STATISTICAL FUSION OF MULTIPLE CUES FOR IMAGE TAMPERING DETECTION

Yu-Feng Hsu and Shih-Fu Chang

Department of Electrical Engineering
Columbia University
{yfhsu,sfchang}@ee.columbia.edu

ABSTRACT

Recent image forensic research has resulted in a number of tampering detection techniques utilizing cues culled from understanding of various natural image characteristics and modeling of tampering artifacts. Fusion of multiple cues provides promises for improving the detection robustness, however has never been systematically studied before. By fusing multiple cues, the tampering detection process does not rely entirely on a single detector and hence can be robust in face of missing or unreliable detectors. In this paper, we propose a statistical fusion framework based on Discriminative Random Fields (DRF) to integrate multiple cues suitable for forgery detection, such as double quantization artifacts and camera response function inconsistency. The detection results using individual cues are used as observation from which the DRF model parameters and the most likely node labels are inferred indicating whether a local block belongs to a tampered foreground or the authentic background. Such inference results also provide information about localization of the suspect spliced regions. The proposed framework is effective and general - outperforming individual detectors over systematic evaluation and easily extensible to other detectors using different cues.

1. INTRODUCTION

With the ease of digital image manipulation, image forgeries through operations like copy and paste (splicing) have become frequent concerns in many applications such as news reporting and criminal investigation. Therefore, verification of content integrity has become increasingly important. By analyzing the image formation process, several works have been proposed for tampering detection. Some explore natural scene properties such as consistent lighting directions [1]. Some use device characteristics such as demosaicking filters [2, 3], CCD sensor noise [4], and Camera Response Function (CRF) [5, 6, 7]. Some are based on post-processing artifacts such as Double Quantization (DQ) effects [8]. These approaches are all passive - no active mechanisms are needed to generate and embed watermarks. Several of them [1, 5, 6, 7] are blind approaches, namely, no prior knowledge of the image or list of known sources is required.

An alternative way to categorize these approaches is based on the type of the detection output - local authenticity or spatial inconsistency. The former focuses on individual sites (pixels or blocks or regions) and analyzes the level of authenticity for each site [1, 2, 3, 4, 8]. The latter takes two candidate sites and verifies whether they come from the same source [5, 6, 7]. These two classes of detectors are of different natures and complement each other. The goal of this paper, therefore, is to propose a sound framework to fuse them in order to improve detection robustness. When one set is unreliable or noisy, we can rely on another set toward a more accurate detection of tampering and localization of the spliced object. To the best of our knowledge, this is the first systematic study of techniques fusing different types of image forgery cues.

We formulate the fusion task as a labeling problem and adopt Discriminative Random Field (DRF) [9] as the fusion framework. The detector outputs are treated as observations and used to recover the hidden labels indicating whether each block in the test image belongs to the foreground spliced object or the authentic background. Experimental results show that fusion is advantageous over individual detectors, both in terms of inference accuracy and object localization. We also propose our unconventional edge structure and justify the assignment of inconsistency scores to block pairs, as supported by additional results.

2. IMAGE TAMPERING DETECTION MODULES

We first review two representative image tampering detectors, one based on local authenticity and the other based on spatial consistency analysis. We will then present a general framework in the next section for fusing individual detectors like these two.

2.1. Local Authenticity Score: Double Quantization

The Double Quantization (DQ) detector explores the widely used JPEG image compression format. As shown in Fig. 1, most spliced images are created using two source images, which are often stored in the JPEG format, a second pass of quantization of applied in addition to the first pass used in the original compression. After splicing, a Double Quantization (DQ) effect (see Fig. 1) can be found in the transform coeffi-

cient histograms of the background region. Such effect results in periodical peaks and/or valleys as opposed to the smooth patterns in the distributions. It will not appear in the foreground regions which either have been quantized only once or using different quantization block structures in two passes.

By detecting abnormal histogram shapes, one can distinguish which 8x8 DCT blocks have been quantized only once and which have been quantized twice [8]. The output is a likelihood map measuring the probability of the DQ effect presence. Usually the foreground object is of lower DQ scores and background of higher scores, however this can be reversed because it is possible that the foreground was quantized twice but not the background. Each 8x8 block is associated with one DQ score between $[0, 1]$ and we will refer to it as a_i (for ith block in the image) in the following sections.

