A New Approach to Picture Quality-independent Robust Text Binarization using Differential Binning

The Overview of the steps involved

1. **Foreground – Background Pixels split**: The first step involves identifying and splitting the pixels of the image into sets of candidate foreground and candidate background pixels. We used a differential binning approach which gives different sets of candidate foreground and background splits. This is the critical step as the next set of steps depends on the fact that there is at least one split where the text pixels fall into one set of pixels and with very little background pixels in the same set.
2. **Homogeneous region detection:** Step 1 gives several sets of candidate foreground and background pixels. This step involves identifying clusters of connected pixels which represent a homogenous region in the image. A homogeneous region is defined as a set of connected pixels in an image whose deviation in RGB values is less than the deviation from the surrounding pixels of that cluster.
3. **Text, non-Text Separation:** The Homogeneous regions extracted in the form of connected set of pixels from Step 2 are now classified into text and non-text using a classifier model. Regions classified as Non-Text are removed and candidate Text regions are retained for combination
4. **Probabilistic Combination:** Step 3 gives us multiple sets of candidate Text pixels across multiple bin sizes used in the differential binning step. We identify the probability of a pixel being a text pixel based on its occurrence across the different bin sizes. We label Pixels in the final binarized image based on a probability outcome
5. **Foreground – Background Pixels split: High Frequency Binning**

The first target is to identify sets of foreground and background pixels and split them. We use two-level overlapped binning and Adjacent Bin Recombination

1. **Two-Level Overlapped Binning**

For a given Bin Size s, we divide the entire range of rgb Euclidian distances into two levels of Bins

*For a given Bin size s:*

**Level 1** *(Bin number 1 to k)*

*= { 0 to (s-1), (s) to (2\*s – 1), (2\*s) to (3\*s – 1) ,*

*……*

*(k-1)\*s to (k\*s) }*

**Level 2** (Bin number k+1 to 2\*k -1)

*= {(0.5\*s) to (1.5\*s – 1), (1.5\*s) to (2.5\*s – 1),*

*…...*

*((k-1.5)\*s) to ((k-0.5)\*s) }*

*Where k = upper(442/s)*

Each Bin is represented as a binary image which is the kth bin for bin size S and ()ij is the value of the ith row and jth column of

= 1 *if k < upper(442/s) &*

*(k-1)\*s < PIXEL-VALUE(i,j) < k\*s*

()ij  = 1 *if k > upper(442/2) &*

*(k-1.5)\*s < PIXEL-VALUE(i,j) < (k- 0.5)\*s*

= 0 *Otherwise*

PIXEL-VALUE(i,j) =

R(i,j) = Red Value of pixel at ith row and jth column of image

G(i,j) = Green Value of pixel at ith row and jth column of image

B(i,j) = Blue Value of pixel at ith row and jth column of image

In order to achieve robustness and quality independence, we use Decremented Interval Binning– using multiple bin sizes with the difference between corresponding bin sizes decreasing the higher we go. At this stage, let it be clear that when we say foreground or background pixel, it is for convenience and we do not label any specific set of pixels as foreground or background. Any one of the sets can correspond to either foreground or a background the next steps will do the work of eliminating the sets of background pixels.



Recombined Bins Flowchart. The Bin Size denotes the size of the original Bins and the Range is the new size after the Adjacent Bin Recombination step

The lines connect subset bins to their superset bins. For example, Bin (374 to 406) is a subset of Bin (367 to 420) and is connected by a line. Later, this relationship will be used to determine stable regions and extract features for text non-text separation.

1. **Adjacent Bin Recombination**

Consider a Homogenous region Ri with PIXEL-VALUE in the range of 30 to 60. We want all homogenous regions that are less than size s to fall into at least one of the bins of size s. If suppose bin size is 32 we want it to fall entirely in one of the bin. However, Let’s look at the pixel values that the first few bins of both the levels in two-level binning accumulates

Level 1

{(0 to 31), (32 to 63), (64 to 95), …. }

Level 2

{(16 to 47), (48 to 79), …… }

We see in both the levels, the region Ri fails to fall into any of the bins despite having a pixel value deviation less than the particular bin size.

To solve this problem, we create a new set of Recombined Bins called as Δ bins

**LEVEL 1**

**LEVEL 2**

Where, k = upper(442/s)

Considering the previous example, Let us see the first few bins for bin size 32