Application of Conditional Random Fields to Pixel-Level Salient Object Detection Through Local, Regional and Global Features

Jimmy Lin
Australian National University
Canberra, Australia
linxin@gmail.com

Christopher Claoué-Long
Australian National University
Canberra, Australia
u5183532@anu.edu.au

Abstract

Making use of OpenCV, DARWIN and the MSRA dataset, we detect the saliency of an object in an image by extracting local, regional and global features at the pixel level, then combine those features with pre-fitted weights derived by logistic regression. A conditional random field is then constructed to capture the spatial continuity of the salient object, with a weighting ratio of combined unary and pairwise terms that is determined by cross-validation. The binary mask inferred by the conditional random field is then used to output a single bounding rectangle around a salient area through a winner-takes-all algorithm to label the detected salient object, by which the performance of our approach is evaluated through mean boundary displacement error.

1. Introduction and Motivation

A long-standing problem in the field of computer vision is how to calculate the saliency of objects in an image, that is, their prominence in the image when compared to their background and surrounds. Human vision can accomplish this quite easily, since the human brain has mechanisms to direct attention to the main objects in the perceived image. This ability has been studied by researchers in multiple fields, from physiology and psychology, to computer science. The property of detecting a salient object is important for the latter because it leads to the ability to single out individual areas of an image, for example in automatic image cropping and vision simulation in robotics, as well as 3D surface reconstruction in augmented reality displays. Salient object detection is still a field of active research, with multiple avenues being currently researched.

2. Related Works

Most existing approaches are based on a bottom-up computational framework, where the computer's simulation of

Figure 1. Example images from MSRA dataset B

visual attention is driven by low-level stimuli in the scene [4][5][8]. These approaches employ three steps, the first of which is feature extraction. Multiple low-level visual features such as intensity, colour, orientation, texture, and motion are extracted from the image at multiple scales and are then incorporated into a framework to aid in the detection of the salient object.

The second step is saliency computation. The saliency is computed by a centre-surround operation, self-information, or graph-based random walk using multiple features. After normalisation and linear/nonlinear combination, a master or saliency map is computed to represent the saliency of each pixel in the image.

Finally, a few key locations on the saliency map are identified by winner-take-all, inhibition-of-return, or other nonlinear operations and output as a label designating the salient area in the image.

3. Our Approach

In our approach, we use the open source DARWIN machine learning framework [3] as the basis for our code. We employ the MSRA dataset as the base images for our salient

object detection, and build an algorithm using several suppositions that relate to how human vision differentiates salient objects from the rest of the perceived field of view. People naturally pay more attention to salient objects in images, such as a person, a face, a car, an animal, or a road sign (Figure 1). This attention can be mimicked in computer vision through several factors, which were brought forward by Liu et. al in their research [6][7].

First of all, it is likely that pixels with a high contrast difference to their near neighbours to be part of a salient object, since they could represent a contour or boundary around an object. This is similar to how the human brain discovers boundaries between objects in its visual field by detecting differences in light intensity.

Salient objects are more often than not quite distinct from their local surrounding region. Calculating the distance between an object and its surrounds can therefore give us information about how salient that object is. If we apply this concept over the entire image, we can gather information about how different various areas are to their surrounds and work out the likelihood that parts of an image are salient.

Finally, humans perceive object prominence through how distinctly coloured they are compared to the rest of the visual field. Salient objects often demonstrate a marked difference in colour to the rest of a scene. Therefore, the more widely distributed a colour is in an image, the less likely it is that the salient object will contain that colour. This global colour distribution can therefore be used to describe the saliency of the pixels in an image.

3.1. Formulation

To turn the problem of salient object detection into a mathematical formulation, we incorporate these three high-level concepts of a salient object into the process of creating a saliency map. Salient object detection can be formulated this way as a binary labelling task that separates a salient object from the background through multiple operations.

For each pixel x of given an image I, the binary mask A_x must indicate whether it belongs to the salient object (1) or not (0). Our objective is to compute this mask A in order to show the location of the salient object in the image.