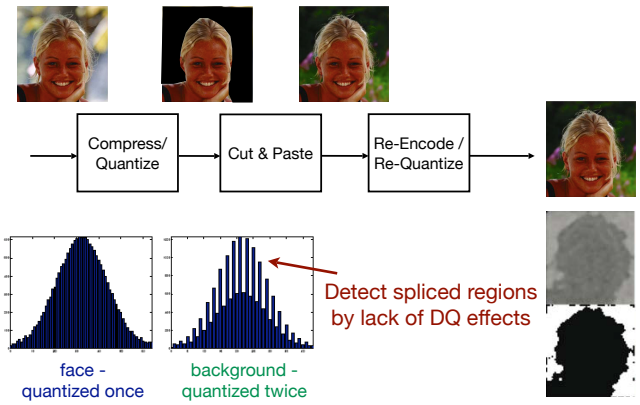


Fig. 1: Authenticity scores a_i's of individual blocks can be estimated by detecting double quantization effect introduced during splicing

2.2. Inconsistency Score: Camera Response Function

The inconsistency output from our previous works [6, 7] is built upon one type of specific device characteristics - Camera Response Function (CRF), the concave function that maps incoming irradiance to cameras to the final intensity data stored in the output image. The hypothesis is different areas within a spliced image should exhibit inconsistent CRF attributes if they come from different sources. We use the Locally Planar Irradiance Points (LPIPs) extracted from a single channel image to estimate the CRF [10] and Support Vector Machine (SVM) classifiers to verify whether points in two different regions of the same image are subject to the same CRF. Each image is first segmented into regions. The SVM classifier is applied to predict whether two neighboring regions sharing a boundary are produced by the same source (thus the same CRF). Our prior experiments over a challenging, heavily post-processed data set showed an accuracy of 70% precision and 70% recall in detecting spliced images. Unlike local authenticity detectors, the output of such inconsistency checking (referred to as c_{ij} later in the paper) indicates the inconsistency relation between neighboring areas. The higher c_{ij}, the more likely the boundary between the areas is caused by splicing.

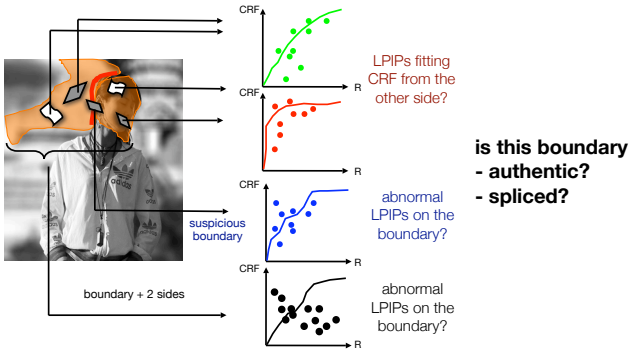


Fig. 2: Spatial inconsistency scores can be estimated by cross fitting Camera Response Functions to Locally Planar Irradiance Points from neighboring segmented regions

3. FORMULATION OF THE FUSION PROBLEM

The objective of this work is to develop a sound and effective framework for fusing multiple image tampering detectors. On one hand, as different detectors explore cues and artifacts at different stages of the image formation process, if an image lacks certain cues, the corresponding detector would not be of use and we have to rely on other detectors for a correct decision. For example, a spliced image may be created with two source images of similar quantization settings but very different cameras. In this case, the splicing will be successfully detected by the CRF inconsistency checker but not the DQ detector. We thus benefit from having both modules at hand rather than only using the DQ detector. On the other hand, if one detector outputs noisy, erroneous scores, having other detectors at hand makes it possible to complement and correct such unreliable decisions. Therefore, the advantage of fusion is twofold: to handle images undergoing diverse types of tampering and to boost the detection accuracy by making different modules work with each other.

The challenge, however, lies in the difficulty harnessing the diversity of detectors. Different detectors are developed based on distinct physical motivations and report decisions based on different segmentation structures. For example, the authenticity scores from the DQ detector cannot be directly combined with the inconsistency scores from the CRF analysis as they carry information of completely different natures. Furthermore, the DQ detector computes a score for each 8x8 pixel DCT block while the CRF inconsistency scores are assigned to two arbitrarily shaped regions sharing a common boundary.

We formulate the fusion task described above as a labeling problem, and adopt the well known Markov Random Fields (MRF) framework. The following subsections define the labeling problem, provide basic review of MRF and a specific variation, Discriminative Random Field (DRF), that fits very well to our problem setting described above.

