Abstract—In this paper, we focus on image tampering detection and tampered region localization. We find that the probability distributions of the DCT coefficients of a JPEG image will be influenced by tampering operation. Hence, we model the distributions of AC DCT coefficients of JPEG image and detect the tampered region from the unchanged region by using their different distributions. Based on an assumption of Laplacian distribution of unquantized AC DCT coefficients, Laplacian Mixture Model (LMM) is employed to model the quantized AC DCT coefficient distribution of a suspicious JPEG image. With the help of Expectation Maximization (EM) algorithm, the probability of an 8 × 8 block being tampered can be estimated; and then, a sophisticated image segmentation method, graph cut, is applied to determine the tampered region. Extensive experimental results on large scale databases prove the effectiveness of our proposed method which is suitable for different tampered region sizes at all levels including pixel, region and image level.

I. INTRODUCTION

With advanced softwares, digital images can be easily manipulated for bad purpose. Forgeries images always leave no visual clues of tampering. With the wide popularization of the Internet and social networks (such as Facebook, Twitter etc.), people can easily download or upload a tampered image without verification. This is actually quite harmful for us to know and face the true world correctly, especially for young people who lean upon the e-world badly. Seeing is no longer believing. How to verify the authenticity and integrity of a digital image is very important for image forensics and information security.

Digital watermarking has been proposed as an active way to protect the integrity of digital content. However, watermark should be embedded into an image when it is generated or before it is distributed. This is not desired for most practical uses. In contrast to the active way, passive way directly gathers evidence of tampering from images themselves. It has more potential for practical use and gains more attentions among researchers in image forensics.

There are several techniques for passive image tampering detection proposed in the recent literature. These include techniques for detecting the inconsistency of lighting [1], resampling artifacts [2], CFA aberrations [3], absence of camera sensor’s noise pattern [4] and some machine learning based methods [5]. However, most of the methods mentioned above are vulnerable to JPEG compression even with high quality factor.

Since JPEG is the most widely used image format, techniques specially designed for detecting JPEG image tampering are becoming indispensable. Popescu proposed a double JPEG compression detection approach in [6], however, detecting double JPEG compression does not necessarily prove malicious tampering. Some researchers tried to distinguish cropped-and-recompressed from single compressed using block artifacts, e.g. [7]. However, how to use the method for tampering detection is not mentioned there. Ye et al. [8] tried to estimate block artifacts as an evidence of tampering, however, it required an estimate of quantization step from an assumed unchanged part which cannot be compressed twice. In [9], Farid proposed a method which can detect whether the part of an image was initially compressed at a lower quality than the rest of the image. However, the method needs to exhaustively recompress the test image using every possible primary quality factor to locate the tampered region. A complementary approach was proposed in [10]. Lin et al. argued that for a tampered JPEG image, the DCT coefficients in its unchanged region undergone double quantization while those in the tampered region could be considered as being quantized only once. They proposed a method to estimate the posterior probability of 8 × 8 block being tampered. However, according to our implementation, the periodicity of histogram of DCT coefficients cannot be accurately estimated, which result in degrading the performance of their method. The DCT coefficient histogram of the image does not always have periodicity since it is the sum of two histograms, one is the part of an image was initially compressed at a lower quality than the rest of the image. However, the method is vulnerable to JPEG compression even with high quality factor.

In this paper, we also focus on detecting a tampered JPEG image and locating the tampered region. Instead of estimating the periodicity, with the help of Expectation Maximization (EM) algorithm, we get the maximum likelihood estimate of the first quantization step of the unchanged region and meanwhile the estimate of the tampered region proportion. Sequentially, a tampered 8 × 8 block can be detected by calculating its posterior probability of being tampered with the estimated values. Based on an assumption of Laplacian distribution of unquantized AC DCT coefficient, Laplacian Mixture Model (LMM) is employed to model the AC DCT
coefficient distribution of the tampered image. To the best of our knowledge, we may be the first to try to estimate the primary quantization step of the doubly quantized region (the unchanged region) in the tampered image. Most previous work estimated the primary quantization of a doubly JPEG compressed image with no singly quantized region [6] or assumed doubly quantized region [8], which is not very suitable for real tampering localization. To locate the tampered region, a sophisticated image segmentation method, graph cut, is utilized to minimize an energy function defined by the estimated posterior probability. Fig. 1 shows the flowchart of our proposed method for tampered image in JPEG format.

