FRUIT FLOWER DETECTION USING HYBRID GENETIC BASED AUTO-ENCODER CONVOLUTIONAL NEURAL NETWORK (HGAECNN)

Dr. M. Nester Jeyakumar & Dr. Jasmine Samraj

Assistant Professor, Department of Computer Science, Loyola College, Chennai, TN, India.
Associate Professor, Department of Computer Science, Quaid-E-Millath Government College for Women (A), Chennai, TN, India.
nesterjk@gmail.com, jasminesamraj@gmail.com

ABSTRACT: On trees, process of counting and detecting fruits count defines the task of crop estimation. Bloom intensity which corresponds to flowers count present in orchard guides critical crop management decisions in the production of fruits. Even with high importance, human visual inspection is used for estimating bloom intensity. Hand engineered methods forms base for the automated computer vision system used for identifying fruits.Under specific conditions only, their performance is defined and it is a limited one. Time complexity of median filtering is high and after removal of noise, located spatial features are not extracted.In this work proposed a, Hybrid Genetic based Auto-Encoder Convolutional Neural Network (HGAECNN), Local Binary Pattern (LBP) and Improved Median Filtering (IMF) algorithms. There are four main stages in this work. They are, deep FCN network training, pre-processing, extraction of feature, reduction of dimensionality and segmentation. On commercial GPU, deep FCN is used for high-resolution evaluation. GPU memory space is needed for computation of fully convolution and based on resolution of image, this requirement increases exponentially. For, reducing or removing image noises, introduced a Improved Median Filtering (IMF) in the next stage. For extraction of feature, LBP is used in third stage. For all pixels of image, it needs to be computed. According to LBP histogram distribution shape, computed texture regularity. In a sliding window manner, smaller patches are formed by splitting high resolution image. From original dataset, pertinent features subset are computed using a process called ILDA with some selection parameter. For AECNN hyper-parameters values are optimized for maximizing segmentation accuracy of genetic algorithm model. A HGAECNN is used for evaluating every patch. For final segmentation task, on obtained score-maps, refinement algorithm is applied. On images of AppleA and AppleB flowers dataset, conducted experimentation. Metrics like Intersection-over-Union (IoU), F-Score(F1), Recall(R) and Precision(P) are used for measuring results.

KEYWORDS: Graphics Processing Unit (GPU), Hybrid Genetic based Auto-Encoder Convolutional Neural Network (HGAECNN),Improved Linear Discriminant Analysis (ILDA),Improved Median Filtering (IMF),Fully Convolutional Network (FCN).

1. INTRODUCTION

For producing better results and high productivity with few amounts, automated yield prediction technology must be incorporated with crop management system. Yield estimate can be used for prediction and important information is given by them. Prescription maps can be created using yield estimates and in tree intensive applications, it is used [1].

Most of the farming works relies on fruit planters manual labor, in fruit agricultural production. Huge energy and time are consumed by great simple quantity and repetitive labors. This increases cost of production and to agricultural production, it brings more uncertainties. In farming judgment, many mistakes are committed by fruit farmers because of knowledge lack and experience [2].

Relationship between fruit quality, load and bloom intensity are established in various studies. In thinning and pruning process, guidance given by information of bloom intensity and climate are very crucial. Fruit taste, coloration, size and load are directly affected by this. Packing houses may be benefitted by bloom intensity's accurate estimation [3].

During early growing season, in orchard, presented flowers count defines bloom intensity. In thinning and pruning process, guidance given by information of bloom intensity and climate are very crucial. Fruit taste, coloration, size and load are directly affected by this. Packing houses may be benefitted by bloom intensity's accurate estimation. Postharvest handling and storage processes optimization are contributed by early crop-load estimation [4].

Albeit different research endeavors have been made in this field, challenges despite everything stay for complex scenes with shifting lighting conditions, low differentiation among leafy foods, closer view impediments and jumbled foundations. The greater part of these applications have been to discover the organic products bloom for

programmed gathering. An as of late new bearing is to discover the natural products for plant rearing purposes to consequently perceive, tally and measure the organic product bloom so as to evaluate the distinctions in nature of the hereditary material. At the point when the estimations are made by a PC, this is regularly alluded to as advanced phenotyping and the field is developing in significance [5].

In image processing field, technological developments are used in industries and other fields for carrying out the analysis. fields like space science, crop industry restorative uses, and soil inquire about use picture handling innovation other than computerized checking systems. The consequences of computerized tallying are cheap, fast solid and simple when contrasted with manual checking [6].

A CNN extracted feature based apple flower detection method is proposed which is inspired by effectiveness of Convolutional Neural Network (CNN) in various computer vision tasks.

In agriculture applications like plant identification from leaf vein patterns, crop classification and quantification of fruit, CNN architectures are adapted in recent works [7]. In those works, apple flowers are detected using combination of classification network and super-pixel-based region proposals. Intrinsic to super-pixels segmentation accuracy and architecture of network are the limitations of those methods.

Images acquired in-place by different camera devices are used in detectors and usingGraphical Processing Units (GPU) these image are processed by a real-time hardware and software system instead of collecting physical samples like plants and leaves and laboratory based analysis. Various task complexities like variation I background contained in plant's surrounding area, object size and illumination condition are dealt effectively using this method [8]. With further machinery, state-of-the-art results are exceeded by pixels-to-pixels or end-to-end trained Fully Convolutional Network (FCN) on semantic segmentation.

In this work proposed a, Hybrid Genetic based Auto-Encoder Convolutional Neural Network (HGAECNN), Local Binary Pattern (LBP) and Improved Median Filtering (IMF) algorithms. There are four main stages in this work. They are, deep FCN network training, pre-processing, extraction of feature, reduction of dimensionality and segmentation. In section 3, four main steps of this work are discussed.

