i of one of the pyramids, center $c \in \{2, 3, 4\}$, $\delta \in \{3, 4\}$, and $\ominus$ denoting the across-scale subtraction, obtained by interpolation to the finer scale followed by point-by-point subtraction. This corresponds to a Difference-of-Gaussian (DoG) approach, since it is well known that subtracting two images which have been obtained by smoothing image I with Gaussians G_1 and G_2 respectively, is equivalent to directly applying a Difference-of-Gaussian filter $DoG = G_1 - G_2$ to I, i.e., $I * G_1 - I * G_2 = I * (G_1 - G_2) = I * DoG$, where '*' denotes the convolution operator. There is however one drawback of this method: processing is restricted to the smoothing factors of the Gaussians used to create the pyramid. In a standard Gaussian pyramid as used in [21], the effective smoothing factors with respect to the original image are powers of 2: the k-th map in the pyramid is effectively smoothed with $\sigma = 2^k$. Thus, also the ratios of the center-surround difference can only be powers of two (the iNVT uses ratios of 1:8 and 1:16); other ratios, e.g., 1:3 or 1:10, are not possible in this framework.

While a finer granularity can be obtained by using a

	iNVT	VOCUS2 (Ours)
Features	intensity (I), color (C), orientation (O)	intensity (I), color (C)
Pyramid structure	one pyramid	**twin pyramids** (main difference)
	one scale per layer	multiple scales per layer
Feature fusion	down-sampling	up-sampling
	weighting by uniqueness	arithmetic mean
	fuse color channels first, then intensity	treat all 3 channels equally

Table 1. Main differences between iNVT [21] and our VOCUS2 system.

scale-space with several scales per layer [28], the scale maps used for subtraction still have to be chosen from the available set of the pyramid and arbitrary center-surround ratios cannot be chosen.

In our new structure, we obtain the surround image S_i instead directly from the center pyramid according to a desired center-surround ratio. Thus, we are able to determine this ratio independently, which results in more flexibility to adapt the system. This adaption has turned out to be the most effective change to increase the performance (cf. Sec. 4.1).

We compute center and surround pyramids for each channel, denoted as $\boldsymbol{C}^f$, $\boldsymbol{S}^f$, where f stands for one of the feature channels $\{I, RG, BY\}$.

3.1.3 Center-surround Contrast

The color and intensity contrasts can be computed now easily by subtracting the center and the surround map. To distinguish bright objects on dark background from dark objects on bright background, the contrast computation is separated into on-off and off-on contrasts, corresponding to cells in the human visual system that respond only to one type of contrast respectively.[3] This gives us two contrast maps for each layer i of the pyramids: $X_i^f = C_i^f - S_i^f$ (for on-off contrasts) and $Y_i^f = S_i^f - C_i^f$ (for off-on contrasts), with $f \in \{I, RG, BY\}$. In both cases, values below 0 are set to 0. We call the resulting pyramids *contrast pyramids*.

3.1.4 Feature Fusion

Next, the images from each contrast pyramid are summed up with across-scale addition to obtain the *feature maps*:

$$F_1^f = \bigoplus_i X_i, \quad \text{with } i \in \{1, .., k\} \tag{1}$$

and

$$F_2^f = \bigoplus_i Y_i, \quad \text{with } i \in \{1, .., k\}, \tag{2}$$

with $f \in \{I, RG, BY\}$. In contrast to iNVT, our across-scale addition $\oplus$ interpolates to the finest scale, not the coarsest one, before adding the maps point-wise.

The two feature maps of each channel are then fused to conspicuity maps,

$$\mathcal{C}^f = f(F_1^f, F_2^f), \text{ with } f \in \{I, RG, BY\}, \tag{3}$$

and these are finally fused to a single saliency map S:

$$S = g(C^I, C^{RG}, C^{BY}), \tag{4}$$

where g is a fusion operation. Different options are possible for this fusion, e.g., the arithmetic mean, a maximum operation, or a non-linear weighting function that favors maps with few peaks as in [21, 12]. The purpose of such a weighting is to strengthen outliers which appear only in one feature channel. This is especially important when simulating eye movements and prioritizing data processing, but less so when performing salient object segmentation. Correspondingly, we obtained the best results by the arithmetic mean that treats all maps equally.

