Chapter 3
Extraction of Scene Text from Video

In this chapter, we present an approach for scene text extraction from natural scene video frame. We assumed that the planar surface contains text information in the natural scene, based upon this, we detect planar surface within the disparity map obtained from pair of video frames using stereo vision technique. It is followed by segmentation or extraction of detected planar surface from other non-text regions. Finally, the text information is extracted from reduced reference i.e. extracted planar surface.

3.1 Introduction

Text information extraction from scene images has been a well-studied topic in the field of computer vision. Text in scene images has been generally defined as text existing naturally in the image. Many researchers achieved substantive and promising results, but this area still facing several difficulties and challenges, thus a sustained research upholds. The more description about scene text, challenges and its applications are presented in chapter 1. Scene text information extraction techniques can be classified into three categories such as CC-based, edge-based and texture-based methods. The survey on the techniques lies within these categories are presented in chapter 1. In this section, we attempted to address the recent techniques developed to extract the scene text from the video sequence.

The main difficulty in scene text extraction process is segmentation part, and most of the earlier approaches had greater challenges and limitations in a complex scene. Therefore, discriminative approaches have been proposed, which are capable to perform segmentation with considerable accuracy. Shahab et al.(2011) proposed a technique to read scene text in ICDAR competition, where the adjacent
characters are grouped together so that candidate image patches are obtained, and these image patches are used to localize text regions. The authors extracted Haar features from both gradient and stroke orientation maps using block pattern method. The extracted features are used to train a classifier based on Adaboost model. The input of the classifier is analyzed to determine text regions, which are later merged into rectangular blocks.

Shivakumara et al. (2011) proposed a two-step Fourier-Laplacian filtering technique. The authors employed a low-pass filter for noise smoothing, while a laplacian mask is used to detect text regions by generating Maximum Gradient Difference (MGD). These procedures are performed in frequency and spatial domains respectively. The analysis of MGD results revealed that text regions had larger values compared to non-text due to larger magnitudes of the positive and negative peaks. The k-means clustering method is used to cluster pixels belonging to text region against those of non-text region.

Huang et al. (2013) presented a Stroke Feature Transform (SFT) method based on the Stroke Width Transform (SWT) in order to address text extraction problems. Lu et al. (2015) designed text-specific features based on contrast, shape structure and paired edges. The approach detects candidate text boundaries with at least one character per boundary is extracted using a local threshold. When all candidate characters have been extracted then refining is done through support vector regression model trained using bags-of-words representation that removes false characters that do not belong to text.

The recent promising directions based on hybrid methods have become the focus of several recent works. They combine the advantages from a number of extraction algorithms. Yi et al. (2009) presented a hybrid method to localize scene text using region and component information. Neighbor component relationship and the unary component properties are used to construct a conditional random field model for connected components. The model parameters are optimized with Minimum Classification Error (MCE) learning and graph cuts inference algorithms.

Other methods such as Maximally Stable Extremal Regions (MSER) based
approaches have recently become the focus. The integration of MSER approaches has demonstrated significant improvements in real-world applications. MSER algorithms are proven better in detecting candidate character features, irrespective of the quality in terms of noise levels and contrast. Yin et al. (2014) proposed a robust and accurate MSER based scene text detection method. The authors explored the hierarchical MSER structure in order to design a pruning algorithm based on simple generated features, where pruning significantly reduces the number of candidate characters to be processed. The authors employed a self-learning algorithm to learn distance weights based on distance metric. These learned parameters and the estimated posterior probabilities of text candidates are incorporated into a character classifier for clustering.

The above-mentioned MSER-based approaches still are insufficient for extraction of text due to the limited number of features contain in the candidate character region. In order to minimize limitations, Iqbal et al. (2014) proposed a method using Bayesian network score obtained through K2 algorithm. This is made possible by establishing causal link on extracted regions’ pixel intensity. The low posterior probability candidate character regions are considered for grouping using selection rules, perceptual grouping filters and repulsion scores from pair-wise filters.

### 3.2 Our Approach

Outdoor images containing sign or advertisement boards, walls, sidewalks, roads, roofs and other objects like vehicles can appear planar when viewed from a distance. Normally, the text information is written on the planar surfaces in order to read and interpret information easily. This motivated us to propose a scene text extraction technique based on detection and extraction of planar surface, which is followed by extraction of scene text within the extracted planar surface.