In section 2, for fruit flower detection in plants, review of various methods of deep learning are presented and section 3, deep-learning-based system is proposed for detection of fruit flower.

2. LITERATURE REVIEW

Dorj and Lee [9] proposed a shading identification and checking calculation for recognizing white shading blossoms in Tangerine tree and tallying Tangerine organic product blossoms to yield better yields as to the current plans. Gaussian channel is utilized to lessen undesirable commotion in Tangerine tree blossom acknowledgment for Tangerine yield mapping framework. It is seen that the recently evolved technique gives better legitimate yield for tangerine tree bloom identification in common open air lighting, with various lighting condition with no elective lighting source to control the luminance. It is critical to bring up that the recently presented technique gives better yield of tangerine tree blossom location, found the productive and compelling contrasted with other existing strategies.

Hočevar et al [10] detected apple fruit flower using an automated tree Flower Clusters (FC) measurement system. In apple fruit production, for high quantity and quality, it is needed to have Tree-specific management practice related to individual tree based on its physiological condition. In a high density apple orchard, for estimating individual trees FC count, Hue, Saturation, Luminance (HSL) images analysis based apple flowering abundance detection is used. Still camera is used for acquiring images in day time and industrial color camera is used in night time. The HSL thresholding is included in FC estimation algorithm with parameter optimization. Hypothetical spraying was done by on/off criterion>100 FC per tree, if industrial camera is used to acquire image acquisition in daytime and identified around 10% incorrect execution. Industrial camera or still camera is used for producing better performance in FC counting.

Diago et al [11] proposed a strong machine vision approach for blossoms, blooming and natural product set are key determinants of grapevine yield. Building up a basic, modest, quick, precise and hearty machine vision approach to be applied to Red, Green, Blue (RGB) pictures taken under field conditions, to gauge the quantity of blossoms per inflorescence naturally. Ninety pictures of individual inflorescences of Vitisvinifera L. cultivars Tempranillo, Graciano and Carignan were procured in the vineyard with a pocket RGB camera preceding blooming, and used to create and test the blossom checking calculation. Solid and critical connections, with R2 above 80% for the three

cultivars were seen among real and robotized estimation of inflorescence bloom numbers, with an accuracy surpassing 90% for all cultivars. The created calculation demonstrated that the examination of computerized pictures caught by pocket cameras under uncontrolled outside conditions had the option to consequently give a helpful estimation of the quantity of blossoms per inflorescence of grapevines at beginning times of blooming.

Dias et al [3]detected apple flowers using a deep learning techniques based novel Convolutional Neural Network (CNN) algorithm. For computer vision applications, state of the art are represented using this technique. Flower sensitiveness is given to Pre-trained CNN by fine-tuning it. In contrast to existing techniques, morphological and color information are combined effectively by proposed CNN using extracted hierarchical features, which leads to performance enhancement to all considered cases. Existing techniques are color analysis based and their application to conditions with illumination or occlusion level change are limited. Four various datasets namely, Peach, AppleC, AppleB, AppleA are used in experimentation. Even with various species of flowers and illumination, better performance is exhibited by proposed CNN-based model with accurate identification. Around 80% of precision and recall values are produced for new datasets without training.

Wang et al [12] proposed a Support Vector Machine (SVM) for mechanized blossoming evaluation framework for mango plantations. Division of blossoms from an unpredictable foundation (for example leaves, branches and ground) was accomplished dependent on (I) shading amendment through change of the brilliance and differentiation to a reference level, to correct the light changeability spatially inside and between pictures; (ii) shading thresholding with fixed edges to isolate blossoms, in spite of the fact that with certain branches and trunks; and SVM based grouping with Speeded Up Robust Features (SURF) to distinguish bogus division, (iii) SVM order to refine the division results, expelling the branch and trunk mistakes. Mango tree shelter pictures (n=160) were gained during a five-week blossoming period, with 15 of the pictures utilized in adjustment and 145 utilized in approval. The proposed strategy had a decent connection with human scoring, with coefficient of assurance (R2) of 0.87.

Wang et al [13] proposed a quicker Region-based Convolutional Neural Networks (quicker R-CNN) of mango plantation blooming includes identification of an inflorescence (a panicle) with blossoms at different phases of improvement. Two frameworks were embraced differentiating in camera, enlightenment equipment and picture preparing. The picture preparing ways were: (I) shading thresholding of pixels followed by SVM grouping to gauge inflorescence related pixel number (panicle territory), and panicle zone comparative with all out covering zone ('blooming power') utilizing two pictures for every tree ('double view'), and (ii) a quicker R-CNN for panicle identification, utilizing either 'double view' or 'multi-see' following of panicles between sequential pictures to accomplish a panicle check for each tree. The relationship coefficient of assurance between the machine vision blooming power and region gauge (way I) and in field human visual tallies of panicles (past 'asparagus' stage) per tree was 0.69 and 0.81, while that between the machine vision (way ii) and human panicle tally per tree was 0.78 and 0.84 for the double and multi-see discovery draws near. The utilization of such data is shown in setting of (I) observing the hour of pinnacle blooming dependent on rehashed proportions of blossoming force, for use as the beginning date inside warmth total models of natural product development, (ii) recognizable proof and mapping of early blooming trees to empower specific early reap and (iii) investigating connections among blooming and organic product yield. For the ebb and flow plantation and season, the relationship coefficient of assurance between machine vision assessments of panicle region and multi-see panicle check and natural product yield per tree was poor.

