Comparison of Spatial Models for Foreground-Background Segmentation in Underwater Videos

Martin Radolko

University of Rostock, Rostock 18051, Germany,
Martin.Radolko@uni-rostock.de

Abstract. The low-level task of foreground-background segregation is an important foundation for many high-level computer vision tasks and has been intensively researched in the past. Nonetheless, unregulated environments usually impose challenging problems, especially the difficult and often neglected underwater environment. There, among others, the edges are blurred, the contrast is impaired and the colors attenuated. Our approach to this problem uses an efficient Background Subtraction algorithm and evaluates it in combination with different spatial models.

Key words: Background Subtraction, Gaussian Switch Model, Markov Random Fields, Belief Propagation, NCut

1 Introduction

Nowadays Computer Vision Systems are used in various fields of applications such as automation, surveillance, human assistance or inspection. Background Subtraction has been used for many years for Computer Vision problems but is still a very valuable source for low level information. It can recognize almost arbitrary objects in any scene, as long as they are in motion. This information can later be reprocessed in different high-level vision tasks.

In order to gather information about the objects of interest in a specific scene, the background of this scene has to be modeled. The task of creating and sustaining an adequate background model is not trivial and associated with many difficulties like changes in the lightning conditions, slightly moving background objects or shadows. A large number of different approaches have been developed to tackle these requirements and create an adequate background model even under harsh conditions. Some use Subspace Learning Models like LDA [1], INMF [2] or PCA [3] to model the background. Other renowned methods adopted techniques like Kalman Filters [4], SVMs [5] or Histograms [6] to background modeling in the hope that they could better cope with these problems.

However, the most promising and common method at the moment is a statistical approach where each background pixel is modeled as a Gaussian Distribution. This approach is justified by the fact that the intensity of a pixel in a completely static scene will vary over time according to a Normal distritution $\mathcal{N}(\mu,\sigma^2)$ due to the inevitable measurement errors inherent in every camera system. A threshold per pixel and channel can be easily derived from the mean and variances of these distributions to distinguish between foreground and background.

There exist some approaches which use just one Normal Distribution per pixel [7], algorithms which use a Mixture of Gaussians [8, 9] or Gaussian-Kernel based methods [10] to model the background. Methods which use a Mixture of Gaussians (MoG) produce in most cases better results than the Single Gaussian (SG) algorithms, because they can model difficult situations (like a swaying tree) better if they are correctly adjusted. Nevertheless, they have a higher memory usage and more parameters which need to be carefully tuned.

A common disadvantage of all Background Subtraction approaches is the missing incorporation of spatial information about the scenes in the model. Natural images are assumed to be very smooth because they usually depict real objects like trees,animals or buildings. This assumption can be used to improve the segmentation which was derived from the Background Subtraction. An example of this is [11], where a simple method is used which erases all connected areas containing less than a certain amount of foreground (or background) pixels. A more sophisticated approach is applied in [12] where a Conditional Random Field models the neighbourhood relations of the pixels. Graph Cuts are used in [13] and in [14] the spatial information is represented in a tensor to whom a Subspace Learning algorithm is applied. The usage of a tensor ensures that all dimensions are treated equally.

We implemented two different spatial models to test both on underwater videos. The first method is based on the popular Normalized Cut (NCut) approach which has been used intensively for single image segmentation [15, 16]. Nonetheless, it has never been applied on videos because of some inherent characteristics which make the NCut unsuitable for videos (see section 2.2) and the high computational costs which forbid any real time application. The first problem can be adressed with a reformulation of the NCut, which adopts it to some video specific requirements. The second problem can be solved by the usage of a simple and fast local optimization algorithm which lowers the computational cost significantly.

The second approach uses Markov Random Fields to represent the spatial relation in the image. This model consists of an undirected graph which is underlaid with a probability map. The probabilites for each pixel are deduced from the Background Subtraction. Nonetheless, the most important model parameter is the neighborhood system, which is a generalized Moore Neighborhood in our case. These large neighborhood systems can model the natural smoothness in images better than the simple 4-connected Neighborhoods normally used.