To do this, we build up a probabilistic model

$$P(A|I) = \frac{1}{Z} e^{-E(A|I)}$$

to determine the probability of a binary mask over an image with $\frac{1}{Z}$ as the normalising factor, and $E(A|I)$ as an energy function incorporating both unary and pairwise potentials between pixels in the image

Formally, the energy function can be represented as

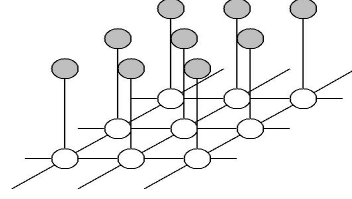$$E(A|I) = \sum_x S_{unary}(a_x, I) + \lambda_0 \sum_{x,x'} S_{pair}(a_x, a_{x'}, I)$$

Figure 2. A Conditional Random Field
White nodes are binary saliencies, grey nodes are composite pixelwise features

where λ is the relative weight between the sum of multiple unary and pairwise features.

The unary potential, combining the three pixel features, is specified as

$$S_{unary}(a_x, I) = \sum_{k=1}^{K} \lambda_k \cdot F_k(a_x, I)$$

where λ_k is the weight of the kth feature.

The pairwise feature $S(a_x, a_{x'}, I)$ exploits the spatial relationship between two adjacent pixels. It can be viewed as a "penalty" for labelling adjacent pixels differently:

$$S(a_x, a_{x'}, I) = |a_x - a_{x'}| \cdot e^{-\beta d_{x,x'}}$$

where x, x' represent two adjacent pixels, $d_{x,x'}$ is the L2-norm (standard norm) representing the colour difference between the two pixels, and $\beta = (2\langle||I_x - I_{x'}||^2\rangle)^{-1}$ is a robust parameter to weight the colour contrast.

The final energy to which each feature $F_k(a_x, I)$ contributes comes from a normalised feature map $f_k(x, I) \in [0, 1]$, where for each pixel:

$$F_k(a_x, I) = \begin{cases} f_k(x, I), & a_x = 0 \\ 1 - f_k(x, I), & a_x = 1 \end{cases}$$

where we penalise the saliency quantity $f_k(x, I)$ of one pixel x when that pixel was predicted not to be within the salient object ($a_x = 0$) and vice versa.

Figure 3. Original Image and Preview of feature maps

Left to Right: Original Image, Multiscale Contrast Map, Centre-Surround Histogram, Colour Spatial Distribution, Composed Unary Potentials

3.2. Feature Extraction

Feature Extraction, widely acknowledged as the most significant component of a computer vision task, represents how we want the computer to interpret raw images. In this project, we focus on three features capable of capturing saliency individually but in different levels of scope. They are respectively multiscale contrast, centre-surround histograms and colour-spatial distribution.

Because of the time expense required to calculate these features on an image, our approach caches the feature maps of each image and uses them for both training and testing.

3.2.1 Multiscale Contrast

Constrast is commonly used as local feature because it simulates the human visual receptive fields. It acts on the fact that the boundary of salient objects tend to have a marked contrast to the surrounding region. Since we may have no prior knowledge about the size of salient object, we compute the contrast at multiple scales to then incorporate back into one map, since this will demarcate the various boundaries in the image. This multiscale contrast map is thus a linear combination of image contrast at all levels of an N-level gaussian image pyramid, using the pixels x in the image I. Formally, this amounts to calculating