Dias et al [14] identified flowers using an automated method. This method is robust against uncontrolled environments and various flower species can be identified. End-to-end residual CNN forms the base for this proposed method and in semantic segmentation, state-of-the-art is represented by it. Apple flower images single dataset is used for fine-tuning this network for enhancing sensitivity of this method to flowers. In order distinguish individual flower instances in a better manner, refinement method is employed as coarse segmentations are produced by CNN. Broad applicability and robustness of proposed method is demonstrated using experimentation on pear flowers, peach and apple images which are acquired in various conditions. It does not require ant dataset specific training or pre-processing.

Chen et al [15] predicted yield with minimum time cost and labor using Region-based Convolutional Neural Network (R-CNN) automatic strawberry flower detection system. In this system, two variety near-ground images namely, Radiance and Sensation are captured using a small Unmanned Aerial Vehicle (UAV) developed by DJI Technology Co., Ltd., Shenzhen, China which is equipped with an Red, Green, Blue (RGB) camera at two various heights of 2 and 3 m with orthoimages of a 402 m^2 strawberry field. The Pix4D software is used to process orthoimages automatically. For detection of deep learning, it is split into sequentialpieces. For counting and detecting immature strawberries, matures strawberries and flowers count, a state-of-the-art deep neural network

model called faster Region-based Convolutional Neural Network (R-CNN) is adapted in this system. At 2 m height, for all detected objects mean Average Precision (mAP) is 0.83and at 3 m height, it is about 0.72. Strawberry flowers accurate count can be computed using this system and using this count, future yield can be forecasted as well as distribution maps can be constructed which tends to assist the farmers for observing strawberry fields growth cycle.

Koirala et al [16]estimated mango fruits flowering using a two-stage deep learning framework. With a single and a two-stage deep learning framework (YOLO and R^2CNN), undertaken the panicles classification into three developmental stages via rotated or upright bounding boxes. Models R^2CNN-upright, R^2CNN(-rotated), YOLOv3-rotated, MangoYOLO-rotated, MangoYOLO(-upright) are used in image set validation and counting total panicle. The R^2 for machine vision is used for counting panicles in tree using various camera and involving various cultivar and orchard images. Well generalization is exhibited by this model and there is no consistency in using rotated over upright bounding boxes. Better results in counting total panicle can be achieved using YOLOv3-rotated model and classification of panicle stage can be done effectively using R^2CNN-upright model. For demonstrating practical applications, for 994 trees orchard, weekly panicle count is made with the application of peak detection routine to document multiple flowering events.

3. PROPOSED METHODOLOGY

This work Figure 1, proposes a Hybrid Genetic based Auto-Encoder Convolutional Neural Network (HGAECNN), Local Binary Pattern (LBP) and Improved Median Filtering (IMF) algorithms. There are four main stages in this work. They are, deep FCN network training, pre-processing, extraction of feature, reduction of dimensionality and segmentation. On commercial GPU, deep FCN is used for high-resolution evaluation. GPU memory space is needed for computation of fully convolution and based on resolution of image, this requirement increases exponentially. Neighboring information are used for spotting noises in images and they should be reduced or removed. Without affecting quality of an image, better filtering method is used for removing it and reinforce image smoothness. So, for, reducing or removing image noises, introduced aImproved Median Filtering (IMF) in the next stage.For extraction of feature, LBP is used in third stage. For all pixels of image, it needs to be computed. According to LBP histogram distribution shape, computed texture regularity. In a sliding window manner, smaller patches are formed by splitting high resolution image. From original dataset, pertinent features subset are computed using a process called ILDA with some selection parameter. For AECNN hyper-parameters values are optimized for maximizing segmentation accuracy of genetic algorithm model. A HGAECNN is used for evaluating every patch. For final segmentation task, on obtained score-maps, refinement algorithm is applied.