All of these approaches have been optimized for air images and have not been evaluated on the more difficult underwater images[17, 18, 8, 16]. In the results section we used different self-made underwater videos to compare the two approaches for spatial modeling. Although Markov Random Fields fall behind in accuracy on air images, the same method wins clearly on the difficult underwater images. This result suggest that more special analysis should be made for underwater images and that maybe special algorithms are required for the same task there.

2 Our Approach

In the first part of this section we will explain the Background modeling with the Gaussian Switch Model (GSM) and Background Subtraction with a voting algorithm[19]. The second part describes the N^{2}Cut [20] as a new spatial model for video segmentation and in the last segment a Markov Random Field combined with a Belief Propagation algorithm is introduced as another spatial model. Both of them will be evaluated on underwater images in the results section.

2.1 The Gaussian Switch Model

As justified previously, Gaussian distributions are used to model the colour values of each pixel. The most obvious approach would be a batch method where the n last pictures are saved and then for each pixel and channel the best fitting normal distribution is calculated for the given data. However, this is extremely resource demanding, wherefore we use running gaussians instead. This method just updates the old Gaussians with the values from the newest frame and does not compute the distributions over the n last data points from scratch every time. Thereby, the algorithm gives the new pixel values automatically a higher weight than old values and thus even improves the results in comparison to the batch method because the newer samples usually carry more information about the current background. To be exact: for every Gaussian, the mean μ and variance σ^2 have to be computed. The mean is initiated with the pixel value taken from the first frame of the video stream and the variance is set to a predefined value. Afterwards, they are updated in the following way

$$\mu^{t+1} = \alpha\,\mu^t + (1-\alpha)\,v^t, \tag{1}$$

$$(\sigma^{t+1})^2 = \alpha\,\sigma^t + (1-\alpha)(\mu^t - v^t)^2. \tag{2}$$

The variable α is the update rate and v^t is the pixel value taken from the t-th frame. With these formulas, the Gaussian distribution of a background pixel can be approximated very efficiently.

Nevertheless, one problem is that the model becomes erroneously when a foreground object is visible because it starts to model the foreground object and not the background. To cope with this problem another Gaussian is introduced, the Background Gaussian $\mathcal{N}(\mu^{bg},(\sigma^{bg})^2)$, which is updated after the segmentation process and only if the corresponding pixel is classified as background. This results in a more stable background model as the corruptions from foreground objects are minimized. Nonetheless, there is an inherent problem with this because the model now only accepts values which agree with the current model and acts like a self fullfilling prophecy. One issue are foreground objects already visible in the first frame. At the beginning the model will assume them as background and afterwards will never include the real background into the model because the real background will be classified as foreground. Foreground objects that become background, e.g. a car that parks, will also never get included into the background model for the same reasons. To eliminate these errors a second Gaussian, the Overall Gaussian $\mathcal{N}(\mu^{og},(\sigma^{og})^2)$, is introduced, which will be updated with every new frame.

If a foreground object was visible but immoble for a long period of time, it should be included into the background. Such events result in an Overall Gaussian with a small variance and a mean which is different from the Background Gaussian mean. If such an incident is detected, the Background Gaussian is set to the values of the Overall Gaussian, so that the Object gets included into the background model. This model is applied to every pixel and every channel seperately and later a voting algorithm is used to unify the results and get a definite label for each pixel.

To make the best use of the color information, a special color space is used, which normalises the different intensities in respect to the illumination [14]. Let R, G and B be the given values for a single pixel in the standard RGB color space, then these will be transformed into the three new image channels

$$\begin{aligned} I &= R+B+G, \\ \tilde{R} &= R/I, \\ \tilde{B} &= B/I. \end{aligned}$$

Afterwards the intensity I is scaled to the range $[0,1]$. The color information stored in $\tilde{R}$ and $\tilde{B}$ are normalised with the intensity and will thus not be altered by small or medium changes in the lightning conditions. This can be used to prevent the detection of shadows as foreground. However, if the shadow is very strong the color information may be completely lost in the image and this approach will fail.