Note that this equal treatment of all three color channels differs from the iNVT, which instead fuses first all color feature maps to one conspicuity map before color is fused with intensity. The latter approach overemphasizes intensity, since the contribution in the saliency map originates fifty-fifty from intensity respectively color. Instead, in our approach, each channel contributes with one third to the saliency map, which has increased the performance considerably (cf. Sec. 4.1).

3.2. Location Prior (optional)

We believe that a general saliency system should not rely on a location prior that prioritizes, for example, the image center, in order to be able to compute saliency for arbitrary input. For applications on mobile systems like head-mounted devices or robots, objects can have any position and size and objects often intersect with the image borders. Thus, a location or size prior does more harm than good. However, for applications like analyzing photo collections, a location prior can be beneficial and, since also most current benchmarks are heavily biased, it helps to obtain good results on these datasets. We include therefore an optional location prior that can be applied to the saliency map by

[3] Note that it depends on the fusion operation whether this separation has an effect. For the arithmetic mean, the separation results in the same maps as without separation. But when applying a non-linear weighting function as it is useful for eye movement simulation, the separation is important. To keep the model flexible for different fusion methods, we recommend the separation of channels.

multiplying the saliency values with a Gaussian centered at the image center:

$$\hat{s}(x,y) = s(x,y) * \exp\{-\frac{||(x,y)-(x_c,y_c)||^2}{2\sigma^2}\}, \qquad (5)$$

where $s(x,y)$ is the saliency map, (x_c, y_c) the image center, and σ determines the width of the Gaussian. σ has to be chosen rather large to enable a strengthening of the image center, but to still consider a reasonably large part of the image (here we use $\sigma = 79$). Fig. 4 shows the large effect this location prior has on current benchmarks, indicating that these datasets are strongly biased towards the center.

3.3. Segment-based Saliency (optional)

While we believe that a general saliency system should be able to compute pixel-precise saliency, it is for some applications necessary to also segment the object(s) from the background. Additionally, saliency maps with precise object boundaries usually obtain better quantitative results, since evaluation measures usually evaluate how precisely the objects are segmented rather than how many objects are detected. However, it is not necessary to use different saliency methods to obtain segment-based saliency maps, instead we propose a simple way to obtain a segment-based saliency map from the pixel-precise one.

The approach we use here is based on a generic object proposal detection method that generates and ranks object candidates. The resulting candidates are finally overlaid to obtain a segment-based saliency map. The proposal method we chose is similar to [30]. We segment an input image using Mean Shift [11], and in parallel, compute a saliency map with VOCUS2. Then, local maxima are found in the saliency map, and region growing is done seeded at the maxima and limited by different thresholds (we use 33% and 40% of the local maximum). As a result, a set of salient blobs is obtained that roughly estimates the location and extent of the objects. Segmentation and saliency are combined by selecting for each salient blob the segments which overlap more than k percent with the blob (we use $k = 30\%$). This set of segments per blob constitutes one object proposal; the number of proposals per frame is thus the same as the number of salient blobs, but the proposals have more precise boundaries than the blobs. The proposals are then ranked by average saliency and non-maxima suppression removes duplicates; finally, we threshold to maintain only proposals with at least p percent average saliency of the best proposal (here: $p = 33\%$). The final segment-based saliency map is obtained by taking for each pixel the maximum value of all proposals covering this pixel, which corresponds to the highest average saliency obtained for this pixel.