Many researchers, including Bobick et al. (1999), Corso et al. (2003) and Konomi et al. (2008) have used stereo disparity, which is estimated from stereo images
in order to navigate the mobile robot based on detection of planar objects. The property that planar surfaces can be represented as linear functions in disparity space and thus have a constant spatial gradient (Corso et al., 2003). This property provides a platform for the detection and extraction of planar surface based on the statistical features of the estimated disparity map. Jeffrey et al. (2010) detected planar surfaces by performing Principal Component Analysis (PCA) on a local neighborhood to approximate local surface normal within the sampled points, and Random Sample Consensus (RANSAC) is used to cluster these points into subsets to fit planar model.

Konolige et al. (2008) integrated appearance and disparity information for object avoidance and used AdaBoost to learn color and geometry models for ideal routes of travel along the ground. S. Zhang et al. (2011) address the problem of low efficiency and unsatisfactory matching of uniform texture regions in binocular stereo vision based on a rapid window-based adaptive correspondence search algorithm using mean shift and disparity estimation. They combined color aggregation and local disparity estimation into matching cost aggregation, in order to reduce the color dynamic range of the original image and to make complex pixel regions of texture to become simple uniform texture areas.

In this research work, we proposed an approach for scene text extraction from video frames. As we mentioned, it is assumed that text is contained in planar surface, this drives our interest to identify a region within the video frames which represent a planar surface on the 3D world. In the preceding chapter, we proposed an approach to extract planar surface (saliency region) from natural scene stereo frames, which helps to assess the quality of frames in reduced reference. The approach presented in the preceding chapter comprises three major steps: a) estimation of disparity map using stereo frames, b) detection of candidate planar surface from the disparity space using gradient derivatives and c) segmentation of candidate text block by mapping connected component analysis of homograph image with detected candidate planar surface. The main drawback lies on the segmentation of text blocks based on the mapping of connected component to the disparity map. The method produces low accurate results and contains some
non-planar regions.

Hence, to solve the problems associated with the approach presented in the preceding chapter, we adopted technique proposed by Jeffrey et al. (2010) in order to detect and extract the planar surface from natural scene video frames. After extraction of planar surface from complex background, further processing can be done by considering the extracted planar surface (called as text block) instead of whole image area. The extraction of planar surface from the complex background reduces the complexity involved while extracting the text from complex background by considering the whole image as processing area. To increase the segmentation accuracy, we introduce plane fitting technique by constructing planar model based on a local surface normal computed through PCA and RANSAC. The image labeling is done by employing Markov Random Field (MRF) with Graph cuts algorithm where planar surface is segmented from other regions based on the labels assigned to it. The process is further extended to extract scene text by filtering the extracted text block (planar surface) with Fourier-Laplacian algorithm to generate points, which are classified using k-means as either belongs to text region or non-text region. Figure 3.1 shows the flow diagram of the proposed approach.

![Flow diagram of our approach](image-url)
3.2.1 Stereo Disparity

Given a pair of left and right video frames shown in the Figure 3.2(a) and Figure 3.2(b) respectively, which are rectified using un-calibrated rectification technique (fundamental matrix) from stereo vision toolbox. The rectification reduces the searching area from 2D to 1D along x-axis during matching of corresponding points involved while estimating the disparity map. We estimated the disparity map using a high speed stereo block matching technique based on Sum of Absolute Differences (SAD) Algorithm.

Generally, the disparity offset $D_d$ of a pixel $p_d$ in the disparity map is generated by computing the differences between pixel components in left reference frame $p_l$ and a corresponding pixel components match of right frame $p_r$ as shown in the following equation:

$$p_d = p_l(u_l, v_l) - p_r(u_r, v_r) \Rightarrow D_d,$$

(3.1)

where $\forall u_l = u_r$, $\forall v_l = v_r + i$ and $i$ is the disparity range.