The rest of this paper is organized as follows. Related background including JPEG image tampering model is introduced in Section II. Section III mainly introduces our proposed algorithm for tampered region localization. The experimental results and analysis are given in Section IV. Conclusions are drawn in Section V.

II. BACKGROUND

Images coming from digital single lens reflex (DSLR) camera basically have two formats: RAW and Jpeg, while most consumer cameras only produce JPEG images. For RAW image, it has to be processed (e.g. demosaic) and converted to lossless (Tiff) or lossy compressed (Jpeg) image with external tools. Hence, according to image format, tampering often can be classified into following ways: Tiff/Tiff, Jpeg/tiff and Jpeg/Jpeg. In this paper, we focus on last two types of tampering. Tampering always occurs in spatial domain and tampered images are usually saved in Jpeg format. Our approach focuses on quantization stage of JPEG compression. We try to use different quantization effect between the unchanged and the tampered region to locate tampering.

A. Image tampering model

We model image tampering process in three steps similar as Lin et al. did in [10] (see Fig. 2).

1) Chose a JPEG image $I_0$ and decompress it.
2) Replace a region of $I_0$ with a new region coming from $I_0$ or another image $I_1$ (either JPEG compressed or not).
3) Save the tampered image in JPEG format or lossless compressed format (e.g. Tiff).

For lossless compressed tampered image, we resave the image as JPEG format with a compression quality 95 before detection starts.

We assume that if the tampered region is from a JPEG image, its JPEG blocks would be misaligned with lattice of the unchanged region (the probability of alignment is 1/64). Based on that, DCT coefficients of the tampered region are quantized once while those of the unchanged region are quantized twice [10]. Hence, the DCT coefficients of the unchanged and the tampered regions should have different distributions.

B. DCT coefficients distributions of tampered JPEG image

The double quantization effect has been discussed in [6] and [10]. It is obvious that doubly quantized DCT coefficients have periodically missing values or their probability distributions show peaks and valleys periodically, which are different from singly quantized DCT coefficients. In this section, we focus more on modeling the distributions of singly and doubly quantized AC DCT coefficients first and then Laplacian Mixture Model (LMM) is introduced to model the AC DCT coefficient distribution at any frequency of tampered JPEG image.

Unquantized AC DCT coefficient is always modeled using generalized Gaussian distribution or Laplacian distribution [11]. We adopt Laplacian distribution since it is more tractable both mathematically and computationally. The probability density function of a Laplacian distribution can be written as

$$f(x) = \frac{\lambda}{2} \exp\left(-\lambda|x|\right)$$

which is characterized by the single parameter $\lambda$. $x$ is unquantized AC DCT coefficient at a frequency.

Denoting the unquantized, singly quantized by factor $q_1$, and doubly quantized by $q_2$, and doubly quantized by $q_1$ followed by $q_2$ DCT coefficient by $v_0$, $v_1$ and $v_2$, respectively, we achieve

$$v_1 = \left[\begin{array}{c} v_0 \\ q_1 \end{array}\right], \quad v_2 = \left[\begin{array}{c} v_0 \\ q_1 \\ q_2 \end{array}\right] .$$

Hence, the probabilities of $\{v_1 = k\}$ and $\{v_2 = k\}$ can be calculated as follows$^1$,

$$P_1\{v_1 = k\} = \int_{q_2(k-\frac{1}{2})}^{q_2(k+\frac{1}{2})} f(v_0) dv_0,$$

$$P_2\{v_2 = k\} = \int_{q_1\left(\frac{q_2(k-\frac{1}{2})}{q_1}\right)^{-\frac{1}{2}}}^{q_1\left(\frac{q_2(k+\frac{1}{2})}{q_1}\right)^{-\frac{1}{2}}} \int_{q_2(k-\frac{1}{2})}^{q_2(k+\frac{1}{2})} f(v_0) dv_0 .$$

Assuming the proportions of the tampered region and the unchanged region are $\alpha_1$ and $\alpha_2$ respectively, we get the probability of AC DCT coefficient distribution at any frequency as

$$P\{v = k\} = \alpha_1 P_1\{v_1 = k\} + \alpha_2 P_2\{v_2 = k\} ,$$

$$\alpha_1 + \alpha_2 = 1 .$$

In the above Laplacian Mixture Model (LMM), $\alpha_1, \alpha_2, \lambda$ and $q_1$ are unknown factors. If these parameters are solved, we can get the posterior probability of each DCT block being tampered using Bayesian inference.