At the end a thresholding is applied at each channel seperately. The statistical approach allows to get an adaptive threshold for each pixel and channel. If the variance in the statistical background model is low (high) the noise level at the corresponding pixel and videoframe can also be expected to be low (high). Hence, the variance can be used as an threshold. If p_R is the new value for the red-channel of a pixel, the thresholding inequality is given by:

$$(p_R - \mu_R^{bg})^2 < \max(\beta \cdot (\sigma_R^{bg})^2, 0.001). \tag{3}$$

The maximum is used because the variance could approach near zero values, especially since only matching values are included into the Background Gaussian. The parameter β can control the range of values which are still classified as “matching the model”.

To derive a decision for a pixel as a whole a voting procedure is chosen. If equation (3) is satisfied for at least two of the three channels, the pixel is marked as background, otherwise as foreground. Thereby, the color information can overrule the brightness information and hence shadows should not be detected as foreground. At the end of this process a pixelwise foreground-background segregation is derived only from the temporal information of the video.

2.2 N^{2}Cut

To incorporate spatial information into this segmentation we evaluated two different methods. The first is based on the NCut. In this approach the image is transformed into a graph with a von Neumann Neighborhood to evaluate the best cut. To create an adequate spatial model the weights of the edges in this graph have to be chosen very carefully. We defined the weight of the edge between the nodes i and j (depicted as w_{ij}) by the Manhattan distance of the corresponding color values.

$$w_{ij} = |r_i - r_j| + |g_i - g_j| + |b_i - b_j| \tag{4}$$

The use of this quite simple metric reduces the computational complexity of building the model. Nonetheless, the weights are accurate enough to build reliable spatial models which produce good segmentation results.

Graphs like these have been used many times in segmentation algorithms [21]. In most cases, an energy function is defined on the graph to evaluate a specific segmentation. This transforms the image segregation problem into a well-known minimization task. In the literature, there are different approaches for this, one example is [22], who use the cut-value as an energy function. A more elaborated energy function is NCut. It maximizes the association in the different regions while minimizing the cut between them [18, 23].

Approaches using NCut usually provide better results but finding the optimal solution is an NP-hard problem [23] which makes approximative methods necessary for the optimization step (e.g. spectral graph theory). Given a weighted graph $G = (V, E, w)$ and a partition $A \dot{\cup} B = V$ the NCut for that partition (segmentation) is defined as follows:

$$Ncut(A,B) = \frac{Cut(A,B)}{Assoc(A)} + \frac{Cut(A,B)}{Assoc(B)} \tag{5}$$

with the standard $Cut(A,B)$ and $Assoc(A)$ terminologies.

$$Assoc(A) = \sum_{i \in A, j \in V} w_{ij} \tag{6}$$

$$Cut(A,B) = \sum_{i \in A, j \in B} w_{ij} \tag{7}$$

This energy function is well suited for the evaluation of segmentations in single images but not for videos. There it can occur that the scene is completely free of foreground objects. These cases cannot be mapped by NCut as an 100% background segmentation would result in a division by zero. Therefore, this energy function inherently works with the false assumption that there are always foreground objects visible. Furthermore, NCut also favors segmentations with roughly equal amounts of fore- and background. If there is only a small amount of foreground, the corresponding association will attain a very small value and hence one of the summands in equation 5 will become very large. This prevents segmentations with only minor foreground or background areas.

These problems can be adressed with a modified normalized cut [20] which has no bias for any specific amount of foreground:

$$N^2cut(A,B) = \frac{Cut(A,B)}{nAssoc(A)} + \frac{Cut(A,B)}{nAssoc(B)}, \tag{8}$$

$$nAssoc(A) = \frac{Assoc(A) + 1}{\sum_{i \in A, j \in V, \exists e_{ij}} 1 + 1}. \tag{9}$$

In equation 9 the association is normalized by dividing by the number of edges contributing to the association. Consequently, the new association is the average edge value which is not dependent on the size of the set. The addition of one to the denominator and numerator of the fraction in equation 9 prevents the divisions by zero for empty sets. An obvious extension to this seems to be the normalisation of Cut(A) in the same way, but this is not reasonable. It would remove the favor of cuts that are short and would hence result in very long cuts zigzagging through the image. This would not reflect the smoothness of natural images.