It becomes computationally costly to match every single pixel $p(u, v)$ in a reference frame as there is a requirement to search all pixels in the pre-defined parameter. To reduce these high computations, the ground control points are incorporated into the matching process to cut back the algorithmic complexity and sensitivity to occlusion-cost assigned to unmatched pixels. This manages the biasing process required to smoothen the solution and the task of choosing crucial previous potentialities to describe the image formation (Bobick et al., 1999). This procedure for estimating the disparity map effectively reduces the cost value and search time from maximum disparity time to a relatively lower range where the result of the optimal disparity map achievable at point $p(u, v)$ is given below:

$$D_p = \arg \min_{d \in S_d} \{E(p(u, v), p_d(u, v))\},$$

(3.2)
Figure 3.2: Results of estimated disparity map: (a) rectified left frame, (b) rectified right frame and (c) estimated disparity map
where $S_d$ is a possible range of disparity and $E$ is cost. The $S_d$ can be defined as:

$$S_d = \bar{D}_p \pm K,$$

(3.3)

where $D_p$ is the estimated disparity at point $p$ and $K$ is the disparity estimation threshold.

By a priori knowledge, it is well known that simple textured regions are in the same depth, therefore, this rule is employed to match the points between a pair of frames to reduce computational time spent computing the whole disparity map. The only requirement is to check if the point $D_p$ is within the simple-textured region. A point of reference $p(u, v)$ is inspected to find out if it’s two neighboring pixels $p(u - 1, v)$ and $p(u, v - 1)$ belong to the same texture region supported by their color intensity. It is when solely the corresponding pixels are within the same color with their corresponding disparity indices, $D(u - 1, v)$ and $D(u, v - 1)$ are equal, then only we can assign:

$$D_p(u, v) = D_p(u - 1, v).$$

(3.4)

The computed disparity offsets from the pair of video frames, produces a result that forms homogeneous regions clustered according to the number of pixel difference. The disparity regions are labeled according to the results of component offsets shown in Figure 3.2(c). The labeled disparity map is solely an integer-valued with regions having no smooth transitions. The transitions are then smoothed using a Gaussian filter.

### 3.2.2 Estimation of Gradient Map

The planar boundaries can be detected as discontinuities on the disparity map along $x$ and $y$-axes (Corso et al., 2003). Based on the property that disparity map of a planar surface exhibits a continuing spatial gradient, we estimate and detect candidate planar surface that represents text block. We detect the planar surface
through the identification of characteristic significant region within the disparity map by computing two separate 1D gradient derivatives along the horizontal and the vertical directions. The absolute values from these two gradient maps are added together to approximate gradient magnitude $G'$ (Figure 3.3(a)).

![Figure 3.3: Intermediate results of our approach: (a) detected planar surface within the gradient map and (b) its corresponding contour map](image)

In order to identify the boundary for detected planar surface, the contour map’s magnitude (Figure 3.3(b)) is calculated by segmenting the approximated gradient map. The threshold value $Th$ is employed, where all values above the threshold is considered to be on a boundary. All these values are set to $n_{(e)}$ (a significant negative value) proportional to the area under a convolution kernel $\sigma$. The invalid or unsatisfactory values arising from the reconstruction of gradient map are also set to the same value $n_{(e)}$. These boundary discontinuities comprise
of positively valued pixels separating a wide homogeneous regions. The whole
contour map is convolved with a signum kernel, counting all the valid gradient
values (Corso et al., 2003).

$$\Sigma_t = \sum_{y=v-\sigma_y/2}^{v+\sigma_y/2} \sum_{x=u-\sigma_x/2}^{u+\sigma_x/2} \text{sgn}(G_t(x, y) + 1).$$  \hspace{1cm} (3.5)

During the computation of $\Sigma_t$, the point $(u_\Sigma, v_\Sigma)$ depicting the maximum value
is estimated to define a planar surface’s seed point. The area $\sigma_x \times \sigma_y$ covered by
kernel $\sigma$ is the acceptable minimum size of a planar surface in the frame.

### 3.2.3 Fitting Planar Model

The contour map shows the boundary of planar surface, and it helps in the
extraction of planar surface. In order to extract the planar surface, first, we need
to fit the planar model using RANSAC. We estimated the planar equation of
the detected planar surface using surface normal parameters of the seed point
$(u_\Sigma, v_\Sigma)$ using PCA.