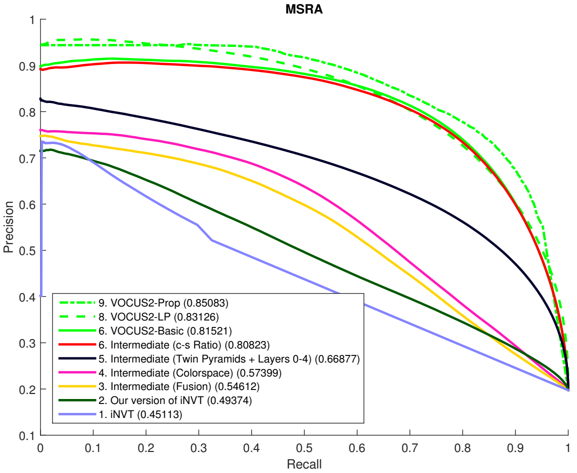

Figure 3. Stepwise improvements of Itti's iNVT saliency system [21], until finally obtaining our VOCUS2 system. Evaluation on MSRA-1000 dataset, AUC values in parentheses.

4. Evaluation

Our evaluation consists of three parts: first, we evaluate the changes that we made with respect to Itti's iNVT by stepwise adaptions of the method. Second, we compare the performance of our saliency system with several other approaches on a large collection of benchmark datasets. And finally, we show that our method is also especially suited for applications on mobile devices. More results, e.g. an evaluation of the most important parameters of the VOCUS2 system, can be found in the supplementary material.

4.1. Step-by-Step Comparison with iNVT

In this section, we show step by step which adaptations of the original Itti system have which effects for the performance for saliency object segmentation. The steps that we compute are the following, where each step includes the adaptations of the previous step:

1. Original Itti-Koch system iNVT [21], (Code from http://ilab.usc.edu/toolkit)
2. Our re-implementation of iNVT (main differences: channels for intensity and color, no orientation; fusion with arithmetic mean (see Sec. 3.1 for details))
3. Equal treatment of all three channels I, RG, BY, instead of first fusing color and then intensity (see Sec. 3.1.4)
4. Adaptation of the color space (see Sec. 3.1.1)
5. Using twin pyramids instead of one pyramid (see Sec. 3.1.2)
6. Adapting the center-surround ratio to 3:13 (see Sec. 3.1.2)
7. Using 2 scales per layer instead of 1 (see Sec. 3.1.2) (version corresponds to VOCUS2-Basic saliency map, eq. 4)
8. added location prior (version corresponds to VOCUS2-LP saliency map, eq. 5)
9. segment-based proposal map according to Sec. 3.3 (version corresponds to VOCUS2-Prop saliency map)

Fig. 3 shows the results of these steps on the MSRA-1000 dataset [1]. It can be seen that the largest performance boosts are obtained by the introduction of the twin pyramid,