$$f_c(x, I) = \sum_{n=1}^{N} \sum_{x' \in W(x)} ||I^n(x) - I^n(x')||^2$$

where W(x) is a window that delineates which area to consider as neighbouring pixels to compare contrast values. The resulting map highlights the edges of different objects in the image, giving high prominence to the contours around the salient object.
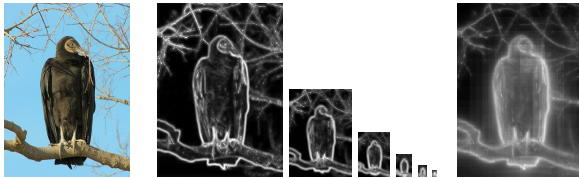
Figure 4. Multiscale Contrast.
Left: Original Image. Right: Multiscale Feature Map.
Middle: multiscale image pyramid from level 1 to 6.
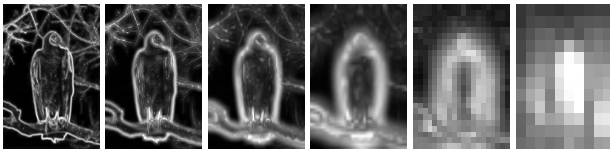

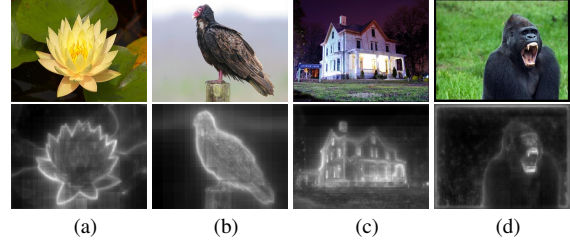Figure 5. Scaled Pyramid Images from level 1 to 6.

Figure 6. Multiscale Contrast Feature Maps

In our implementation, we choose the total number of pyramid level N to be 6 and the size of the window to be 9×9.

As can be seen in Figure 4, the derived multiscale contrast map provides a highly accurate distinction between boundary and non-boundary pixels. This provides us a precise description of where the boundary of a salient object could exist in the output binary mask. When a salient object also has high contrast inside its boundaries, this feature also manages to capture the inside of the salient object, such as the bird in Figure 5(b) and the house in Figure 5(c).

But there are some drawbacks that are tough to avoid. The boundaries of all objects are highlighted, which is not desirable when searching for a single salient object. This is because we need to give a high probability to marking pixels that are relevant to the salient object rather than every boundary possible. If this doesn't happen, there is the possibility of a "saliency leak" into non-salient areas of the final result that could deteriorate the detector's precision.

Another disadvantage to only using multiscale contrast is that the inner regions of salient objects are poorly represented. For example in the gorilla in Figure 5, the teeth are labelled as higher contrast to their surroundings compared to that of the body of the gorilla to the grass. Since each entry of the output feature map is quantitatively normalised, the contrast of the gorilla's body to the grass is rendered trivial compared to the contrast of the tooth. The tooth is not what the human receptive field would label as salient because it is a smaller part of the animal, and thus it should not be labelled as the salient object. This flaw may result in low recall comparing to the ground truth data when contrast is used as the sole method to distinguish a salient object.

3.2.2 Centre-Surround Histogram

Multiscale contrast only partially detects salient objects, since it is only sensitive to the boundaries. In order to detect the object as a whole, we make use of another static salient feature which captures the regional information of an area, computed using various low-level features in a centre-surround difference calculation.

To do this, we create a colour RGB histogram for both

the rectangle and a surrounding frame with the same area as the rectangle, at a certain resolution (number of "bins" in the histogram). We then measure the difference between the area centred at each pixel x and its surrounds by calculating the χ^2 distance between the two histograms representative of those regions. We do this for multiple aspect ratios $\{0.5, 0.75, 1.0, 1.25, 1.5\}$, and keep the largest (most distinct) χ^2 value through the formula

$$R(x) = \underset{R(x)}{\arg\max} \chi^2(R(x), R_s(x))$$
$$= \underset{R(x)}{\arg\max} \frac{1}{2} \cdot \sum_{i \in bins} \frac{(hist_{R(x)_i} - hist_{R_s(x)_i})^2}{hist_{R(x)_i} + hist_{R_s(x)_i}}$$

To reduce the computational complexity involved in this calculation, the size of the rectangle bounding the area in the centre is reduced to 12 discrete ratios $\{0.18, 0.2, 0.25, 0.3, 0.35, 0.4, 0.45, 0.5, 0.55, 0.6, 0.65, 0.7, 0.75\}$ with regards to $min(w, h)$, the minimal value of width and height of the processed image.