A planar surface becomes linear function mapping pixels from one image to
its corresponding matching image. Let $(x, y, z)$ be a point in the world coordinate
and $(u, v)$ be a point in the pixel coordinates, then a point $w$ with a depth $d$ in
the 3D space from the viewing position can be represented by the plane equation
as:

$$ax_w + by_w + cz_w = d. \hspace{1cm} (3.6)$$

And for a non-zero depth it can be rewritten as:

$$\frac{x_w}{z_w} + \frac{y_w}{z_w} + c = \frac{d}{z_w}. \hspace{1cm} (3.7)$$

The disparity of a planar surface can be estimated by mapping the above
expression into image coordinate as:
\[ au + bv + c = D(u, v), \tag{3.8} \]

where \( u = fl \times \frac{xw}{z_w} \) and \( v = fl \times \frac{yw}{z_w} \), when the camera focal length \( fl = 1 \).

Here, by using 2D video frames captured through the video cameras, it is attainable to estimate the distance of objects based on component offsets. Near objects portray higher component offset value, whereas farther objects have lower component offsets. By analyzing these disparity differences and grouping neighbor pixels with an equal number of offsets, we can get pixels that belong to the same group, except on region points that cannot be viewed by both cameras. These sets of pixels which cannot be captured by both cameras are called occluded regions and can be estimated partially. The computed sub-pixels produce extremely reliable and most correct values, which can be exploited for further processing.

The planar surface that has large homogeneous regions based on constant gradient derivative values have local surface normal estimated throughout the region. This property is used to estimate a local surface normal through PCA on the seed neighborhood with a valid disparity. RANSAC is used to fit a model from an array of 3D points based on the plane parameters.

The main aim is to find large homogeneous regions in a depth disparity map, where we can fit a planar model and is done by generating a number of gradient regions from the disparity map. The regions are labeled as per the gradient magnitude values with plane support points extracted. The plane support points define local regions in the image that support planes of various orientations. We seek to maximally cluster similar labeled support points, which will classify and define the largest spatial region that may correspond to a single plane.

A local surface normal is then computed for every homogeneous region. Every maximum valued point within each labeled regions corresponding to the pixel \( p(u, v, D(u, v)) \) where \( u \) and \( v \) are image axes, whereas \( D(u, v) \) is a valid disparity value of that point and is selected at every iteration as a seed point of the region.

The \( 3 \times 3 \) neighborhood kernel centered on the seed point \( p(u, v) \) is extracted as it
gives a better overview of the region. Using the extracted kernel values, the PCA components' row-wise variances are computed from their corresponding means \( \bar{p}_i \), and the row-wise means are calculated as:

\[
\bar{p}_i = \frac{1}{9} \sum_{j=1}^{9} p(u + j - 1, v),
\]

where \( i = 1, ..., 9 \). The covariance matrix of the point is computed through the variance results using the equation shown below:

\[
COV = \frac{1}{9} \sum_{j=1}^{9} \sum_{i=1}^{9} (p_{i,j} - \bar{p}_i)^2.
\]

A diagonal matrix of generalized eigen values are obtained from the above computed covariance matrix \( COV \). The vector containing the eigen values are further processed to generate eigenvectors. The eigenvectors corresponding to the two biggest eigen values define the normal vector of the local neighborhood. To compute the normal vector \( \hat{n} \) of the neighborhood, we calculate the cross product of the two eigenvectors’ value:

\[
\hat{n} = (\hat{\alpha}, \hat{\beta}, \hat{\epsilon}).
\]

We use RANSAC algorithm to robustly fit a plane to a set of 3D world data points generated from the disparity map, \( p(u, v, D(u, v)) \), where \( u \) and \( v \) are coordinate values and \( D \) is a valid disparity value, A random sample set \( S = 2000 \) points are taken from the disparity map with the inlier distance threshold \( t = 0.05 \), where in every iteration a subset of points marked to support a local surface normal parameter \( \hat{n} = (\hat{\alpha}, \hat{\beta}, \hat{\epsilon}) \) to fit a planar model using the following form of plane equation:

\[
\hat{\alpha}u + \hat{\beta}v + \hat{\epsilon}D(u, v) + \phi = 0.
\]