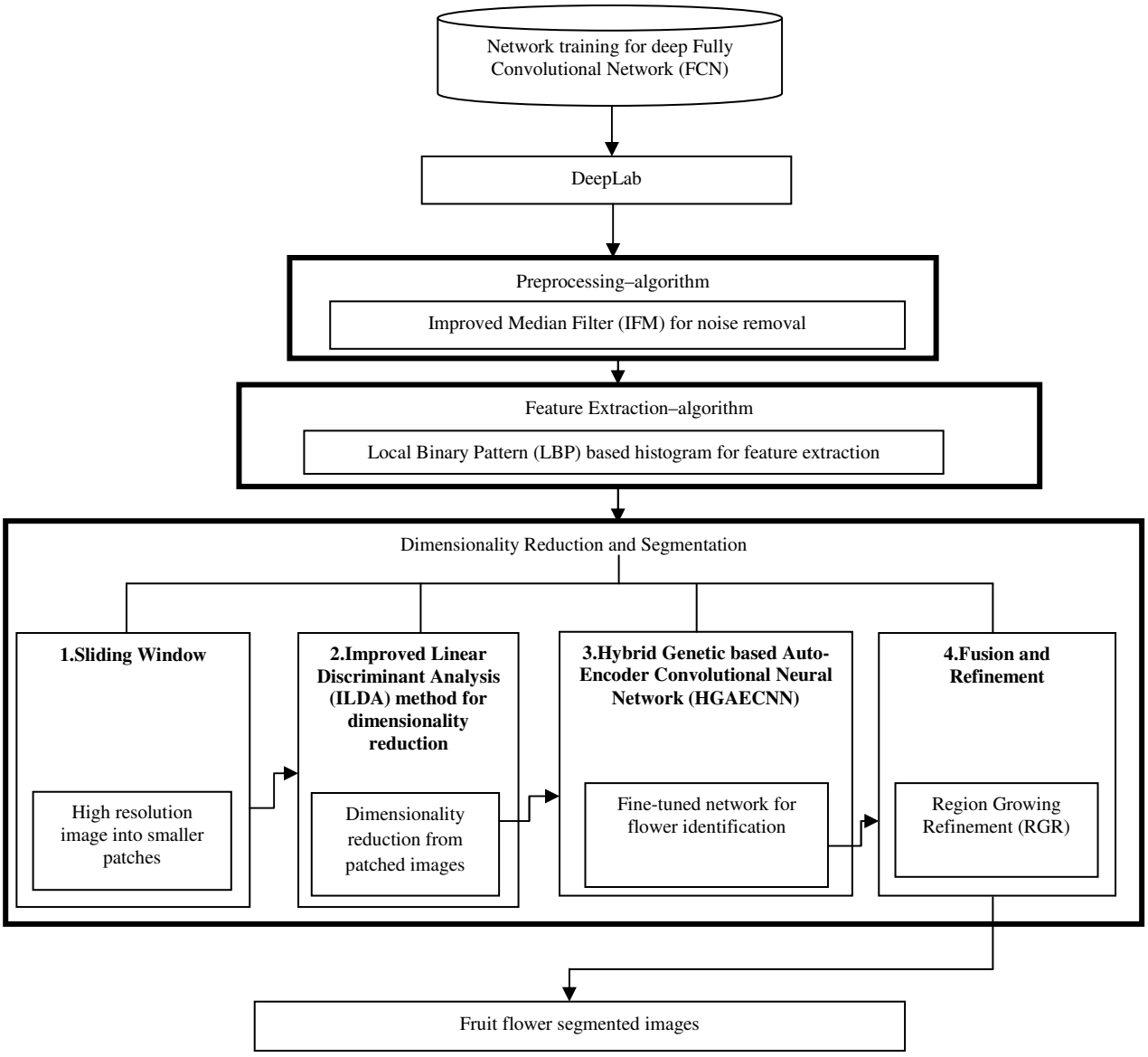

Figure 1: PROPOSED ARCHITECTURE OF FRUIT FLOWER DETECTION

A. Network training for deep Fully Convolutional Network (FCN)

For segmentation of binary flower, this work used DeepLab. In this fine-tuning and network surgery procedures are performed. In trees, undesired branch pruning corresponds to surgery procedure. For interested class segmentation, responsible branches with connections and weights are preserved out of all original branches. Segmentation of fruit flower branch with sigmoid classification unit are used for preserving architecture.

Unbounded the transferred flower branch generated score and saturation of sigmoid happens in an easy manner, without model's original softmax layer induced normalization. Model tuning with two-branches is adopted for alleviating the difficulties in learning which are produced by poor initialization. This is adopted based on the fact that, foreground branch generated predictions can be normalized properly by allowing the network to learn representation of background via second branch. For flower segmentation, major source for misclassification is nearby leaves as indicated in experimentation.

With the application of pre-trained model to training dataset, highest activations are presented for class leaf predictions. Because of this, for initializing two-branch flower segmentation network, branch together with one associated with flowers are adopted. Training set which is shown in Table 1 is used for fine-tuning adapted

architecture. There are 100 apple tree images in training set. For 10,000 iterations, this procedure is performed for this experimentation using Caffe framework [17], with 10^{-4} as initial learning rate and it has $10^{-4} \times \left(1 - \frac{i}{10000}\right)^{0.9}$,polynomial decay. Where, iteration number is represented as i. Procedure of model pre-training is used in model fine-tuning, where every training portrait is evaluated at $(0.5, 0.75, 1{:}\,0, 1.25, 1.5)$ times its original resolution.

TABLE 1: DATASETS SPECIFICATIONS

Dataset	No. images	Weather	Background panel	Camera model	Resolution	Camera support
AppleA	100 (train)+30 (val)	Sunny	No	Canon EOS 60D	5184×3456	Hand-held
AppleB	18	Sunny	Yes	GoPro HERO5	2704×1520	Utility vehicle

The 100 training images are split into 321×321 pixels using original network parameterization, which corresponds to a total of 52,644 training portraits after augmentation.

B. Preprocessing

In images, original pixel values are modified to various intensity values by noises. From an image, reduction or removal of noises are done using the process called noise removal algorithms. It is essential to have image smoothening methods for noise removal and in most of image processing applications, standard or best filters are used for the same. For image noise processing, de-noising of an image is an asset which includes various filtering methods of image de-noising. Various algorithms are used for solving the same. Neighboring information are used for spotting noises in images and they should be reduced or removed. Without affecting quality of an image, better filtering method is used for removing it and reinforce image smoothness.

1. Median Filtering

In digital image processing, an order-statistics filtering non-linear method is termed as Median filter. Impulse noises are commonly removed using a Median filter. In a signal, it runs in a cell by cell manner and replaces every cell value using median of intensity values of neighboring pixels with mathematical accuracy. Across the entire image, this method applied. So, both noisy and non-noisy pixels are transformed. In an image, presented fine details may be removed because of de-noising. This is done at the cost of blurred as well as distorted features possessed in filtering method. Due to impulse noise reduction ability and good edge preservation characteristics, in digital image processing, it is widely used [18].