$^1$Without loss of generality, we assume $k > 0$. Based on the symmetry of (1), we can easily get the probabilities when $k \leq 0$. 

![Fig. 2. Three steps of JPEG image tampering model [10]. (a) Decompress a JPEG image (the grid represents $8 \times 8$ DCT blocks) and choose a region (inside the dashed curve) to alert. (b) Replace the region (shaded) with new content. (c) Save the tampered image in JPEG format. After three steps, the unchanged region (blank blocks) is doubly quantized while the tampered region (shaded blocks) is singly quantized.](image-url)
III. TAMPERED REGION LOCALIZATION

In this section, we employ Expectation Maximization (EM) algorithm to get the maximum likelihood estimates of parameters in LMM (5). Then, the posterior probability of an 8 \times 8 block being tampered is calculated with the estimated parameters. To locate the tampered region, a sophisticated image segmentation method, graph cut, is utilized to minimize an energy function defined by the estimated posterior probability.

A. Posterior probability estimation

The EM algorithm is a general method of finding the maximum likelihood estimates of the parameters of an underlying distribution from a given data set when the data is incomplete or has missing values [12]. In our scenario, observed data is DCT coefficients while the label (indicating where the data come from, the tampered region or the unchanged region) of each DCT coefficient is missed.

We assume the following probabilistic model:

\[
P(x | \Theta) = \alpha_1 P_1 (x | \lambda) + \alpha_2 P_2 (x | \lambda, q_1),
\]

where the parameters are \( \Theta = (\alpha_1, \alpha_2, \lambda, q_1) \) and \( \alpha_1 + \alpha_2 = 1 \). We consider the observed DCT coefficients \( X \) as incomplete and posit the existence of unobserved data items \( Y \) whose values inform us which components “generated” DCT coefficients. That is, we assume that \( y_i \in \{1, 2\} \) for each \( i \), and \( y_i = k \) if the \( i^{th} \) sample was generated by the \( k^{th} \) mixture component.

The EM algorithm first finds the expected value of the complete-data log-likelihood \( \log P(X, Y | \Theta) \) with respect to \( Y \) given \( X \) and the current parameter estimates. That is, we define:

\[
Q(\Theta, \Theta^{i-1}) = E \left[ \log P(X, Y | \Theta) | X, \Theta^{i-1} \right]
\]

where \( \Theta^{i-1} \) are the current parameters estimates that we used to evaluate the expectation and the M-step is to maximize the expectation.

\[
\Theta^i = \arg \max Q(\Theta, \Theta^{i-1})
\]

The expectation and maximization steps are repeated as necessary. Each iteration is guaranteed to increase the log-likelihood and hence the algorithm is guaranteed to converge to a local maximum of the likelihood function.

Therefore, for each AC frequency, we can get the optimal estimations of \( \alpha_1, \alpha_2, \lambda \) and \( q_1 \). The posterior probability of \( i^{th} \) DCT block being singly \( (P \{ y_i = 1 | x_i \}) \) and doubly \( (P \{ y_i = 2 | x_i \}) \) quantized can be calculated based on Bayes’ theorem. Different frequencies should have different \( \lambda \)’s and \( q_1 \’s.

\[
\begin{align*}
P \{ y_i = 1 | x_i \} &= \frac{\alpha_1 P_1 (x_i)}{\alpha_1 P_1 (x_i) + \alpha_2 P_2 (x_i)} \\
P \{ y_i = 2 | x_i \} &= \frac{\alpha_2 P_2 (x_i)}{\alpha_1 P_1 (x_i) + \alpha_2 P_2 (x_i)}
\end{align*}
\]

Though DCT coefficients are of 63 AC frequencies in total for each channel (Y’, Cb or Cr), we only use low and medium frequencies coefficients in Y channel for the posterior probability estimation. This is because the high frequency DCT coefficients in Y channel and the coefficients in Cb and Cr channels are often quantized to zeros, especially for low compression quality. They do not have enough statistical information for estimation.