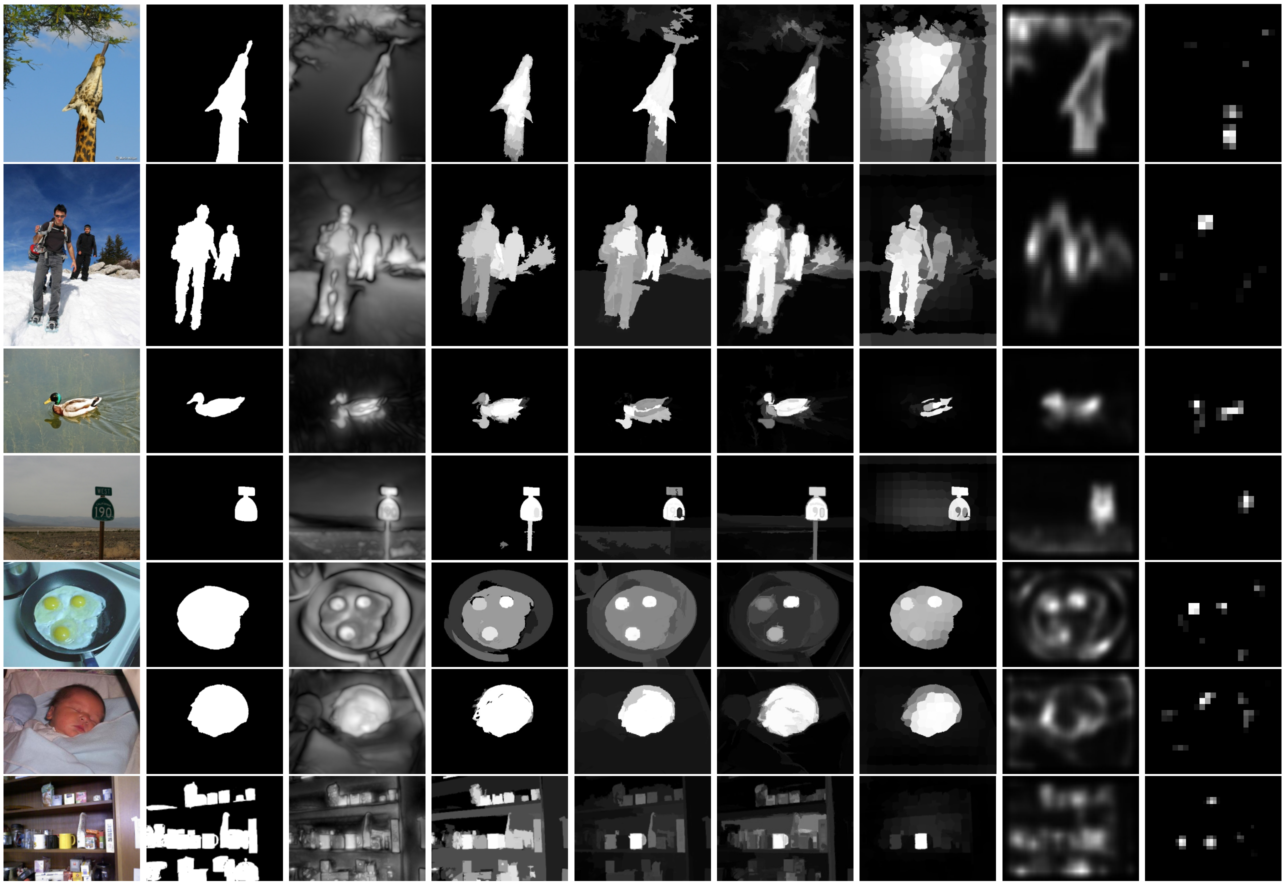

Figure 2. Examples saliency maps. From left to right: original image, ground truth, saliency map for our VOCUS2-Basic, our VOCUS2-Proposals, HSaliency, DRFI, Yang13, HZ08, iNVT

and the corresponding adaptation of the center-surround ratio, which is only possible with twin pyramids. But also minor changes as changing the color space or the order of fusing color and intensity have surprisingly large effects.

4.2. Comparison with Other Saliency Methods

We have compared our approach with several other saliency methods with two evaluation measures on a large collection of benchmark images from several datasets. The datasets we use are MSRA-10k [10], SED1 and SED2 [4], ECSSD [38], and PASCAL-S [26]. All datasets together consist of 12050 images. We did code optimization however only on the MSRA-1000 dataset, as described in the previous section.

We compare our method with the following saliency methods: Itti's iNVT [21], the SaliencyToolbox (STB) [35], HZ08 [18], AIM [9], AC09 [1], AC10 [2], CoDi [24], HSaliency [38], Yang 2013 [39], and DRFI [22]. They have been chosen due to their popularity and frequency of citations [21, 35, 9, 18] or due to their recency and very good results on similar tasks [2, 24, 38, 39, 22]. Code of all methods was obtained from the authors' webpages. Fig. 2 shows several examples saliency maps.