Median value of neighborhood is used for replacing digital sequence or images noisy value. Based on gray levels, mask pixels are ranked. Noisy value is replaced by storing group's median value. $g(x,y)\ \text{med}\{f(x-i, y-j), i, j \in W\}$ is the output of median filtering, where, original image is represented as $f(x,y)$, output image is represented as $g(x,y)$, two-dimensional mask is represented as W, size of the mask is $n \times n$(with n as an odd value) like $3 \times 3, 5 \times 5$. Cross, circular, square, linear shaped mask can be used [19]. For image having random noise, mathematical analysis of median filtering is a complex one because of its non-linear nature. Under normal distribution, image having zero mean, median filters noise variance can be approximated as,

Noise-reducing Performance of the Median Filter

For image having random noise, mathematical analysis of median filtering is a complex one because of its non-linear nature. Under normal distribution, image having zero mean, median filters noise variance can be approximated as,

$$\sigma_{med}^2 = \frac{1}{4nf^2(n)} \approx \frac{\sigma_i^2}{n + \frac{\pi}{2} - 1} \cdot \frac{\pi}{2}$$

Where, input noise power is represented as σ_i^2, median filtering mask size is represented as n, noise density function is represented as $f(\bar{n})$. Average filtering noise variance is represented as,

$$\sigma_0^2 = \frac{1}{n}\sigma_i^2$$

In reduction of random noise, better performance is exhibited by median filtering when compared with average filtering performance. But less effective performance is exhibited on impulse noise of narrow pulse with less than n/2 pulse width. If average filtering algorithm is combined with median filtering algorithm, then its performance can be enhanced and based on noise density, mask size is varied adaptively [20]. Filtering mask's shape and size defines median filter noise-reducing effects and computation of median value defines complexity of algorithm. So, for enhancing median filter's noise-reduction performance, Improved Median Filter (IFM) is developed.

2. Improved Median Filtering(IFM)

Improvement of the Filtering Mask

Mostly n × n cross or square mask is used as filtering mask. In order to have mask symmetry, commonly odd number is used for n. Better retaining of image details are preserved using small mask size and with huge mask size, noise reduction performance is reduced and retaining of image details are reduced by huge mask size and noise reduction performance enhanced [21]. Adaptive filtering algorithm is introduced for solving contradiction. Based on noise levels in mask, mask can be resized adaptively in filtering process. In mask, gray levels maximum value is represented as max and minimum value is represented as min, average value is represented as average, median value is represented as med and central value is mask is represented as f(i,j) and mask size is represented as n.

Two steps are needed in adaptive filtering method,

Step 1: Mask is resized adaptively,

a. Initialization Phase: let $n = 3$;

b. Computation Phase: $A1 = med - min, A2 = med - max$

c. Judgment phase: if $A1 > 0$and $2 < 0$, then go to step 2; if not, then mask size is enlarged, let $n = n + 2$.

Step 2: median filtering.

Improvements of the Median Filtering

For filtering random noises, better performance is exhibited using average filter. Certain filtering mask size is used for combining average filter and median filter. Strong correlation is exhibited using neighboring pixels for natural image. Edge pixels and neighboring pixels gray value are very close to every pixel. Pixel is contaminated by noise, if pixel value is greater or less than neighboring pixel value, else the pixel is an available pixel.

In the process of noise reduction, every pixel is checked sequentially and the pixel is judged as a noisy pixel, if average value of pixels in the mask is less than the pixel value and median value of pixels in mask are used for replacing it, else original pixel value is maintained as it is. In mask, median value of is used for replacing the original value and in the next computation process, using new pixel values, average value is computed. Iterative process is formed using this and noise-reducing effect is enhanced better while minimizing the time complexity.

For Example

Pixel (i, j)'s noise is reduced with 3 × 3 as neighborhood size. Value of median is $f'(i,j)$, If $-average > 0$. If $f'(i,j) < f\ (i,j)$, then $f'(i,j)$ is a noisy pixel.Pixel $(i, j+1)$'s average and median values in conventional algorithm are expressed as,

$$Average = \{f(i-1,j) + \cdots f(i,j) + f(i,j+1) + \cdots f(i+1,j+2)\}/9$$

$$Med = \{f(i-1,j) + \cdots f(i,j) + f(i,j+1) + \cdots f(i+1,j+2)\}$$

If improved algorithm factor$'(i,j)$ is used for replacing $f(i,j)$, then average and median values are computed as,

$$average = \{f(i-1,j) + \cdots f'(i,j) + f(i,j+1) + \cdots f(i+1,j+2)\}/9$$

$$med = \{f(i-1,j) + \cdots f'(i,j) + f(i,j+1) + \cdots f(i+1,j+2)\}$$

The *average* is less than *Average* as per $f'(i,j) < f\ (i,j)$. So, there will be an increase in noise reduction's spatial extend and reduction in improved algorithm's time complexity. Improved algorithm steps are given by,

a. For searching center element *f* (*i*, *j*), over image, mask slides and image pixel and mask center is getting overlapped.
b. Read the corresponding mask pixels value;
c. Mask's average value (*average*) is computed;
d. Average value is compared with every pixel value. Median value is searched if average value is less than every pixel value and assume let $f(i, j) = med$, else retain the pixel's original value as it is.
e. Until $= j = n$, step (4)is repeated.

Fast Computation of the Median Value

Median value searching speed can be enhanced using statistical histogram.

Following are the steps of this method.

a. For gray histogram hist[i] computation of $n \times n$mask, assume $N = n \times n$, median value *med* is computed and record *ltmed*. Where, gray value range is represented as G, $0 < i < G$, pixel values count less than med is given by ltmed.
b. If shifting out value pixels is less than *med*, then $ltmed - 1$ and perform left row shift out in histogram.
c. If shifting value in pixels is less than*med*, then $ltmed - 1$ and perform right row shift in histogram.
d. If $ltmed < N/2$, then $med + 1$, $ltmed + hist[med]$ is repeated until $ltmed = N/2$;
e. If $ltmed > N/2$, then $med - 1$, $ltmed - hist[med]$is repeated until $ltmed = N/2$;
f. Return to median value*med*.

When compared with conventional median filtering algorithm, two enhancements are made in improved algorithm. One is, sliding mask's historical information is used for making compared pixels count as N and mask's original median value is always compared with every pixel value. Second one is to use limited gray-level distribution range characteristics for greatly decreasing median algorithm's complexity.