The centre-surround histogram feature map at each pixel x is output as

$$f_h(x, I) \propto \sum_{x'|x \in R(x')} w_{xx'} \chi^2(R(x'), R_s(x'))$$

This map reflects how distinct each pixel is from its surrounds, assigned from the largest χ^2 value between the area centred at each pixel of the image and its surround, weighted by its spatial distance from the centre of that area.

As we can see from Figure 6, the salient block within the image is given large emphasis comparing to its surrounds in the output feature map. However, unlike multiscale contrast, the centre-surround histogram does not provide an accurate description of the boundary of the object – an inherent drawback of this regional feature. Used together with the multiscale contrast map, the two features complement the other's weakness to create a stronger sense of the location of salient objects.
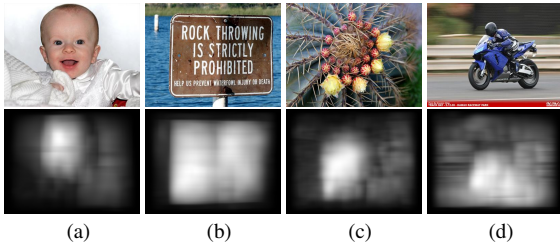

(a)   (b)   (c)   (d)

Figure 7. Centre-Surrond Histogram Feature Maps

(a)   (b)   (c)   (d)
(a) Original (b) No reduction (c) Single Level Reduction
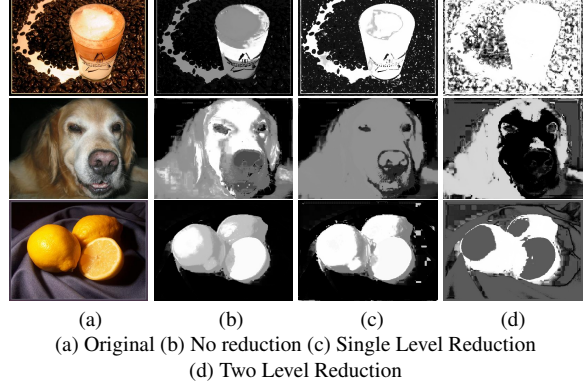(d) Two Level Reduction
Figure 8. Example Colour Spatial Distributions on Reduced Pixels

3.2.3 Colour Spatial Distribution

The goal of using colour spatial distribution is to take into account the global colour information in the image, that is, the information about how widely the colours occur in one image are distributed within it. The computation of this feature is done in several parts.

First of all, we create a Gaussian mixture model with 5 components in order to capture relative colour distribution in the image. This can be represented as $\{c, \mu_c, \Sigma_c\}$ with regards to the colour component c and fit of these components, making use of only a subset of pixels from the whole image. This is done to save on computational time required to process the feature, while at the same time not losing much accuracy as can be seen in Fig.7. We test three levels of pixel reduction via a Gaussian pyramid model: none, single-level reduction, and two-level reduction. Single-level reduction halves the number of pixels used in creating the gaussian mixture model, whilst also providing a smoother distribution without sacrificing much accuracy. The maximum number of iterations over the image is limited to 100 instances, and the convergence criterion is also lowered to 10^{-1} in order to reduce complexity without deteriorating the model. Such simplification is possible only because the exact distribution is not required for capturing the approximate location of a salient object.

Using this model, each pixel is associated to a colour component with the probability

$$P(c|I_x) = \frac{\omega_c \mathcal{N}(I_x|\mu_c, \Sigma_c)}{\sum_c \omega_c \mathcal{N}(I_x|\mu_c, \Sigma_c)}$$

where ω_c is the weight, μ_c is the mean colour, Σ_c is the covariance, and $\mathcal{N}(I_x|\mu_c, \Sigma_c)$ is the multivariate normal distribution of the cth component.