B. Segmentation

We first extract DCT coefficients at each candidate frequency to build an “image” of size \((h/8, w/8)\) where \( h \) and \( w \) are height and width of the DCT coefficient matrix of Y channel. And then image segmentation algorithm is performed on the “image” to locate the tampered region based on the estimated posterior probabilities. We can use the posterior probabilities to get the estimated tampered region with simple thresholding strategy; however, this often gets scattered located regions with many noises. Therefore, we adopt graph cut algorithm [13] which explores high order correlations of neighborhood pixels to get concentrated and smooth tampered region. In our segmentation scenario, object means singly quantized DCT coefficients and background means doubly quantized DCT coefficients. The illustration of the graph cut algorithm as shown in Fig. 3.

Note a set of the pixels in the constructed image as \( P \) and a set of all unordered pairs \( \{p, q\} \) of neighboring pixels in \( P \) as \( N \). We define a vector \( A = (A_1, ..., A_p, ..., A_{|P|}) \) as a segmentation. Each \( A_p \) can be either “obj” or “bkg” (abbreviations of “object” and “background”) as a specify assignment to each pixel \( p \). The cost function considering boundary and region properties of the segmentation can be described as follows:

\[
E(A) = \omega R(A) + B(A),
\]
where

\[ R(A) = \sum_{p \in P} R_p(A_p) \] (regional term)

and

\[ B(A) = \sum_{\{p,q\} \in N} B_{p,q} \cdot \delta_{A_p \neq A_q} \] (boundary term) \hspace{1cm} (11)

The parameter \( \omega \geq 0 \) specifies a relative importance of \( R(A) \) against \( B(A) \). According to (9), we set

\[ R_p(\text{“obj”}) = -\ln P(y_p = 1|x_p), \]

\[ R_p(\text{“bkg”}) = -\ln P(y_p = 2|x_p), \]

\[ B_{p,q} \propto \exp \left( -\frac{(x_p - x_q)^2}{2\delta^2} \right) \frac{1}{\text{dist}(p,q)} \] \hspace{1cm} (12)

where \( x_p \) and \( y_p \) are the value of \( p \)th pixel in \( P \) and its corresponding label. \( \text{dist}(p,q) \) is Euclidean distance between \( p \) and \( q \). \( \delta \) is the standard deviation of pixel values. Graph cut algorithm try to find minimum cost cut that gives an optimal segmentation (vector \( A \) ) corresponding to the minimum of energy (10). In (11), \( R_p(\cdot) \) reflects on how the DCT coefficient at location \( p \) fits into the probability distribution of singly or doubly quantized DCT coefficient; and \( B_{p,q} \) is large when pixels \( p \) and \( q \) are similar and decreases as distance between \( p \) and \( q \) increasing.

We construct a graph \( \mathcal{G} = (\mathcal{V}, \mathcal{E}) \) of the image where \( \mathcal{V} \) is set of vertices and \( \mathcal{E} \) is set of edges connecting neighboring vertices. The vertices of the graph represent image pixels (the DCT coefficients). There are also two specially designated terminal vertices \( S \) (source) and \( T \) (sink) that represent “obj” and “bkg” labels, respectively. Neighboring pixels are interconnected by edges in 4-neighborhood fashion. Edges between pixels are called \( n \)-links where \( n \) stands for “neighbor”. Another type of edges, called \( t \)-link, is used to connect pixels to the terminals. All graph edges \( e \in \mathcal{E} \) including \( n \)-links and \( t \)-links are assigned some nonnegative weights which are shown in Table I. Minimum cost cut of the graph can be computed efficiently in low-order polynomial time using max-flow/min-cut algorithm.

If we have \( N \) candidate frequencies, we can get \( N \) segmentation results. All the segmentations vote for the final tampered region localization result.

C. Validation

For the tampered image, we expect its located tampered blocks cluster. Furthermore, the AC DCT coefficients of the located tampered region should obey the distributions of singly quantized AC DCT coefficients which are governed by the estimated parameters. That is, if the probability distribution of DCT coefficients of the located region has the periodicity which coincides with that of the whole image, we consider the localization result is invalidate. Based on this, we can extract three features, including connectivity \( C \) (perimeter–area ratio of the located region), validity \( V \) (correlation between the DCT coefficient histograms of the located region and the whole image) and the proportion of the located region \( A \). We use \( C \) and \( A \) as features since scattered and small located regions are usually behaved as false alarm. These three features are fed into a classifier to decide whether the image is tampered. If the output is positive, then the located region is decided as the tampered region of the image. Otherwise, the located region is a false alarm which should be discarded and the image is accordingly decided as an authentic one.