Analysis of the Complexity of the Algorithm

Considers an array for solving median value as $X = \{X_i\}$ $(i = 1, 2, \ldots, N)$, where, $0 \leq x_i \leq M$ and integer is represented as x_i. if median value is computed using statistical histogram method, then algorithm complexity is about *O* (*N*), with maximum required number as N.

C. Feature Extraction

For processing, required resources count are minimized using feature extraction. This will not leads to relevant or important information loss. In input image, for extracting features set, proposed a feature extraction technique based on Local Binary Pattern (LBP).

Local Binary Pattern (LBP) based histogram

Surface's texture characteristics are described using method called LBP. Probability of texture pattern can be summarized as histogram by LBP application. For all pixels in an image, it is required to compute LBP values. According to LBP histogram's distribution shape, compute Texture regularity. This operator is an effective texture operator. Image pixels are labelled by thresholding every pixels neighborhood and result is considered as binary number. In various applications, it is applied. From input image, extract the LBP histograms, then single vector is formed by concatenating these histograms [22].

LBP is a texture description operator and differences signs between central and neighboring pixels forms base for this operator. For every pixel value of an image, its neighboring values are thresholded using central pixel for computing binary code. Neighboring pixels values which is less than threshold are assigned with 0 value, in other cases, if it is greater than threshold value, then it is assigned with value of 1. For computing binary patterns frequency values, constructed a histogram. In an image, possibility of binary pattern is represented using every pattern.

In LBP computation, involved pixels count defines histogram bins count. For example, if 8 pixels are used in LBP computation, then there will be 2^8 or equal to 256 histogram pins. For a 3×3 neighbor pixels, center pixel is used as a threshold in LBP operators basic version. Binary pattern which represents texture characteristics is created using threshold operation. The LBP bitmaps are computed using textures as object. For every LBP bitmap, extract the histogram. Basic LBP's expression is given as follows,

$$\mathrm{LBP}(\mathrm{x_c}, \mathrm{y_c}) = \sum_{\mathrm{n}=0}^{7} 2^{\mathrm{n}} \mathrm{g}(\mathrm{I_n} - \mathrm{I}(\mathrm{x_c}, \mathrm{y_c}))$$

At center pixel (x_c, y_c), LBP value is expressed as $\mathrm{LBP}(x_c, y_c)$, neighboring pixel value is represented as $\mathrm{I_n}$ and center pixel value is represented as $\mathrm{I}(x_c, y_c)$. Neighboring pixels index is represented as index n. If $\mathrm{x} < 0$, then the function g(x)=0, and if $x \geq 0$, then the function g(x)=1. Neighboring pixels values which is less than threshold are assigned with 0 value, in other cases, if it is greater than threshold value, then it is assigned with value of 1. Between weight and binary matrices, scalar multiplication is applied for computing LBP value. At last, LBP value is represented using addition of all results of multiplication.

From denoised input mage, features quality are extracted. Extracted a LBP feature set, which is not effected by image orientation and effects of illumination. In a fingerprint image I, for a specified central pixel, LBP(P, R)is calculated as,

$$LBP_{P,R} = \sum_{P=0}^{P-1} s(I_p - I_c) 2^P$$

$$s(x) = \begin{cases} 1, x \geq 0 \\ 0, x < 0 \end{cases}$$

Where, thresholding function is represented as *s(x),* in I, central pixel color is represented as I_c, neighbors color values are represented as I_p , neighbors count considered for LBP feature computation is represented as P, neighborhood user's radius is represented as R. Input image's size is assumed as *M x N*. Building histogram is used for representing entire input image, after the computation of every pixel *(i, j)'s* LBP.

$$H(k) = \sum_{i=1}^{M} \sum_{j=1}^{N} f(LBP_{P,R}(i,j), k), k \in [0, K]$$

$$f(x, y) = \begin{cases} 1, & x = y \\ 0, otherwise \end{cases}$$

Where, in input image, existing maximum LBP value is represented as K.

D. Dimensionality Reduction and Segmentation

There are four main stages in this Fruit flower segmentation. They are, formation of small patches by splitting high resolution image in a sliding window manner, for reducing dimensionality Improved Linear Discriminant Analysis (ILDA) is used, fine-tuned HGAECNN based patch evaluation, final segmentation mask computation using computed score-maps via application of refinement algorithm.

1. Sliding Window

Resolution of images in dataset ranges from2704×1520 to 5184×3456 pixels. Resampling artifacts are avoided using sliding window technique. Formed a set P of n portraits $\mathrm{p^{(i)}} \in \mathrm{P}$ by spitting every input image. Size of every portrait is 321×321 pixels i.e. $\mathrm{p^{(i)}} \in \mathrm{R^{r \times r}}$ with r = 321. Artificial boundaries are introduced by cropping of non-overlapping portraits from original image, quality of detection is compromised by this. So, for this, every portrait made overlapped in this technique. Every immediate neighbor area percentage is represented as s. In this experimentation, value of s is 10%. Discarded the results respective to overlapping pixels, if fusion of scoremaps happens. For subsequent portrait pair, this process is illustrated in figure 2. For every portrait, obtained score is depicted as heatmap. Lower scores are represented with blue and red is used for representing high scores.