The N^{2}Cut is based on the spatial information in one single image only. To get meaningful segmentations the temporal information derived form the GSM Background Subtraction have to be added. This is done by taking the GSM segmentation as a starting point for the N^{2}Cut optimization. Based on this a local optimization process is run and produces the final segmentation. It is important that the optimization algorithm is only local and may get stuck in local minima because this ensures that the basic structure of the segmentation is derived from the temporal information (GSM) and the N^{2}Cut optimization only makes it spatially coherent.

2.3 Markov Random Fields

Another way to add spatial information to the GSM results are Markov Random Fields (MRF). To achieve this the MRF described in [19] is used. It models the spatial relations between single pixels and hence forces the segmentation to be locally coherent.

The most important part of a MRF is the neighborhood system. We use a generalized Moore Neighborhood because it assures the homogeneity of the MRF and also can easily be changed in size. In the generalized Moore Neighborhood, the neighborhood for a pixel is defined by a square which is centered at that pixel and which can vary in size. The number of different combinations of neighbouring pixels (cliques) will increase radically with the size of the square. The input data, probabilities of being background or foreground for each pixel, is derived from the GSM Background Subtraction.

After constructing the MRF model of the spatial relations of the image, the most likely state (segmentation) of that model has to be computed. This maximum a posteriori (MAP) is very difficult to compute and can only be approximated for a problem of reasonable size. First a cost function is needed which can evaluate the different segmentations based on the MRF model. This function consists of two parts, one part measures how good the segmentation matches the GSM result. Basically, the smaller w_i, the higher is the penalty for labeling the pixel i as foreground. The second part of that function evaluates the spatial coherence of the segmentation. As our assumption is that natural images are smooth, neighbouring pixels should have the same label. If this is violated, there will be a penalty to the cost function.

This cost function is then converted to a factor graph and optimized with a loopy max-product Belief Propagation algorithm. Although this will only approximate the MAP, it can still take a long time and requires a lot of memory to do so. This is due to the fact that the amount of cliques increases so drastically with the size of the neighborhood. To reduce this effect we decided to simplify the model and only take one clique size (the largest cliques) into account. Also, the spatial component of the energy function was kept as simple as possible to further reduce the computational load. It returns zero if all neighbours of the pixel have the same label and one if at least one neighbour has a different label. These simplifications allowed us to build and optimize the MRF model on an 1920×1080 image in around one minute. Without them it would have been infeasible to do so in less than a week.

3 Results

To evaluate these algorithms we tested them first on the popular but old wallflower dataset. The results can be seen in Table 1 and show that our methods perform quite well in air and that N^{2}Cut clearly outperforms the Markov Random Fields there. Additionally to the accuracy increase, the optimization of the N^{2}Cut is also 2 orders of magnitude faster and can be done in real time.

For the evaluation in underwater environments no data sets are freely available at the moment. Hence, we took some underwater videos ourselves with a Go Pro Hero 3 and manually created some ground truth data for them. Two frames of these videos and the corresponding segmentations can be seen in Fig. 1. To measure the accuracy of these segmentations we use the F1-Score and Matthews Correlation Coefficient [24]. They are a better indicator of the quality of segmentations than the simple amount of wrongly classified pixels (which is the standard measure for the Wallflower dataset and was also used here for comparison reasons), especially when the amount of foreground is very small. The reason for this is that, the weight of foreground and background pixels changes according to the amount of foregound visible in the image.