IV. EXPERIMENTS

A. Datasets

We synthesized tampered image datasets for the evaluation of our proposed method. We used UCID image dataset (all images are in Tiff format) [14] to generate tampered images with different JPEG compression configurations. We first compressed the UCID images with different quality factors, \( Q_1 = 55, 65, \ldots, 95 \), to generate a set of source images which we named as SourceSet. For each image in SourceSet, we randomly chose a region and replaced it with the same content which is from the UCID dataset. Then, we compressed the generated tampered images with new quality factors, \( Q_2 = 55, 65, \ldots, 95 \). Thus, we generated twenty-five tampered image datasets named \( Q55Q55, Q55Q65, \ldots, Q95Q85 \) and \( Q95Q95 \). For each set, it has two subsets according to the size of tampered region which are 10% and 30% of image size.

B. Performance evaluation metrics

Tampered region localization can be considered as a binary classification problem. We denote the pixels in the tampered region as positive samples (\( P \)) and those in the unchanged region as negative samples (\( N \)). To evaluate the accuracy of localization at pixel level, we intuitively define the localization

\[ W_{\text{edge}} \] \hspace{1cm} (weight of edges in the graph \( \mathcal{G} \))

<table>
<thead>
<tr>
<th>edge</th>
<th>weight</th>
<th>for</th>
</tr>
</thead>
<tbody>
<tr>
<td>{p, q}</td>
<td>( B_{p,q} )</td>
<td>{p, q} \in \mathcal{N}</td>
</tr>
<tr>
<td>{p, S}</td>
<td>( \omega \cdot R_p(\text{“bkg”}) )</td>
<td>( p \in \mathcal{P} )</td>
</tr>
<tr>
<td>{p, T}</td>
<td>( \omega \cdot R_p(\text{“obj”}) )</td>
<td>( p \in \mathcal{P} )</td>
</tr>
</tbody>
</table>
accuracy, $ACC_{\text{pixel}}$, as the proportion of pixels that are correctly classified.

To evaluate the performance of our proposed method at region level, we introduce the $F1$-measure [15] which mainly focuses on positive samples including true positive and false positive samples. The performance of a localization method at region level can be evaluated by region accuracy $ACC_{\text{region}}$ which is the proportion of samples whose $F1$-measure $\geq T$. For each tampered image, if $F1$-measure $\geq T$, we consider that our algorithm successfully located the tampered region. Otherwise, even if there is a suspicious region located by our algorithm, our algorithm fails. In the following experiments, we set $T = 2/3$. That is, for a successful case, at least 50% of tampered region is located correctly and the size of falsely located region is at most the same as the size of correctly located tampered region. Thus, $F1$-measure $\geq 2/3$ is a quite strict condition.

For image level evaluation, the ROC curve is adopted. Three features introduced in Section III-C are extracted from each image and fed to a SVM classifier to get a decision result.

C. Experimental results

We run our proposed method on each of the tampered datasets mentioned above and compared it with the method proposed in [10] ², since our algorithm is make the same assumption about the tampering model as [10] which proposed a workable complete solution for tampering detection and localization, and our method is inspired by their work. We selected first 20 AC frequencies in Y channel to locate the tampered region with $\omega = 0.5$ (Table I). The following shows the performances of our proposed method at three different levels based on metrics $ACC_{\text{pixel}}, ACC_{\text{region}}$ and ROC curve, respectively.

1) pixel level: Table II and III show $ACC_{\text{pixel}}$ of our proposed method and the method proposed in [10] on datasets with different tampered region sizes, respectively. $Q_1$ and $Q_2$ are compression factors of standard JPEG compression for first and second compression. The “-” means our proposed method does not suitable for those settings since $Q_1 = Q_2$ indicates both the tampered region and the unchanged region are compressed once. Surprisingly, the $ACC_{\text{pixel}}$ drops obviously as tampered region size increases. We believe this is mainly caused by different ratios of tampered region to unchanged region for different datasets. Even if our algorithm does not locate any tampered region in a testing image, its pixel level accuracy 90% for tampered region proportion of 10% and 70% for tampered region proportion of 30%. There is no need to compare the performances of these two methods when $Q_1 > Q_2$, since their pixel level accuracies are lower than 90%, for tampered region proportion of 10%, and 70%, for tampered region proportion of 30%, which means both two methods always located tampered region falsely. When $Q_1 < Q_2$, our method performs better than the method proposed in [10].