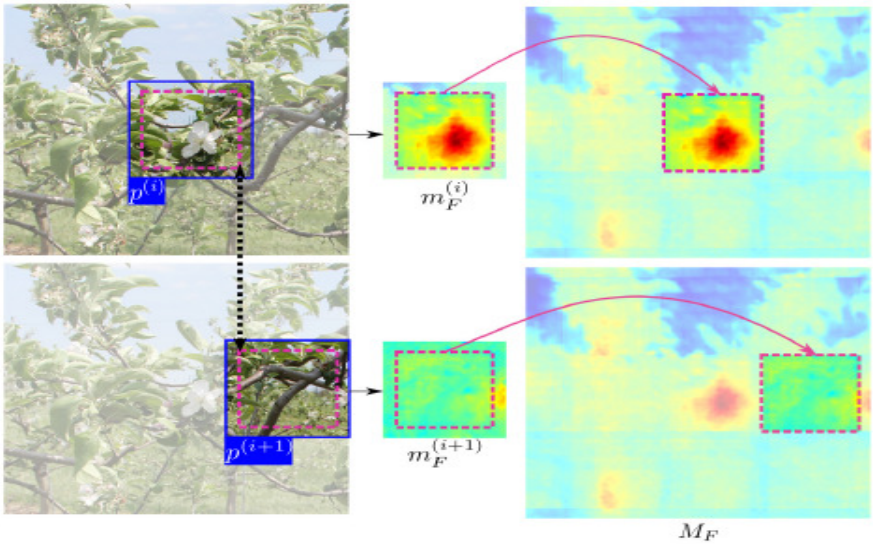

FIGURE 2: BEST VIEWED IN COLOR

2. Improved Linear Discriminant Analysis (ILDA) for dimensionality reduction

Low dimensional space is formed by transforming original input features using subspace learning method. Linear Discriminant Analysis (LDA) is most popular subspace learning method. It used some selection parameter for computing pertinent feature subset from original dataset. Noisy, redundant and irrelevant features are removed in this method for reducing image's dimensionality. So, time of computation is minimized by futile feature removal and dimensionality reduction [23].

Linear Discriminant Analysis (LDA)

For reduction of dimensionality, well-known statistical technique called LDA is used. From a high-dimensional space, low dimensional features are computed using it. A linear projection matrix $W \in R^{d\times m}$ is computed using LDA based on Fisher's criterion by minimizing within-class scatter matrix ration and maximizing between-class scatter matrix ratio, which is expressed as,

$$\arg\max_{W} tr((W^T S_w W)^{-1}(W^T S_b W))$$

$$S_b = \sum_{n=1}^{m} k_n(\mu_n - \mu)(\mu_n - \mu)^T$$

$$S_w = \sum_{n=1}^{m} \sum_{i=X_n} (p_i - \mu_n)(p_i - \mu_n)^T$$

Where, n[th] classes index set is represented as $X_n$and its mean vector is represented as μ_n.

There are three scatter matrices are in LDA, they are, total scatter matrix, between-class matrix and within-class matrix. Using following expression, total scattering matrix is computed as,

$$S_t = S_w + S_b$$

With stronger sub-groups, LDA cannot be applicable. So, proposed an Improved Linear Discriminant Analysis (ILDA) is for solving this issue.

Improved Linear Discriminant Analysis (ILDA)

For reducing subspace, instead of trace value, for between-class scatter matrix and within class scatter matrix, computed Euclidean Norm in ILDA. Given input matrix diagonal values are only considered for calculating trace value. For best feature subset computation, it becomes a drawback. So, entire matrix is considered in ILDA technique for computing Euclidean Norm value.

$$I_{LDA} = \|(W^T S_w W)^{-1}(W^T S_b W)\|$$

Where, between-class matrix is represented asS_b and within-class matrix is represented as S_w.

3. Hybrid Genetic based Auto-Encoder Convolutional Neural Network (HGAECNN)

a. Auto-Encoder Convolutional Neural Network (AECNN)

For flower identification, every portrait $p^{(i)}$is given with fine-tuned network in parallel. Function f corresponds to AECNN as,

$$f: p^{(i)} \rightarrow \{m_F^{(i)}, m_B^{(i)}\}$$

Every input $p^{(i)}$ is mapped into two pixel-dense scoremaps using above function. Pixel-wise likelihood of pixels in $p^{(i)}$ belonging to foreground is represented as $m_F^{(i)} \in R^{r \times r}$, pixel-wise background likelihood is represented as $m_B^{(i)} \in R^{r \times r}$.

Auto-Encoder

There are two-parts in auto-encoder. Figure 3 shows encoder and decoder. Deterministic mapping function is used for converting input x to a hidden representation y which is a feature code in encoder. It is an affine mapping function, which follows non-linearity [24].

$$y = f(Wx + b)$$

Where, weight between hidden representation y and input x is represented as W and bias is represented as b. Construction process of output z using y is implemented by decoder and it is expressed as,

$$z = f'(W'y + b')$$

Where, weight between output z and hidden representation y is represented as W' and bias is represented as b'. In auto encoder, major role is played by parameters like bias (b) and weight (W). It is difficult to tune these parameters. So, in this work, introduced a genetic algorithm (GA) for solving this issue.

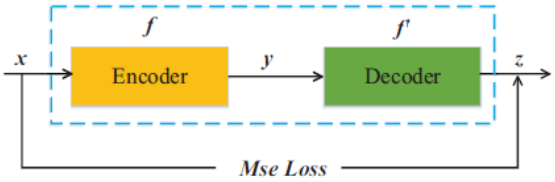

FIGURE 3: THE ARCHITECTURE OF AN AUTO-ENCODER

Reconstruction error can be minimized by training an auto-encoder, cost function J_{AE} minimization is used for realizing the same.

$$J_{AE} = \frac{1}{p}\sum_{i=1}^{p} L[x_i, z_i]$$

Where, input images count is represented as p, input image is represented as x_i and reconstructed image corresponds to $x_i$is represented as z_i, input image x_i's reconstruction error is represented as $L[x_i, z_i]$. Cross entropy or Mean Square Error (MSE) are used for measuring this value. MSE between reconstructed patch of image $z_i(i = 1,2, \dots, p)$ and input image $x_i, (i = 1,2, \dots p)$ is utilized in this work. The reconstructed image can be represented as,

$$L_{AE}[x_i, z_i] = \|x_i - z_i\|^2$$

Inputs are added with convolution operation by combining auto-encoder with local convolution connection using Convolutional auto-encoder. There exist a convolutional decoder and convolutional encoder in convolutional auto-encoder. Convolutional conversion process to feature maps from input is realized using convolutional encoder and convolutional conversion process to output from feature map is carried out using a convolutional decoder.