We evaluated the systems with two measures: first, with the popular precision-recall method from [1]: saliency maps are thresholded with an increasing $k \in [0, 255]$. This results in binarized maps which are matched against the ground truth to obtain precision and recall values. This measure is the most common comparison measure for salient object segmentation. However, it was recently shown that this measure has several limitations and ranks saliency maps differently than humans would [29]. The authors proposed instead a new measure, called the weighted f-measure F_β^w, that overcomes these limitations. We use F_β^w as our second evaluation measure.

Fig. 4 (left) shows F_β^w, averaged over all datasets. It shows that our segment-based VOCUS2 saliency method (V2-Prop) outperforms all other methods, and the pixel-precise VOCUS2 method (V2-LP) outperforms all methods that compute pixel-precise saliency maps (right part of figure). On these datasets, we use the location prior, since most of the data has a strong center bias. On the right of Fig. 4 we show the precision-recall plots averaged over all datasets. While the measure is flawed according to [29], it is still the most common evaluation method and we include the evaluation for completeness. Here, our segment-based

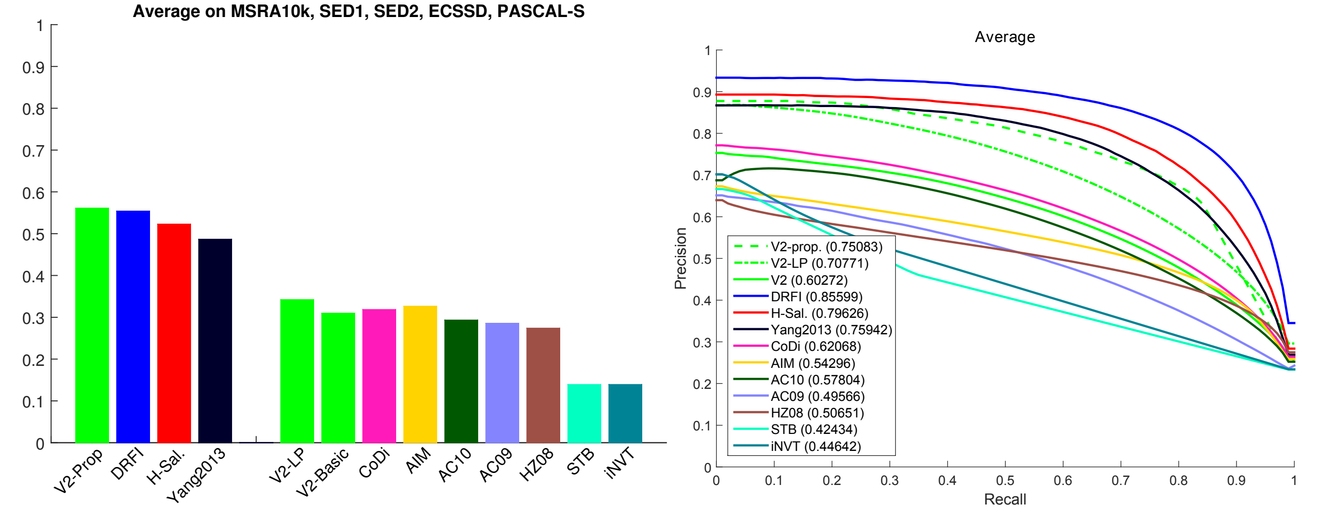

Figure 4. Results averaged over MSRA-10k, SED1, SED2, ECSSD, and PASCAL-S datasets. Left: Weighted F-measure [29]. Left part of plot (4 methods): segment-based methods, right part (9 methods): pixel-precise methods. Right: precision-recall curves. Our approach is denoted as V2-Basic (pixel-precise), V2-LP (with location prior), and V2-Prop (segment-based).

method is ranked third, together with the method of Yang et al. [39]. Our pixel-precise method VOCUS2-LP clearly outperforms all other pixel-precise methods. Individual plots for each dataset can be found in the supplementary material accompanying this paper.