To reconcile the differences in image segmentation used by different detectors, we adopt the fixed size 8x8 pixel blocks of JPEG as the common data unit since the arbitrary shape

segmented regions are usually larger than an 8x8 block. The DQ score is readily computed for each block, while the CRF inconsistency score between two given blocks can be computed based on the score between the two regions that contain these blocks.

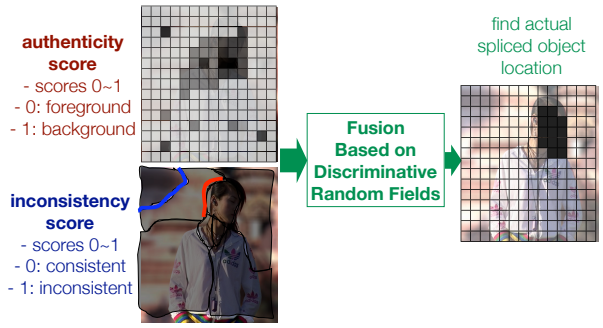


Fig. 3: The proposed framework fuses individual tampering detection scores to infer the likely splicing boundary

3.1. Labeling Problem and Markov Random Field

In a typical labeling problem, each node i has a corresponding label y_i which takes on binary values $\{-1, +1\}$. These labels are usually hidden and unobserved. What is observed is the noisy single node signal x_i at node i and pairwise signal z_{ij} between nodes i and j. The labeling problem starts with observations $\mathbf{x}, \mathbf{z}$ and attempts to recover the unknown labels $\mathbf{y}$.

Markov Random Field (MRF) offers well-established theories for solving such labeling problems. Mathematically, the MRF formulation looks for *maximum a posteriori* (MAP) labels $\mathbf{y}$ based on single node observations $\mathbf{x}$.

$$\begin{aligned}\hat{\mathbf{y}} &= \arg\max_{\mathbf{y}} p(\mathbf{y}|\mathbf{x}) = \arg\max_{\mathbf{y}} p(\mathbf{x}|\mathbf{y})p(\mathbf{y}) \\ &= \arg\max_{\mathbf{y}} \prod_i p(x_i|y_i) \prod_{i,j} p(y_i, y_j) \qquad (1)\end{aligned}$$

Note traditional MRF requires models for emission probabilities $p(x_i|y_i)$, where Gaussians or Mixture of Gaussians are widely used. Also, it only includes single-node observations $\mathbf{x}$ but not pairwise inconsistency scores $\mathbf{z}$. The relation between two nodes i, j is purely a smoothness constraint, enforced by the prior $p(y_i, y_j)$ which usually favors $y_i = y_j$ and penalizes $y_i \neq y_j$. Lastly, as exact MAP solutions for $\mathbf{y}$ is intractable, there have been a significant number of approximate solutions developed, including Mean Field (MF) and Loopy Belief Propagation (LBP) [11, 12].

3.2. Unconventional Edge Structure

Fig. 4 shows an illustrative example on a 24-block (4x6) image segmented into 3 areas. As shown in Fig. 4(a), traditional MRFs are built only upon neighboring blocks following the Markov assumption. However, our inconsistency scores c_{ij} are defined across segmentation boundaries. As long as there is a score for the boundary segment between two areas, any block pair from these two areas has a well-defined c_{ij}, even

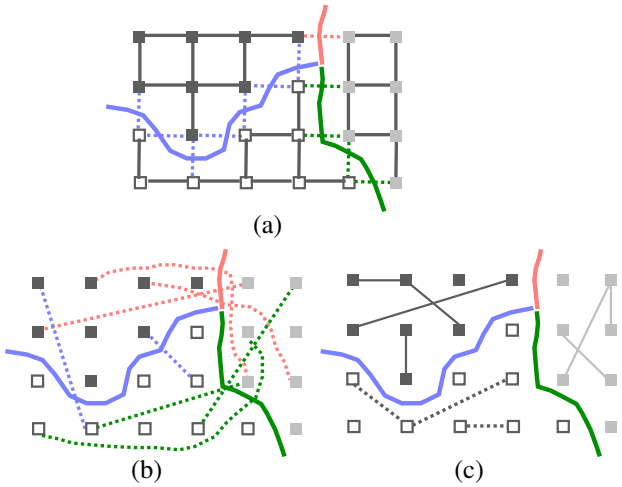


Fig. 4: Edge structures (a) traditional MRF (b) relaxed structures that link non-adjacent blocks across the segmentation boundary (c) relaxed structures that link blocks within same area

though these two blocks might not be neighbors of each other (Fig. 4(b)). To better utilize the power of our inconsistency scores (instead of limiting them only to the boundaries, as the colored dashed lines in Fig. 4(a)), we use an unconventional edge structure which relaxes the Markov assumption.