In both pictures the N^{2}Cut performs substantially worse than the MRF algorithm (see Table 2). In the right image even the GSM Background Subtraction without any spatial model is better. This behaviour is quite constant in all the underwater videos we took, although not as strong as in these two selected examples. The reason for this is that the MRF approach smoothes the segmenation just based on the background subtraction result as opposed to the N^{2}Cut which alligns the segmentation to the nearest edges in the image. However, this allignment fails in underwater images because the blurring impedes any clear edges. This behavior is enhanced by the often low color disparity between fishes and the background, which enables them to hide from enemies. In the end, instead of aligning the segmentation to the edges the N^{2}Cut often degenerates foreground objects to simple rectangles because there are no clear edges to which the object can be alligned to. All in all, MRF is better suited as a spatial model in underwater situation if real-time capability is not an issue.

4 Future Work

In the future, we want to use some underwater image enhancement algorithms (mainly deblurring and color correction methods) on the images before the segmentation process starts. We hope that these will allow the N^{2}Cut to perform better and will give the same accuracy and speed advantages in underwater circumstances as it does achieve for air images.

Algorithm	Errors
Single Gaussian [7]	35133
Mixture of Gaussian (MoG) [8]	27053
Kernel Density Estimation [10]	26450
MoG with PSO [25]	13916
MoG in improved HLS Color Space [9]	9739
MoG with MRF [17]	3808
Gaussian Switch Model (GSM) [this paper]	9718
GSM with MRF [this paper]	7169
GSM with N^{2}Cut [this paper]	5064

Table 1. The results of different algorithms on the Wallflower [11] data set.

		GSM	GSM + MRF	GSM + N^{2}cut
Left Image	F1-Score:	0.990687	0.991428	0.982705
	MCC:	0.852013	0.879739	0.796699
Right Image	F1-Score:	0.995831	0.996647	0.996094
	MCC:	0.424601	0.540039	0.43656

Table 2. The F1-Score and Matthews Correlation Coefficient for the different segmentations in Fig. 1.

Acknowledgements

This work has been funded by the federal state of Mecklenburg-Vorpommern and EFRE within the project Basic and Applied Research in Interactive Document Engineering and Maritime Graphics.

References

1. Tae-Kyun Kim, Kwan-Yee Kenneth Wong, B. Stenger, J. Kittler, and R. Cipolla. Incremental linear discriminant analysis using sufficient spanning set approximations. In *Computer Vision and Pattern Recognition, 2007. CVPR '07. IEEE Conference on*, pages 1–8, June 2007.
2. S.S. Bucak, B. Gunsel, and O. Guersoy. Incremental nonnegative matrix factorization for background modeling in surveillance video. In *Signal Processing and Communications Applications, 2007. SIU 2007. IEEE 15th*, pages 1–4, June 2007.
3. Marghes, Bouwmans T., and Vasiu R. Background modeling and foreground detection via a reconstructive and discriminative subspace learning approach. In *Proceedings of the 2012 International Conferecne on Image Processing, Computer Vision and Patternrecognition*, pages 106–113, 2012.
4. G.T. Cinar and J.C. Principe. Adaptive background estimation using an information theoretic cost for hidden state estimation. In *Neural Networks (IJCNN), The 2011 International Joint Conference on*, pages 489–494, July 2011.
5. Horng-Horng Lin, Tyng-Luh Liu, and Jen-Hui Chuang. A probabilistic svm approach for background scene initialization. In *Image Processing. 2002. Proceedings. 2002 International Conference on*, volume 3, pages 893–896 vol.3, June 2002.