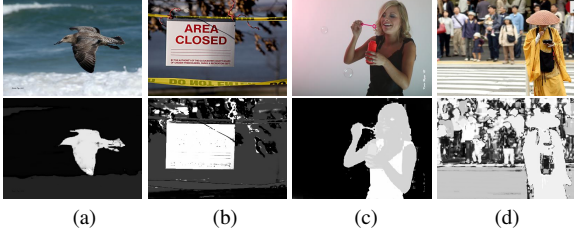For each fitted gaussian colour component c, we com-

Figure 9. Colour Spatial Distribution Feature Maps

pute its horizontal variance $V_h(c)$ as

$$V_h(c) = \frac{1}{|X|_c} \sum_x p(c|I_x) \cdot |x_h - M_h(c)|^2$$

where x_h is horizontal coordinate of pixel x, $|X|_c$ is normalising factor and $M_h(c)$ is the mean of the gaussian component

$$|X|_c = \sum_x p(c|I_x) \quad M_h(c) = \frac{1}{|X|_c} \sum_x p(c|I_x) \cdot x_h$$

Its vertical variance $V_v(c)$ is defined similarly, and we combine the horizontal and vertical variances to derive the unnormalised composite variance $V_h(c)$:

$$V'(c) = V_h(c) + V_v(c)$$

We then employ a min-max approach to normalise the composite covariance over the image

$$V(c) = \frac{V'(c) - min\big(V'(c)\big)}{max\big(V'(c)\big) - min\big(V'(c)\big)}$$

where $V(c)$ is the normalised composite covariance of the cth component, contained between 0 and 1.

Finally, due to the supposition that a salient object tends to have a less widely distributed colour, we set a penalty on the pixels which have a high variance colour. The ultimate colour spatial distribution feature for pixel x is therefore defined as a weighted sum of its colour distribution

$$f_s(x, I) \propto \sum_c p(c|I_x) \cdot (1 - V(c))$$

The feature map $f_s(\cdot, I)$ is normalized to the range $[0, 1]$. The following figure show colour spatial distribution feature maps over several images. The salient objects are labelled effectively by this global feature when there is a high distribution of other colours in the image.

It is evident that the global feature gives a huge amount of accuracy to the detection algorithm when the background is monotonous, as illustrated by the first three images in Figure 8. However, in the case of a varied and colourful background, the colour spatial distribution map fails to distinguish the salient object in the image. Figure 8 (d) demonstrates us this undesired property of the global feature.

3.3. Learning

The three features discussed beforehand all have strengths and weaknesses in different areas, however it goes without saying that adding them together provides an accurate description of the salient area in an image. It would be over-simplistic to treat these features in equal weight since one feature may be better than the others at providing information about saliency. One effective and reasonable approach is to determine the optimal weight to combine three features with the help of a machine learning algorithm. In this project, we implement logistic regression to decide the optimal weight on training data.

As mentioned before, the salient object detection is formulated as binary decision problem. From this perspective, we directly formulate the posterior distribution over the binary saliency variable A_x as sigmoid function.

$$y_x = p(A_x = 1|\boldsymbol{\phi}_x) = \frac{1}{1 + exp(-\boldsymbol{\lambda} \cdot \boldsymbol{\phi}_x)}$$

where the vector $\boldsymbol{\phi}_x$ contains three normalised energy $F_k(x, I)$ for each pixel x, and the parameter vector $\boldsymbol{\lambda} = \{\lambda_1, \lambda_2, \lambda_3\}$ indicates how three extracted features form the unary term in the energy function utilised in the latter CRF inference.

Thus, the likelihood function of this whole image with given human label $\boldsymbol{t}$ is

$$p(\boldsymbol{t}|\boldsymbol{\lambda}) = \prod_x y_x^{t_x} (1 - y_x)^{1-t_x}$$

where the target variable $\boldsymbol{t}$ is the ground truth map, and $\boldsymbol{t}_x$ indicates the human-labelled binary saliency in the pixel x.