In summary, the results show that our approach clearly achieves state-of-the-art performance: with the weighted f-measure, it outperforms all other methods, according to the precision-recall measure it is on rank three of the segment-based methods and outperforms all pixel-precise methods. Additionally, we show in the next section, that the method is perfectly suited for more complex real-world data as obtained from mobile devices.

4.3. Saliency on Mobile Devices

Here, we show that our method is not only applicable to photos and web images, but also to data from mobile devices such as robots or wearable cameras. This is not self-evident, since many methods rely heavily on center or background bias as well as size priors to obtain good performance. Our VOCUS2 method instead is able to perform well without such priors and produces precise saliency maps also for complex scenes. These maps are well suited for subsequent object detection on mobile systems, as we show in [30].

For evaluation in such settings, we use the Coffee Machine Sequence (CMS) [13][4]. It is a video sequence of an office scene with 600 frames, and contains up to 43 objects per frame. Object ground truth is available for every 30-th frame.

The results on the CMS dataset are shown in Fig. 5 (F_β^w measure on the left, and precision-recall curves on the right). Both measures show that our methods clearly outperform all the other methods. Fig. 2 shows in the last row an example image and saliency maps of this sequence.

For mobile devices, it is also important that a system is real-time capable. Our system is able to compute saliency maps in about 10ms for an input image of 300 × 400 pixels on a standard machine (8 × i7-3632QM, 2.2GHz), and it could be further speeded up by code optimizations and parallelization.

5. Conclusion

Plenty of saliency systems have been introduced during the last decade and it is amazing and interesting in how many different ways it is possible to compute saliency. However, when looking closely at the methods, it turns out that the basic essence of all of them is a contrast measure that determines the difference of a pixel, segment, or patch to its local or global surround. We believe that the old approach by Itti and colleagues that based on the psychological feature integration theory is still valid, and that it can be used equally well as other methods to find salient objects in images. We have shown in this paper how this model has to be adapted to obtain state-of-the-art performance and believe that such a system has due to its clear and elegant structure and due to its real-time capabilities a high benefit for many applications, especially on mobile systems.

Acknowledgment This research has been funded by the German Research Foundation (DFG) within the project ”Situated Vision to Perceive Object Shape and Affordances” (FR 2598/5-1) under the D-A-CH lead agency program. We also thank Johannes Teutrine for his extensive help with the evaluations.