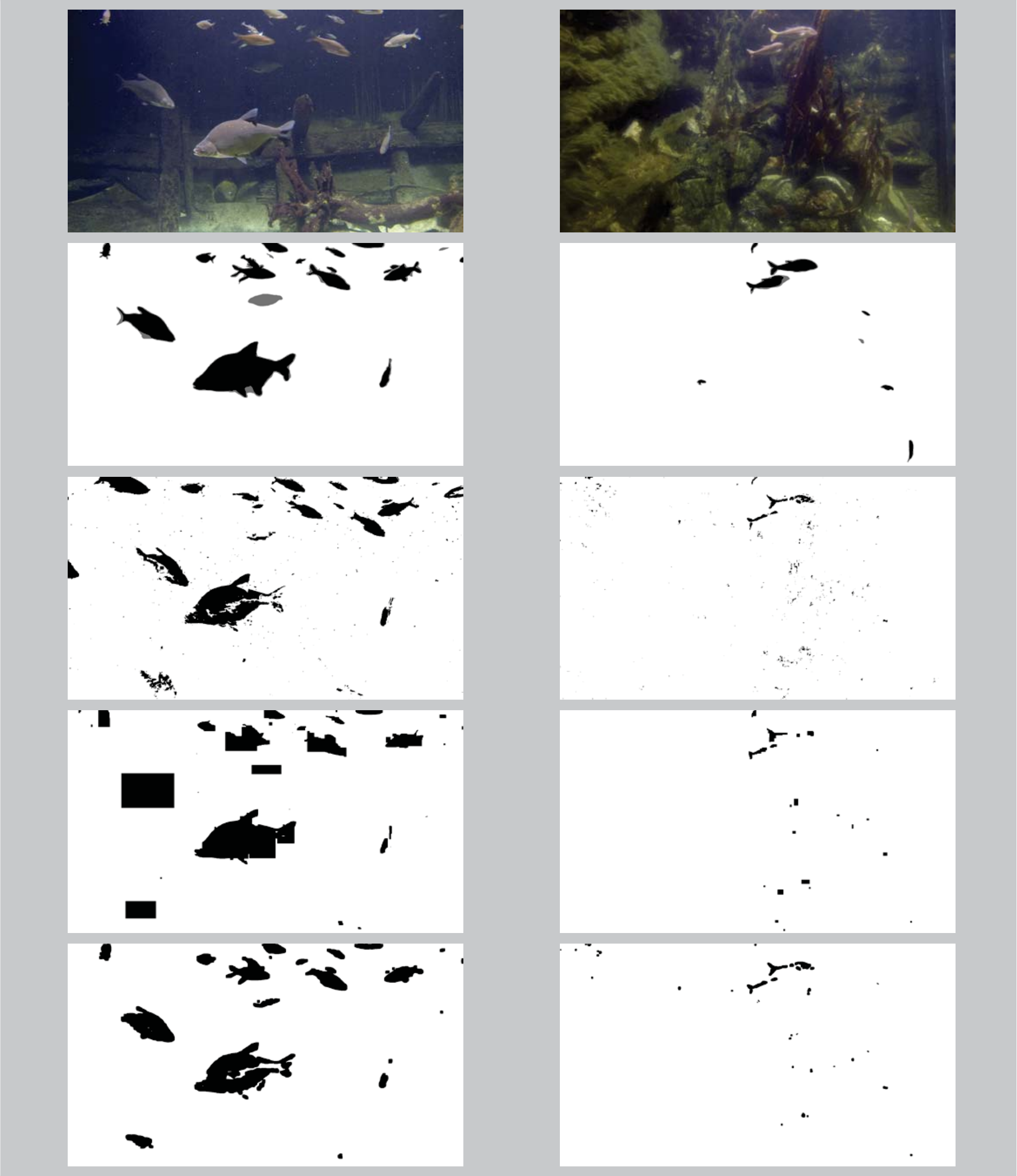

Fig. 1. Two exampls frames of the underwater images we took. The first row shows the original frame, the second row the hand segmented ground truth data, then the result after the background subtraction (GSM), the next row shows the segmentation after combining the N^{2}Cut with GSM and the last row shows the combination of GSM and MRF.

6. Shengping Zhang, Hongxun Yao, and Shaohui Liu. Dynamic background subtraction based on local dependency histogram. *International Journal of Pattern Recognition and Artificial Intelligence*, 23(07):1397–1419, 2009.
7. Christopher Wren, Ali Azarbayejani, Trevor Darrell, and Alex Pentland. Pfinder: Real-time tracking of the human body. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19:780–785, 1997.