The sample points \( S \) from the left reference video frame is used to find out if
those points belong to the most supported planar model and if true it is marked as an inlier point \((i)\), where \(\forall i \in S\). When all the sampled points have been tested and all supporting points marked as inliers, they are removed from the sampled points and an array of points are generated from their corresponding values to fit the planar model. The process is repeated for the next planar model with again calculating the local surface normal of the neighborhood of the next largely supported planar model. The process iterates until 80% of sampled points have been fitted or when the RANSAC fails to find a consensus of the sampled points as observed in (Jeffrey et al., 2010). A well supported plane is extracted since RANSAC finds a largest consensus (Fischler et al., 1981), which is further processed by labeling all other pixels supporting these planar models through MRF technique discussed below. Figure 3.4 shows the fitted planar model.

![Figure 3.4: The fitted planar model on the disparity image shown in the Figure 3.2(c)](image)

### 3.2.4 Markov Random Field Labeling

The problem arising is how to assign each pixel with a label that represents some local quantity, such as disparity, for the estimation of MRF, which gives a framework to add prior to the observation. This pixel labeling is naturally
Chapter 3. *Extraction of Scene Text ...*

represented in terms of energy minimization where energy function has two terms: one term to penalize a solution inconsistent with observed data, while the other term to enforce some kind of spatial coherence (Szeliski et al., 2006).

We sought to infer the correct configuration of labels based on the mid-level and high-level representations by leveraging the work done by Jeffrey et al. (2010). The mid-level representation infers the correct configuration of assigning a label either as a planar or a non-planar surface, while the high-level representation seeks to infer the labeled planar pixels into their corresponding candidate planar models. Pixels labeling are based on their orientations through the computation of local surface normal. The initial labeling field is usually not optimal, and it can be possibly improved with the spatial-temporal constraints. This is performed through the energy minimization method based on MRF with Graph cuts technique as an optimization algorithm for pixel labeling.

Optimization in MRF problem is finding the maximum of the joint probability over the graph, usually with some of the variables given by some observed data, or alternatively, it can be done by minimizing the total energy. This in turn requires the simultaneous minimization of all the clique potentials. In addition with the set of prior and likelihood information from the MRF labeling, we estimated a maximum a posteriori by incorporating both of this information into a Bayesian inference:

\[ pb(f|r) = \frac{pb(r|f)pb(f)}{pb(r)}, \]  

(3.13)

where \( pb(f) \) of configuration \( f \) and the likelihood densities \( pb(r|f) \) of the observation \( r \).

Our representation consists of two coupled MRFs to label pixels belonging to the planar surface based on the confidence allocated to each pixel. The pixels that are contained in the image must be labeled according to their configuration. Each pixel must have a neighborhood that will determine the pixel labeling, by the prior identification of planar pixels, which satisfy the planar equation. All other pixel neighbors are tested to see if it satisfies any of the plane equations.
and if true, a label corresponding to that plane candidate is assigned to it.

In the first MRF, the labeling of pixels takes \{0, 1\}, All pixels which have been positively identified belong to a plane are labeled 1 and pixels which do not belong to any plane are labeled with a zero (0). While in the second MRF the number of labels depends on the identified plane candidates thus \( l = \{0, \ldots, m\} \), candidate planes \( c = \{1, \ldots, m\} \) each shall be labeled. The zero is a label assigned to pixels, which fail to belong any candidate plane and thus forms the non-planar regions within the image lattice. We perform foreground (planar) and background (non-planar) separation using energy minimization through \( \alpha \)-expansion of the Graph cuts (Boykov et al., 2001) as shown in Figure 3.5.

![Figure 3.5: Labeling of planar surface from the image shown in Figure 3.2(c) using MRF image segmentation](image)

### 3.3 Text Information Extraction

The text information extraction process of our approach consists of a number of steps, which are employed to segment text regions from the background. The result of previous step consists of two types of regions, which have been positively identified as planar region and non-planar regions. The extracted planar surface result is mapped on the original left reference frame to extract all the features.
The non-planar region features are discarded by assigning a zero value and the results shown in Figure 3.6(a). The image map shown in Figure 3.6(a) contains a number of regions, which comprise of text candidate region (white portion) and two types of background: first, the part we shall name it global background (dark portion), which previously was part of non-planar regions, and the second named local background (orange portion) which is the planar surface portion that encloses the probable candidate text region, but its pixels are not part of candidate text.