For any two areas with a shared boundary (eg., black and gray sharing the pink boundary in Fig. 4(b)), we randomly sample a number of block pairs and assign the c_{ij} associated with the shared boundary to these block pairs (Fig. 4(b), color coded in the same way as the boundaries). For block pairs within the same area (Fig. 4(c)), we make a simple assumption and trust the segmentation results and determine they are strictly consistent with each other, receiving 0 as their assigned c_{ij}'s.

3.3. Discriminative Random Field

Although the traditional MRF offers a seemingly straightforward solution to the labeling problem, it does not match well our problem setting and objectives. Firstly, the inconsistency observations $\mathbf{z}$ are not included. Secondly, the emission probability used in Eqn. 1 is not of the greatest interest to us. Instead, the posterior probability of $\mathbf{y}$ given $\mathbf{x}$ and $\mathbf{z}$ is more relevant to our goal. To this end, we adopt a slightly different framework Discriminative Random Field (DRF) [9], an extension of the Conditional Random Field family. The DRF model is represented as

$$\begin{aligned}\hat{\mathbf{y}} &= \arg\max_{\mathbf{y}} p(\mathbf{y}|\mathbf{x}, \mathbf{z}) \\ &= \arg\max_{\mathbf{y}} \prod_i p(y_i|x_i) \prod_{i,j} p(y_i, y_j|z_{ij}) \qquad (2)\end{aligned}$$

Note the DRF also looks for the MAP labels $\mathbf{y}$. However, rather than modeling the emission probabilities $p(x_i|y_i)$, it directly maximizes the posterior probabilities $p(y_i|x_i)$ and $p(y_i, y_j|z_{ij})$. This makes the model consistent with our optimization objective, focusing on the posterior information. It also avoids the extra loop through generative models. Unlike

in traditional MRF where the only pairwise relation is spatial smoothness, the DRF includes pairwise inconsistency as extra observations in addition to single node observations.

We use logistic models for posterior probabilities $p(y_i|x_i)$ and $p(y_i, y_j|z_{ij})$, parameterized by vectors $\mathbf{w}$ and $\mathbf{v}$, as used in the original DRF work proposed in [9]:

$$\begin{aligned} p(y_i|x_i) &= (1 + e^{-y_i \mathbf{w}^T \mathbf{x_i}})^{-1} \\ p(y_i, y_j|z_{ij}) &= (1 + e^{-y_i y_j \mathbf{v}^T \mathbf{z_{ij}}})^{-1} \end{aligned} \quad (3)$$

This choice is theoretically justified in that the logistic model is a natural form of posterior probabilities if the emission probability belongs to the exponential family [13]. As most real world data follow exponential family distributions, using logistic models for posteriors is a sensible choice.

In this paper, as we are using one detector for authenticity scores and one for inconsistency scores, both single node observation vector $\mathbf{x}_i$ and pairwise observation vector $\mathbf{z}_{ij}$ are of dimension 2: $\mathbf{x}_i = [1, a_i]^T, \mathbf{z}_{ij} = [1, c_{ij}]^T$. If the number of detectors is to be increased, the DRF framework can be easily adapted by appending additional scores to $\mathbf{x}_i$ or $\mathbf{z}_{ij}$, resulting in higher dimensional observation vectors.

3.4. Learning Discriminative Random Field

Having identified an appropriate solution to our fusion problem, we follow the standard learning procedure for DRFs. The learning process iterates between two steps: parameter estimation (look for optimal $\mathbf{w}$, $\mathbf{v}$) and inference (look for optimal $\mathbf{y}$ based on the estimated $\mathbf{w}$, $\mathbf{v}$ of the current step). As the exact MAP solution for $\mathbf{y}$ is intractable, we use MF and LBP as inference engines. Among these two options, LBP achieves higher inference accuracy and better converges behavior, therefore we report results based on LBP only. We use the open source CRF2D toolbox implemented by [14].