8. Chris Stauffer and W.E.L. Grimson. Adaptive background mixture models for real-time tracking. In *Proceedings 1999 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Vol. Two*, pages 246–252. IEEE Computer Society Press, June 1999.
9. N. Setiawan, S. Hong, J. Kim, and C. Lee. Gaussian mixture model in improved ihls color space for human silhouette extraction. In *16th Int Conf on Artificial Reality and Telexistence*, pages 732–741, 2006.
10. Ahmed M. Elgammal, David Harwood, and Larry S. Davis. Non-parametric model for background subtraction. In *Proceedings of the 6th European Conference on Computer Vision-Part II*, ECCV '00, pages 751–767, London, UK, UK, 2000. Springer-Verlag.
11. Kentaro Toyama, John Krumm, Barry Brumitt, and Brian Meyers. Wallflower: Principles and practice of background maintenance. In *Seventh International Conference on Computer Vision*, pages 255–261. IEEE Computer Society Press, Septempber 1999.
12. K.-F. Loe Y. Wang and J.-K. Wu. A dynamic conditional random field model for foreground and shadow segmentation. *IEEE Trans. Pattern Analysis and Machine Intelligence*, pages 279–289, 2006.
13. Yuri Boykov and Gareth Funka-Lea. Graph cuts and efficient n-d image segmentation. *International Journal of Computer Vision*, 70:109–131, November 2006.
14. Xi Li, Weiming Hu, Zhongfei Zhang, and Xiaoqin Zhang. Robust foreground segmentation based on two effective background models. In *Proceedings of the 1st ACM International Conference on Multimedia Information Retrieval*, MIR '08, pages 223–228, 2008.
15. P. Pouladzadeh, S. Shirmohammadi, and A. Yassine. Using graph cut segmentation for food calorie measurement. In *Medical Measurements and Applications (MeMeA), 2014 IEEE International Symposium on*, pages 1–6, June 2014.
16. R. Hansch, O. Hellwich, and Xi Wang. Graph-cut segmentation of polarimetric sar images. In *Geoscience and Remote Sensing Symposium, 2014*, pages 1733–1736, 2014.
17. Konrad Schindler and Hanzi Wang. Smooth foreground-background segmentation for video processing. In *Proceedings of the 7th Asian Conference on Computer Vision - Volume Part II*, ACCV'06, pages 581–590, 2006.
18. M.A.G. de Carvalho, A.L. da Costa, A.C.B. Ferreira, and R. Marcondes Cesar Junior. Image segmentation using component tree and normalized cut. In *Graphics, Patterns and Images (SIBGRAPI), 2010 23rd SIBGRAPI Conference on*, pages 317–322, Aug 2010.
19. Martin Radolko and Enrico Gutzeit. Video segmentation via a gaussian switch background-model and higher order markov random fields. In *Proceedings of the 10th International Conference on Computer Vision Theory and Applications Volume 1*, pages 537–544, 2015.
20. Martin Radolko, Fahimeh Farhadifard, Enrico Gutzeit, and Uwe Freiherr von Lukas. Real time video segmentation optimisation with a modified normalized cut. In *Image and Signal Processing and Analysis, 9th International Conference on*.
21. Faliu Yi and Inkyu Moon. Image segmentation: A survey of graph-cut methods. In *Systems and Informatics (ICSAI), 2012 International Conference on*, pages 1936–1941, May 2012.
22. Yanmin Peng and Rong Liu. Object segmentation based on watershed and graph cut. In *Image and Signal Processing (CISP), 2010 3rd International Congress on*, volume 3, pages 1431–1435, Oct 2010.
23. Jianbo Shi and Jitendra Malik. Normalized cuts and image segmentation. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 22(8):888–905, Aug 2000.
24. B. W. Matthews. Comparison of the predicted and observed secondary structure of T4 phage lysozyme. *Biochim. Biophys. Acta*, 405:442–451, 1975.
25. B. White and M. Shah. Automatically tuning background subtraction parameters using particle swarm optimization. In *Multimedia and Expo, 2007 IEEE International Conference on*, pages 1826–1829, July 2007.