We adapted the method proposed by Shivakumara et al. (2011) to highlight and differentiate between text and non-text regions. Because of resultant image map (Figure 3.6(a)) having low-contrast, we convert to gray-scale, then, transform it from spatial domain to frequency domain. An ideal low-pass filter is applied to smooth the noise which forms part of high-frequency components, then a second-order derivative of Laplacian operator is applied, which gives a stronger response to fine details. It highlights the difference between text and non-text regions as its row-wise profile results reveals that text regions have a higher number of positive and negative peaks when compared to non-text regions. The frequent zero crossings arising from positive and negative peaks correspond to transitions between text and background, ideally there should be the same number of text-to-background and background-to-text transitions.

The resultant filtered map is transformed back to spatial domain to extract the maximum gradient difference (MGD) and the result is shown in Figure 3.6(b). The MGD is obtained by moving a local $1 \times n$ window over the filtered image, and computing the differences between the maximum ($M_{x_{val}}$) and minimum ($M_{n_{val}}$) values as shown below:

$$M_{x_{val}} = \max_{\forall t \in [-N/2,N/2]} g(x, y - t), \quad (3.14)$$

$$M_{n_{val}} = \min_{\forall t \in [-N/2,N/2]} g(x, y - t), \quad (3.15)$$
\[ MGD = M_{x_{val}} - M_{n_{val}}. \]  

(3.16)

Text regions have larger MGD value compared to non-text regions due to larger magnitude of the positive and negative peaks. The k-means clustering is applied to classify all pixels into two clusters, candidate text \( CL_1 \) (Figure 3.6(d)) with a cluster mean \( M_1 \), and non-text \( CL_2 \) with the cluster mean \( M_2 \) using the Euclidean distance as shown below:

\[
TextCL = \begin{cases} 
CL_1 & \text{if } M_1 > M_2 \\
CL_2 & \text{Otherwise.}
\end{cases}
\]  

(3.17)

The pixel points from MGD results belonging to text regions are used in the color clustering part to identify the text cluster. We label all pixels within the region of Figure 3.6(e) based on color clustering using k-means classification where they are clustered into two classes: 1) background (non-text) and 2) foreground (text). The two types of background we named local and global are normalized to give a single background overview. The result is binarized and a geometric analysis based on horizontal-vertical aspect ratio is further applied to the binarized image to filter out non-text components, which are clustered as text yet they are not text. Figure 3.6(f) shows the accurate text extraction result using our approach.

### 3.4 Experimental Results

We believe that, this is the first work done directed to building a TIE technique based on stereo frames. Since there is no benchmark dataset, which includes stereo frames of the natural scene with text information. The stereo camera setup consists of two similarly calibrated video cameras positioned horizontally 3-4 inches apart with a frame rate of 29.99 frames per second. The video sequences were captured only focusing on the outdoor scene, where majorally planar objects contain text information. The video cameras are positioned just orthogonal to
Figure 3.6: Intermediate results of our approach: (a) mapping of original left frame with planar surface extracted from the map shown in Figure 3.5, (b) MGD results, (c) candidate text cluster result, (d) candidate text cluster result after morphological operation, (e) text localization and (f) extracted text
the objects of interest. The captured pair of video sequences of the same scene are between the length of 5 to 10 seconds. We created our own dataset from the pair of video sequences, which consists of high-quality stereo frames. The selection of high-quality stereo frames from video sequences was carried out using the approach presented in chapter 2.

The stereo frames selected from video sequence must be of the same scene which contains text information. Our dataset contains one pair or more than one pair of frames per scene object and evaluation of our approach is carried out by giving input as one pair of frames per execution. The stereo frames are high quality and are between $320 \times 270$ and $450 \times 350$ resolutions. In our experiments, we extracted two types of planar orientations namely: 1) large region that has a zero constant gradient value both vertically and horizontally in disparity space, and 2) region that has zero constant gradient value vertically and a varied gradient value horizontally. These regions are assumed to be text-rich areas as they portray uprightness property which characterizes text planar surfaces.