²We implemented it by ourselves since we cannot get the original codes

<table>
<thead>
<tr>
<th>Q</th>
<th>55</th>
<th>65</th>
<th>75</th>
<th>85</th>
<th>95</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ours</td>
<td>68.92%</td>
<td>93.03%</td>
<td>83.12%</td>
<td>85.00%</td>
<td>83.26%</td>
</tr>
<tr>
<td>[10]</td>
<td>69.60%</td>
<td>69.13%</td>
<td>69.71%</td>
<td>79.42%</td>
<td>83.32%</td>
</tr>
</tbody>
</table>

TABLE III

$ACC_{\text{region}}$ for tampered region proportion of 30%

<table>
<thead>
<tr>
<th>Q</th>
<th>55</th>
<th>65</th>
<th>75</th>
<th>85</th>
<th>95</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ours</td>
<td>67.19%</td>
<td>62.95%</td>
<td>59.71%</td>
<td>45.23%</td>
<td>65.62%</td>
</tr>
<tr>
<td>[10]</td>
<td>69.79%</td>
<td>69.86%</td>
<td>67.93%</td>
<td>65.62%</td>
<td>65.62%</td>
</tr>
</tbody>
</table>

2) region level: Table IV and V show $ACC_{\text{region}}$ of our proposed method and the method proposed in [10] on datasets with different tampered region sizes, respectively. From these tables, we can find that our proposed method performs better than the method proposed in [10] in most cases except for some cases when $Q_2 = 95$. Besides, we find that for $Q_1 > Q_2$, both of the two methods fail. The reason should be that when $Q_1 > Q_2$, $P\{y_i = 1|x_i\}$ often smaller than $P\{y_i = 2|x_i\}$ especially for small tampered region (small $a_1$). Thus, most blocks of a testing image are detected as unchanged ones. When $Q_1 < Q_2$, our proposed method performs much better. The reason should be that for doubly quantized coefficients, they have periodically some missing values. If these values appeared in a testing image, they must be from tampered region. However, $ACC_{\text{region}}$ for $Q55Q65$ and $Q65Q75$ are 0% or nearly 0% regardless of tampered region size. We believe this is because that the two consecutive heavy quantizations result in little statistic information for accurately estimation of first quantization step. The bad performance of [10] at region level indicates the high false localization rate of their proposed method.

3) image level: Since both our algorithm and the algorithm proposed in [10] fail when $Q_1 > Q_2$, $Q_1 = 55$, $Q_2 = 65$ and $Q_1 = 65$, $Q_2 = 75$, we do not consider these configurations for image level evaluation. Fig. 4 shows the ROC curves of these two methods. We utilize the SVM with RBF kernel as a classifier in our experiment. It is obvious that our proposed method performs much better than their method.

4) real examples of tampering localization: Fig. 5 shows some successful localization results on our published tampered image dataset, CASIA TIDE v2.0 [16]. We can find that our proposed method can locate the tampered region accurately.
under different tampering manners and tampered region sizes.

V. CONCLUSIONS

In this paper, we have proposed an algorithm which can detect a tampered JPEG image and locate the tampered region accurately when the unchanged region is the output of JPEG decompression. We have utilized the difference of the probability distributions of AC DCT coefficients between the tampered region and the unchanged region as the cue to locate tampering. EM is employed to estimate the proportion of tampered region, first quantization step of unchanged region and the parameter of Laplacian distribution which models the distribution of AC DCT coefficients. After that the posterior probability of each candidate AC DCT coefficient being singly quantized is calculated based on Bayes’ theorem. Finally, we have introduced graph cut to locate the tampered region with the estimated posterior probabilities. The experimental results on the large scale databases have proved the effectiveness of our proposed algorithm for different tampered region sizes at all levels including pixel, region and image level. For further analysis, we may investigate post-processing like morphology operation to improve the localization accuracy of our proposed method. Also, how to overcome the failure cases when the second quantization step is larger than the first quantization step is our future work.

REFERENCES