The learning procedure is outlined as follows:

1. Randomly initialize parameters $\mathbf{w}$ and $\mathbf{v}$
2. Based on current parameters $\hat{\mathbf{w}}$, $\hat{\mathbf{v}}$, infer labels $\hat{\mathbf{y}}$
3. Based on current labels $\hat{\mathbf{y}}$, update parameters $\hat{\mathbf{w}}$ and $\hat{\mathbf{v}}$
4. Iterate between steps 2 and 3 until convergence

The learning stops when the total number of iterations have reached 5. Thanks to the LBP inference engine, we have observed that most test cases converge quite fast, usually by the third run.

Note the above inference process is completely unsupervised. There is no need for annotation of training sets. Given a new test image, the optimal parameters and the inferred labels are estimated.

4. EXPERIMENTS

We conduct our experiments on a data set of 90 spliced images taken with four cameras: Canon G3, Nikon D70, Canon EOS 350D, and Kodak DCS330. These cameras range from low-end point-and-shoot models (Kodak DCS330) to high-end DSLR models (Nikon D70, Canon 350D) so that we may ensure different qualities and settings in captured images. Each spliced image has content from exactly two cameras, with visually salient objects from one image copied and pasted onto another using Adobe Photoshop without post processing. They are originally stored in uncompressed TIFF format. To obtain DQ scores, we create the spliced images by first compressing authentic images with a higher quality factor (Q=85), copying and pasting the foreground object, then recompressing the spliced images with a lower quality factor (Q=70). We choose these quality factors following the settings used in [8].

Typical image sizes range from 757x568 to 1152x768 pixels. This results in 94x71 (a total of 6674) to 144x96 (a total of 13824) 8x8 DCT blocks within each image. The number of randomly sampled c_{ij} edges is fixed regardless of image size. We select 250,000 block pairs across segmentation boundaries and 250,000 for block pairs from the same segmented area. The computation time varies with the convergence behaviors. A random field of this size can take 10 minutes or as long as 60 minutes to reach a steady state on a 2GHz quad core CPU machine.

The DQ scores a_i's are generated over 8x8 DCT blocks. The inconsistency scores from the CRF checker is associated with boundaries between adjacent segmented regions. As discussed in Sec. 3, we use the finest segmentation granularity (in this case 8x8 DCT blocks) as the labeling unit and compute the c_{ij} of a block pair using the inconsistency score of the boundary separating the regions that contain the two blocks.

For each image, we run 5 different $\mathbf{w}$, $\mathbf{v}$ initializations. The performance is evaluated by the inference accuracy, i.e., percentage of correctly inferred labels within the image.

5. RESULTS

Inference results from 90 test images comparing $a_i + c_{ij}$ fusion against a_i-only (DQ-only) are evaluated. For every image, the detection accuracy is averaged over its own 5 runs. Overall average accuracy (across all 90 images) increases from 80.87% to 83.49% when fusing a_i's with c_{ij}'s. Among these 90 images, the most significant improvement can reach as high as 18.44%.

Fig. 5 shows two sets of visual examples. It shows that the fusion does not only increase the inference accuracy, but also leads to more compact inference outcome: the detected foreground object is connected, rather than the scattered blocks as those obtained by using a_i alone. This is desirable because in practical scenarios, the spliced object tends to be a compact, connected component.

In the following we justify our edge weight assignment between blocks in the same segmented area. Recall in Sec 3.2, we use actual CRF inconsistency scores for block pairs belonging to two different areas, and assign zero as the c_{ij}'s between block pairs in the same segmented areas. This may appear restrictive, as the fact that they belong to the same segmented area does not guarantee their source consistency. The automatic segmentation process may miss the actual splicing boundary and thus blocks in the same segmented area may

still come from different sources. To study the effect of such assignment, we drop these edges in the random field and only keep c_{ij}'s from across segmented boundaries.

The average inference accuracy over 90 images drops from 83.49% to 80.25%, exhibiting no advantage over a_i alone (80.87%). The degradation can be as dramatic as 39.37%. This implies that although same area block pairs do not have proper c_{ij} scores, the image segmentation itself can still serve as another source of inconsistency scores - blocks in the same segmented area are more likely to come from the same source. Therefore, we are essentially fusing three sets of scores: single node scores from the DQ detector, neighboring area block pair inconsistency scores from the CRF checker, and same area block pair inconsistency scores from automatic image segmentation.

The role of zero c_{ij}'s is obvious by showing the visual examples of detection results in Fig. 6. With such scores, the inference favors same labels for the blocks belonging to the same segmented area, resulting in a more connected foreground object whose shape loosely follows that of the segmentation boundary.