Measuring the performance of text extraction is extremely difficult and until now, there has been no comparison of the different extraction methods (Jung et al., 2004). Since, our work is focusing on text extraction rather than text recognition, we evaluated the performance of text localization process, which is one of the important steps in text extraction. Two metrics were adopted for evaluation, including: 1) recall—the fraction of positives which are detected rather than missed, 2) precision—the fraction of detections, which are positives.

### 3.4.1 Evaluation Metrics

We carried out experiments using dataset, which contains stereo frames with horizontal text and experimental results were compared with other popular text extraction methods. The dataset contains 179 pairs of video frames, which were captured in the outdoor scene with horizontal English and regional Kannada script. In the pair of video frames, only left frame is used for employing other popular methods such as edge-based method (Liu et al., 2005), CCA method.
(Zhong et al., 1995), and gradient-based (Shivakumara et al., 2011). Ground truth is marked by hand on frames of dataset. Given the marked ground truth and detected result by the algorithm, we can automatically calculate the recall and precision. The recall and precision rates have been computed based on the area ratio \( r \) of the bounding box between ground truth and result of our technique as shown in Figure 3.7.

\[
\text{ratio}(r) = \frac{\text{Area}(\text{DetectedBox} \cap \text{GroundTruthBox})}{\text{Area}(\text{DetectedBox} \cup \text{GroundTruthBox})}.
\]  

(3.18)

The following definitions were used for results evaluation process:

**Truly Detected Box (TDB):** A detected box truly detects the text if the area ratio \( r \), defined is at least 50%.

**False Detected Box (FDB):** The false detected box detects the text if the area ratio \( r \), defined is less than 50%.

**Ground Truth Box (GTB):** manually marked the text box by hand on test samples.

\[
\text{recall} = \frac{\#TDB}{\#GTB},
\]  

(3.19)

\[
\text{precision} = \frac{\#TDB}{\#TDB + \#FTB},
\]  

(3.20)

\[
\text{f - measure} = \frac{2 \times \text{recall} \times \text{precision}}{\text{recall} + \text{precision}}.
\]  

(3.21)

The Figure 3.8 shows the result of our approach for extraction of planar
Table 3.1: Comparison of text localization results using recall, precision and f-measure

<table>
<thead>
<tr>
<th>method</th>
<th>recall</th>
<th>precision</th>
<th>f-measure</th>
</tr>
</thead>
<tbody>
<tr>
<td>edge-based</td>
<td>0.67</td>
<td>0.64</td>
<td>0.65</td>
</tr>
<tr>
<td>CC-based</td>
<td>0.69</td>
<td>0.73</td>
<td>0.71</td>
</tr>
<tr>
<td>gradient-based</td>
<td>0.82</td>
<td>0.59</td>
<td>0.69</td>
</tr>
<tr>
<td>proposed</td>
<td>0.96</td>
<td>0.93</td>
<td>0.94</td>
</tr>
</tbody>
</table>

surface from video frames of natural scene. The natural scene is synonymous with complex background. The experimental result shows that the proposed method detects and extracts the planar surface accurately. Table 3.1 shows the recall, precision and f-measure obtained for stereo frames of our dataset using existing methods and proposed method. The results shown in the Table 3.1 are the average value of recall, precision and f-measure computed for all the stereo frames of dataset. The proposed method has the highest recall, precision and f-measure when compared with the existing methods. The experimental results show that the proposed method achieves high accuracy for scene text extraction for the video frames with complex background includes trees, building and other objects.