[4] CMS dataset: http://www.iai.uni-bonn.de/∼martin/datasets.html

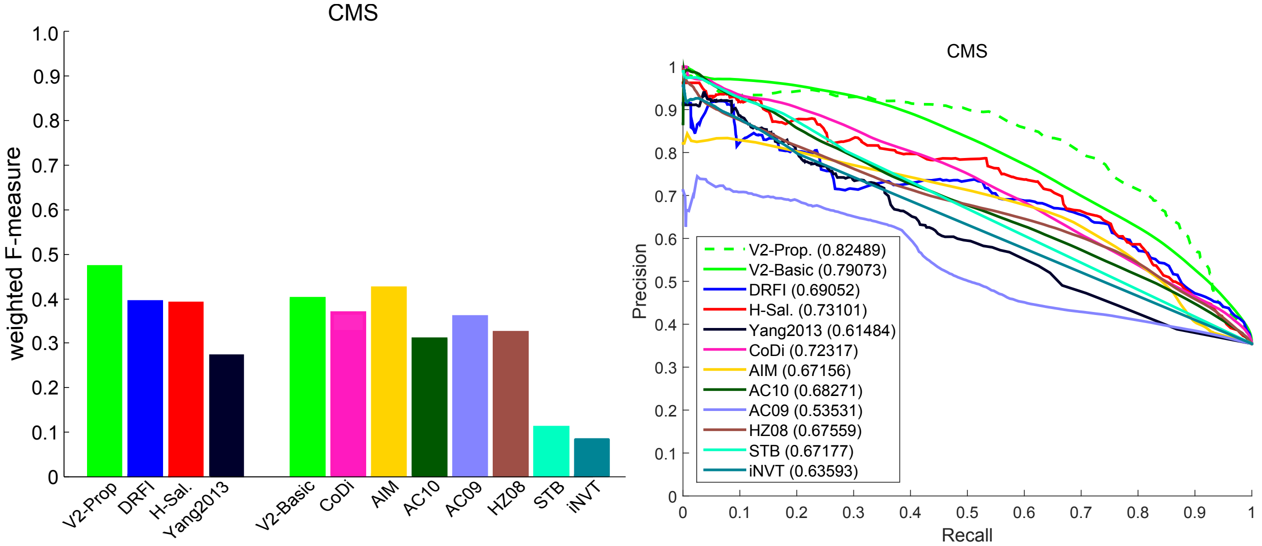

Figure 5. Results on the CMS dataset. Left: weighted F-measure [29]. Right: precision-recall curves. AUC values in parentheses. Our approach is denoted as V2-Basic (pixel-precise), V2-LP (with location prior), and V2-Prop (segment-based).

References

[1] R. Achanta, S. Hemami, F. Estrada, and S. Süsstrunk. Frequency-tuned salient region detection. In *CVPR*, 2009.

[2] R. Achanta and S. Süsstrunk. Saliency Detection using Maximum Symmetric Surround. In *ICIP*, 2010.

[3] B. Alexe, T. Deselaers, and V. Ferrari. What is an object? In *CVPR*, 2010.

[4] S. Alpert, M. Galun, R. Basri, and A. Brandt. Image segmentation by probabilistic bottom-up aggregation and cue integration. In *CVPR*, 2007.

[5] A. Borji and L. Itti. State-of-the-art in visual attention modeling. *TPAMI*, 2010.

[6] A. Borji and L. Itti. Exploiting local and global patch rarities for saliency detection. In *CVPR*, 2012.

[7] A. Borji and L. Itti. Salient object detection: A benchmark. In *ECCV*, 2012.

[8] A. Borji, D. N. Sihite, and L. Itti. Quantitative analysis of human-model agreement in visual saliency modeling: A comparative study. *IEEE Transactions on Image Processing*, 2013.

[9] N. D. B. Bruce and J. K. Tsotsos. Saliency, attention, and visual search: An information theoretic approach. *J. of Vision*, 9(3), 2009.

[10] M.-M. Cheng, N. J. Mitra, X. Huang, P. H. S. Torr, and S.-M. Hu. Global contrast based salient region detection. *TPAMI*, 37(3):569–582, 2015.

[11] D. Comaniciu and P. Meer. Mean shift: a robust approach toward feature space analysis. *TPAMI*, 24(5), 2002.

[12] S. Frintrop. *VOCUS: A Visual Attention System for Object Detection and Goal-directed Search*, volume 3899 of *LNAI*. Springer, 2006.

[13] S. Frintrop, G. M. García, and A. B. Cremers. A cognitive approach for object discovery. In *ICPR*, 2014.

[14] D. Gao, S. Han, and N. Vasconcelos. Discriminant saliency, the detection of suspicious coincidences, and applications to visual recognition. *TPAMI*, 31(6), 2009.

[15] D. Gao and N. Vasconcelos. Bottom-up saliency is a discriminant process. In *ICCV*, 2007.

[16] A. Garcia-Diaz, X. R. Fdez-Vidal, X. M. Pardo, and R. Dosil. Saliency from hierarchical adaptation through decorrelation and variance normalization. *Image and Vision Computing*, 30(1), 2012.

[17] X. Hou, J. Harel, and C. Koch. Image signature: Highlighting sparse salient regions. *TPAMI*, 2012.

[18] X. Hou and L. Zhang. Dynamic visual attention: Searching for coding length increments. In *Advances in Neural Information Processing Systems*, 2008.