We apply the maxmum likelihood estimation to evaluate the parameter $\boldsymbol{\lambda}$, based on the above likelihood function. Under this logistic regression framework, the ultimately learned parameter $\boldsymbol{\lambda}$ is $\{1.58137, 2.62738, 0.563939\}$.

3.4. CRF Inference

The weighted unary potential made up from these weighted feature maps

$$S_{unary}(a_x, I) = \sum_{k=1}^{K} \lambda_k \cdot F_k(a_x, I)$$

is likely to already provide a good estimate of the salient area in the image. However, to capture the spatial continuity of the salient area we infer the maximum likelihood assignment of pixel-wise variables under a conditional random field (CRF) framework. A standard message-passing approach is infeasible when dealing with images at the pixel level, so we instead use α-expansion inference, which successively segments all α and non-α pixels using graph cuts [1]. The algorithm changes the value of α at each iteration using the graph cut algorithm; by this means, inferring

(a)   (b)   (c)   (d)

(a) Raw Image (b) $\lambda_0 = 1$ (c) $\lambda_0 = 10$ (d) $\lambda_0 = 100$

Figure 10. Examples for Binary Mask with various Lambda λ_0

(a) Raw Image (b) Combined Unary Map (c) Binary Mask ($\lambda = 8$)

Figure 11. Examples for CRF inference by α-expansion

the maximum likelihood assignment for each binary variable (or equivalently, the minimal energy function) is much faster.

This requires a way to determine the parameter λ_0 which indicates to what extent, relative to the unary potential of the combined feature map, we ought to consider the pairwise potential in deciding the binary saliency of a pixel. This parameter is key to influencing the smoothness of the resulting binary mask, since as discussed above it can be regarded as a penalty for labelling adjacent pixels with different values.

λ_0	0.1	1	10	50
Precision	0.0	0.0	0.0	0.0
Recall	0.0	0.0	0.0	0.0
F-Measure	0.0	0.0	0.0	0.0
BDE	0.0	0.0	37.78	0.0

To determine a suitable λ_0, we apply cross validation at various magnitudes. Using this, the optimal parameter λ_0 for calculating the final binary mask is $\lambda_0 = 8$.

4. Result Evaluation

To evaluate our approach, we randomly pick 1000 images from the MSRA dataset B to form a training set. We then choose another 500 images to form a testing set.

Figure 12. Examples images for bounding box output

4.1. Bounding Box

The binary mask derived from the CRF inference can be used directly in a multitude of applications. However, in order to evaluate the accuracy of our approach, we make use of OpenCV's findContours algorithm [9] to output a bounding box rectangle around the detected salient object, based on the derived pixel-wise binary mask. The dimensions of this bounding box are written to a text file holding all resultant labels for that image directory.

4.2. Evaluation Criteria

As to this project, we utilise the both approaches to evaluate our result: region-based and boundary-based measurements.

4.2.1 Region-based measurement

Precision and Recall both take significant roles in the information retrieval as a performance indicator. They are used in this project in evaluating the correctness of the output rectangle.

The precision represents the percentage of pixels that are correctly detected in ground truth, which is the ratio of the number of pixels in overlapped region to pixel number contained in the ground truth box. The recall reflects the percentage of pixels that are correctly detected in resulted detection, which is the ratio of pixel number the overlapped region contained to the number of pixels.

It is evident that all three results in Figure 12 are far from satisfying. The salient detection in the leftmost image has high precision but the recall is very low, meaning that it has failed to detect all truly salient pixels. The middle image is no better, with high recall but low precision, with a saliency leak into the surrounding region.