Figure 3.9 shows the visual comparison of text localization results of our approach with other popular existing methods for the stereo frames of our dataset. The connected component method locates the text area partially. This method locates the whole text area for the first frame, but includes background partially. The gradient based method locates the text area and part of background with many false positives. Edge based method failed to locate text area precisely, because scene images are rich in text like features such as tree-leaves, which produces many edges and are classified as text. The second row of Figure 3.9(c), which is an edge based result, is just visually same as the original image, due to the overlapping of the bounding boxes used to enclose text area. Figure 3.10 shows intermediate results of edge-based method shown in the second row of Figure 3.9(c).
Figure 3.8: The results of planar surface extraction obtained by our approach on video frames of our dataset: first column-original left frames and second column-extracted planar surface.
Figure 3.9: The comparison of text localization result on our dataset: first row-original left frames, second row-results of connected component method, third row-results of edge-based method, fourth row-results of gradient-based method and fifth row-results of our approach
The proposed method initially extracts the planar surface as the first part of text localization (Figure 3.8) and thereafter proceeds to process these regions by employing Fourier-Laplacian technique to locate the text area accurately and precisely. The Figure 3.9(e) shows the text localization result of proposed method for video frames of the natural scene captured in a complex background. The proposed method locates all the text area correctly for third frame, but it shows the one false positives and two true negatives for the first and second frame respectively. The true negative for the second frame is due to occlusion of tree-leaves. The proposed method successfully locates the text area of the third frame, even though text is slightly oriented. The limitation of the proposed method is that it locates the supporting poles of the sign board, this due to fact that while extracting the planar surface it considers the supporting poles as part of planar surface.

Figure 3.10: The text localization results from edge based method: (a) vertical edge results of the image shown in Figure 3.8(a), (b) morphological operation results, (c) text area localization based on bounding box and (d) text localization
Figure 3.11: Sample results of text extraction for slightly oriented text by our approach: first column-original left frames and second column-corresponding extracted text information
3.4.2 Discussion

The literature survey reveals that majority of the images or video frames used in experiments of TIE technique are having text information, which is occurred in the image from border to border. This means that the images or video frames were captured very close to the camera. Because of this property of the image or video frames, the edge or CC-based techniques perform well for these types of images. In case of our dataset, the edge-based or CC-based techniques show poor performance. This is because our dataset consists of frames, which were captured far away from the camera but at a considerable distance, and text is not spanned from border to border. Further, another complexity found in our dataset is the background includes trees, pathways and buildings. The experimental results were evaluated using recall, precision and f-measure. Our approach achieves good performance compared with the existing methods for our dataset. Based on the experimental results, we can conclude that our technique has excellent accuracy for localization evaluated through recall, precision, and f-measure.

Figure 3.11 displays sample results of text extraction by our approach for slightly oriented text information contained in video frames. The text extraction results for slightly oriented text demonstrate that our approach can be employed for both horizontal and slightly oriented text. The Figure 3.12 illustrates more extracted text results. It can be seen from the results that, most of the text is well detected and localized despite variations in character font, size, color, texture, script and orientation. We obtained good results for the video frames with complex background consisting mainly trees and some other non-planar objects, which were easily labeled as non-text planar properties.

Figure 3.13 illustrates some failure examples. In the first frame, the text is not localized accurately. The reason is that it contains multi-colored text, during color clustering it considers and cluster the text with high-contrast color, and text having low-contrast color are treated as background at binarization stage. The false alarm in second frame is due to low contrast, and text does not appear properly because the frame was captured too far away from the camera.
The proposed approach was implemented using MATLAB and good results obtained. The whole process of building TIE vision system using outdoor scene image dataset took much time to complete on a PC with a 2.93 GHz core 2 duo processor and 2 GB memory. Among our experiments, the worst experimental result was due to dominance of certain colors as black and white. This made it difficult to generate pixel offsets as all the color channels had similar values and cannot give a reliable significance. The weakly textured planar surface also makes it hard for the algorithm to match points. The object targeted must be considerably not too close/far from the camera in order to produce desired results.
3.5 Chapter Summary

We presented a TIE technique for extraction of scene text from stereo frames acquired in the natural scene. We achieved a high-accuracy rate both in planar surface extraction and text area localization. We performed planar surface extraction using MRF with Graph cuts technique by labeling regions based on estimated planar parameters. The text block (planar surface) is further processed to extract text by applying second derivative of Fourier-Laplacian to generate text edges and classifying it using k-means clustering based on computed MGD values. The low false alarm rate ensures that our approach extract more accurate text information in order to recognize the text properly. The fact that, our dataset is captured far away from the camera, and our approach does not require any training of samples. Therefore, our approach can be utilized for TIE applications without any further effort. The main limitation of our approach is that, it is more expensive in terms of computation time. However, our approach can extract the text properly in a complex background, and it resembles a human visual system.