6. CONCLUSION

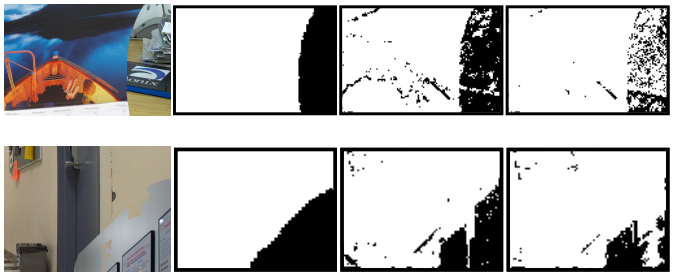
In this paper, we proposed a general, effective framework for fusing multiple cues for image tampering detection. We addressed the challenges in integrating results from diverse tampering detection components that explore different physical characteristics and different segmentation granularities. We formulated it as a labeling problem and apply Discriminative Random Field based methods to incorporate both local-block authenticity and inter-block inconsistency measures. The process is completely unsupervised, without the need of any training data annotation. Results have shown the advantage of fusion over individual detectors, both in terms of accuracy and visual compactness of the detection results. This framework is not restricted to the use of any specific cues or detectors and hence can be easily extended to include other tampering detection methods.

7. ACKNOWLEDGEMENTS

This work has been sponsored by NSF Cyber Trust program under award IIS-04-30258.

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

Fig. 5: Visual examples: (a) original image (b) ground truth label (c) fusion output (c) a_i-only output



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

Fig. 6: Visual examples: (a) original image (b) ground truth label (c) fusion, including zeros c_{ij}'s (c) fusion, dropping zeros c_{ij}'s

8. REFERENCES

[1] M.K. Johnson and H. Farid, "Exposing digital forgeries by detecting inconsistencies in lighting," in *ACM Multimedia and Security Workshop*, 2005.

[2] A.C. Popescu and H. Farid, "Exposing digital forgeries in color filter array interpolated images," *IEEE Transactions on Signal Processing*, vol. 53, no. 10, 2005.

[3] A. Swaminathan, M. Wu, and K.J.R. Liu, "Component forensics of digital cameras: a non-intrusive approach," in *Conference on Information Sciences and Systems*, 2006.

[4] J. Lukáš, J. Fridrich, and M. Goljan, "Detecting digital image forgeries using sensor pattern noise," *Proceedings of the SPIE*, vol. 6072, 2006.

[5] Z. Lin, R. Wang, X. Tang, and H.-Y. Shum, "Detecting doctored images using camera response normality and consistency," in *International Conference on Computer Vision and Pattern Recognition*, 2005.

[6] Y.-F. Hsu and S.-F. Chang, "Detecting image splicing using geometry invariants and camera characteristics consistency," in *International Conference on Multimedia and Expo*, 2006.

[7] Y.-F. Hsu and S.-F. Chang, "Image splicing detection using camera response function consistency and automatic segmentation," in *International Conference on Multimedia and Expo*, 2007.

[8] J. He, Z. Lin, L. Wang, and X. Tang, "Detecting doctored jpeg images via dct coefficient analysis," in *ECCV (3)*, 2006.

[9] S. Kumar and M. Hebert, "Discriminative random fields," *International Journal of Computer Vision*, vol. 68, no. 2, 2006.

[10] T.-T. Ng, S.-F. Chang, and M.-P. Tsui, "Using geometry invariants for camera response function estimation," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2007.

[11] K. P. Murphy, Y. Weiss, and M. I. Jordan, "Loopy belief propagation for approximate inference: An empirical study," in *Uncertainty in Artificial Intelligence*, 1999.

[12] Y. Weiss, "Comparing the mean field method and belief propagation for approximate inference in mrfs," in *Saad and Opper, editors, Advanced Mean Field Methods, MIT Press*, 2001.

[13] M. I. Jordan, "Why the logistic function? a tutorial discussion on probabilities and neural networks," Tech. Rep., Computational Cognitive Science 9503, Massachusetts Institute of Technology, 1995.

[14] S. V. N. Vishwanathan, N. N. Schraudolph, M. W. Schmidt, and K. P. Murphy, "Accelerated training of conditional random fields with stochastic gradient methods," in *International Conference on Machine Learning*. 2006, ACM Press.