[19] L. Hurvich and D. Jameson. An opponent-process theory of color vision. *Psychological review*, 64(6), 1957.

[20] L. Itti and P. Baldi. Bayesian surprise attracts human attention. *Vision Research*, 49(10), 2009.

[21] L. Itti, C. Koch, and E. Niebur. A model of saliency-based visual attention for rapid scene analysis. *TPAMI*, 20(11), 1998.

[22] H. Jiang, J. Wang, Z. Yuan, Y. Wu, N. Zheng, and S. Li. Salient object detection: A discriminative regional feature integration approach. In *CVPR*, 2013.

[23] D. A. Klein and S. Frintrop. Center-surround divergence of feature statistics for salient object detection. In *ICCV*, 2011.

[24] D. A. Klein and S. Frintrop. Salient Pattern Detection using W_2 on Multivariate Normal Distributions. In *DAGM-OAGM*, 2012.

[25] C. Koch and S. Ullman. Shifts in selective visual attention: towards the underlying neural circuitry. *Human Neurobiology*, 4(4):219–227, 1985.

[26] Y. Li, X. Hou, C. Koch, J. M. Rehg, and A. L. Yuille. The secrets of salient object segmentation. In *CVPR*, 2014.

[27] T. Liu, Z. Yuan, J. Sun, J. Wang, N. Zheng, X. Tang, and H.-Y. Shum. Learning to detect a salient object. *TPAMI*, 2009.

[28] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *Int'l J. of Computer Vision (IJCV)*, 60(2):91–110, 2004.

[29] R. Margolin, L. Zelnik-Manor, and A. Tal. How to evaluate foreground maps? 2014.

[30] G. Martín-García, E. Potapova, T. Werner, M. Zillich, M. Vincze, and S. Frintrop. Saliency-based object discovery on RGB-D data with a late-fusion approach. In *ICRA*, 2015.

[31] R. Milanese, H. Wechsler, S. Gil, J. Bost, and T. Pun. Integration of bottom-up and top-down cues for visual attention using non-linear relaxation. In *CVPR*, 1994.

[32] F. Perazzi, P. Krahenbuhl, Y. Pritch, and A. Hornung. Saliency filters: Contrast based filtering for salient region detection. In *CVPR*, Jun. 2012.

[33] B. Schauerte and R. Stiefelhagen. Quaternion-based spectral saliency detection for eye fixation prediction. In *ECCV*, 2012.

[34] A. M. Treisman and G. Gelade. A feature integration theory of attention. *Cog. Psych.*, 12, 1980.

[35] D. Walther and C. Koch. Modeling attention to salient proto-objects. *Neural Networks*, 2006.

[36] J. M. Wolfe and T. S. Horowitz. What attributes guide the deployment of visual attention and how do they do it? *Nature Reviews Neuroscience*, 5:1–7, 2004.

[37] R. J. X. Sun, H. Yao. What are we looking for: Towards statistical modeling of saccadic eye movements and visual saliency. In *CVPR*, 2012.

[38] Q. Yan, L. Xu, J. Shi, and J. Jia. Hierarchical Saliency Detection. In *CVPR*, 2013.

[39] C. Yang, L. Zhang, H. Lu, X. Ruan, and M.-H. Yang. Saliency Detection via Graph-based Manifold Ranking. In *CVPR*, 2013.

[40] L. Zhang, M. H. Tong, T. K. Marks, H. Shan, and G. W. Cottrell. Sun: A bayesian framework for saliency using natural statistics. *J. of Vision*, 8(32), 2008.

[41] X. Zhang, L. Zhaoping, T. Zhou, and F. Fang. Neural activities in V1 create a bottom-up saliency map. *Neuron*, 73:183–192, 2012.

[42] L. Zhu, Z. Cao, D. A. Klein, S. Frintrop, and A. B. Cremers. A multi-size superpixel approach for salient object detection based on multivariate normal distribution estimation. *TIP*, 23(12), 2014.