For an overall performance measurement, the F-measure indicator is suitable, with $\alpha = 0.5$ giving the weighted har-
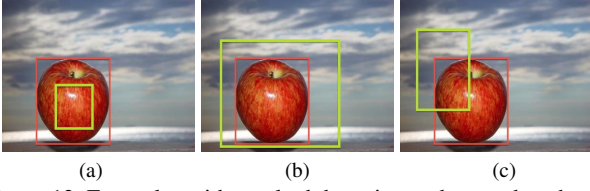
Figure 13. Examples with resulted detection and ground truth
Red Box: ground truth label. Green Box: detected label.

monic mean of precision and recall:

$$F_{0.5} = \frac{1.5 \times Precision \times Recall}{0.5 \times Precision + Recall}$$

4.2.2 Boundary-based measurement

Boundary Displacement Error (BDE) measures the average of positional difference of ground truth and resulted detection. It does this by calculating the average minimum Euclidian distance between the set of pixels in one label B_1 and each pixel x in the other label B_2 [2]

$$BDE(B_1, B_2) = \frac{\sum\limits_{x \in B_1} \min\limits_{y \in B_2} \{d_E(x, y)\}}{|B_1|}$$

A perfect score is 0, where every point in the result output label is contained in the set of points in the truth label. As the results differ more and more from the ground truth, the average BDE value becomes higher.

4.3. Results and Discussion

The average BDE score for our algorithm is **ACTU-ALSCORE**, which is based on training and test sets from the MSRA dataset B. We were not able to calculate the F-measure for performance.

Part of the reason for this somewhat low score is that the contours around the saliency mask map are not always fully closed, leading to high precision with a bounding box rectangle around a single point since it is the largest discovered closed area. Creating a bounding box around all salient points in an image has the opposite effect, with a large recall but multiple non-salient areas of the image contained within the bounding box because of small areas where a "saliency leak" has occurred.

5. Conclusion

In this project, we have demonstrated a supervised approach for salient object detection, formulated as a binary labelling problem using a set of local, regional, and global salient object features. However, there is much room for improvement, especially in the labelling of the salient areas from the derived binary mask. There is also a lack of comparison with other methods using numerical evaluations, which may demonstrate greater effectiveness of the approach in this report.

There are still several remaining issues for further investigation. These include the ability to detect multiple salient areas in an image, improving on previous and current work to form a computational model capable of describing a scene. Another application could be in real-time detection for use in robotics and recording or playback applications. Failure cases can be remedied by creating a more adept algorithm that captures the closest 98% of salient pixels, resulting in much higher accuracy overall. There may also be better ways to derive a label from the raw binary mask of an image to pursue.

References

[1] Y. Boykov, O. Veksler, and R. Zabih. Fast approximate energy minimization via graph cuts. *Pattern Analysis and Machine Intelligence, IEEE Transactions on 23.11*, pages 1222–1239, 2001. 6

[2] J. Freixenet, X. M. noz, D. Raba, J. Martí, and X. Cufí. Yet another survey on image segmentation: Region and boundary information integration. *Proc. European Conf. Computer Vision*, pages 408–422, 2002. 7

[3] S. Gould. Darwin: A framework for machine learning and computer vision research and development. *Journal of Machine Learning Research*, 13:3533–3537, 2012. 1

[4] L. Itti. Models of bottom-up and top-down visual attention. Master's thesis, California Institute of Technology Pasadena, 2000. 1

[5] 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*, 20(11), November 1998. 1

[6] T. Liu, J. Sun, N.-N. Zheng, X. Tang, and H.-Y. Shum. Learning to detect a salient object. *Computer Vision and Pattern Recognition*, CVPR(7):IEEE Conference on. IEEE, 2007. 2

[7] T. Liu, Z. Yuan, J. Sun, J. Wang, N. Zheng, X. Tang, and H.-Y. Shum. Learning to detect a salient object. *Pattern Analysis and Machine Intelligence, IEEE Transactions on 33.2*, pages 353–367, 2011. 2

[8] Y.-F. Ma and H.-J. Zhang. Contrast-based image attention analysis by using fuzzy growing. *Proceedings of the eleventh ACM international conference on Multimedia. ACM*, 2003. 1

[9] S. Suzuki and K. Abe. Topological structural analysis of digitized binary images by border following. *CVGIP*, 30(1):32–46, 1985. 6