b. Genetic Algorithm (GA)

Natural fruit flower selection process is inspired in a meta heuristic algorithm called Genetic Algorithm (GA) which is huge type of evolutionary algorithms (EA). For optimization, high-quality segmented accuracy is generated using GA and for search problems better results are produced using bio-inspired operators like selection, crossover, mutation, etc. [25]. There are set of properties for every candidate segmented accuracy, it may be of genotype or chromosome and they are altered and mutated. Binary strings of 1s and 0s are used for representing traditionally selected images with the possibility of other encodings.

Species evolution are simulated through natural selections for optimization technique implementation in GA. There are two process in GA. Selection of fruit flower image individually is a first process, which is used for next generation production and selected individual fruit flower images manipulation is a next process, which is used for next generation formation. These are done using mutation and crossover methods. Being individual is betters a major selection technique principle and makes high probability of being a parent. Genetic operations like mutation, crossover and selection are used for searching better segmented accuracy in GA.

i. Selection Operation

In current population, parents are selected using fruit flower image elitist individuals in selection operation and offspring are generated using it. Elitist nature of individuals are computed using a parameter called fitness. Various techniques like elitism selection, steady-state selection, rank selection, tournament selection, Boltzman selection and few other methods are also used for selecting best images. Few methods are described in the following section.

Roulette Wheel Selection

Based on fitness value, selected the parents. More chances of getting selected is received by better chromosomes. Consider a roulette wheel with place for all chromosomes of population and their places are defined based on its fitness value. High times of selection is received by chromosome having huge fitness value.

Rank Selection

With huge fitness variation, there arises few problems in the method described previously. For instance, best chromosome with 90% fitness value is having higher chance of getting selected in fruit flower image and the chances of other chromosomes will be very less. Based on fitness, population is sorted in rank selection and from this ranking, fitness is assigned to every chromosome. Fitness 1 is assigned to worst and next will receive fitness 2 and so on. Fitness N is assigned to the best one, which is population's chromosome's count. Thereafter, chances for getting selected in fruit flower image is given to all chromosomes. Instead of fitness, rank of a chromosome in sorted list defines the probability of that chromosome in getting selected for fruit flower image.

Elitism Selection

If mutation and crossover operations are used for creating new population, there is a chance of missing out the best chromosome. In an elitism method, best chromosome or best few chromosomes are copied to new population and classical way is followed for rest of the process. Loss of best chromosome is avoided in elitism, so, there will be GA algorithm performance enhancement in a rapid manner with enhanced segmentation accuracy.

ii. Crossover Operations

Current population's selected members are recombined and mutated using operator set, which determines GA. Mutation and crossover are the two commonly used operators. From two parents, two new offspring are generated by crossover operator via copying selected AECNN parameters of every parent. From any one of two parents, AECNN parameters at ith position in every offspring is copied from AECNN parameters at ith position. Addition string termed as crossover mask defines, which parent is contributing AECNN parameter at ith position. There exist three classes of crossover operators, namely uniform, two-point and single point operators.

Single-Point Crossover

Always constructed a cross-over mask in single point crossover. There are n contiguous 1s string which is followed by required number count of 0s for string completion. Offspring is produced by it. In this, one parent is contributed with firstn AECNN parameters and second parent is contributed with remaining AECNN parameters. At random, selected crossover point n during the application of single point crossover operator at every time and created a crossover mask and applied. From first parents AECNN parameters, first five are selected in this offspring and from second parents, balance six bits are selected. For every AECNN parameter positions, these choices are

specified by crossover mask 111110000000. Two parents roles are switched in second offspring, but same crossover mask is utilized. Unused AECNN parameters of first offspring will be in this second offspring.

Two-Point Crossover

Middle of second parent string is substituted with first parents intermediate segmentation accuracy for creating offspring in two-point crossover. Crossover mask is generated with n0 zeros in the beginning followed by n1 ones contiguous string and followed by required zeros for string completion. Integers n0 and n1 are selected randomly for generating mask during the application of two-point crossover at every time.

Uniform Crossover

From two parents, AECNN parameters are sampled uniformly and combined using uniform crossover. AECNN's random parameter corresponds to generated crossover mask in this case with randomness and independency among every bit.

iii. Mutation Operations

Two parents parts are combined for producing offspring in recombination operation which is a type of mutation operation. In another type of operation, single parent is used for producing offspring. In a random manner single bits are selected and their values are changed for producing small random changes to bit string in mutation operator.

4. Fusion and Refinement

Predictions obtained for every $p^{(i)} \in P$ are combined for generating two global scoremaps$M_B$and M_F, after the evaluation of every portrait. After padding pixel elimination in I, $p^{(i)}$'s pixel coordinates are represented as $c^{(i)}$. Same resolution as like I is contained in scoremaps$M_B$and $M_F$as defined by fusion procedure.

$$\forall p^{(i)} \in P, M_{F,B}\left(c^{(i)}\right) = m^{(i)}_{F,B,}$$

Exactly one portrait is used for obtaining single prediction score for each image pixel. This is done to avoid artificial boundaries produced artifacts. Softmax function is used after fusion for normalizing scoremapsM_B and $M_F$into scoremaps$\widetilde{M}_B$ and $\widetilde{M}_F$.

$$\widetilde{M}_{F,B}(q_j) = \frac{\exp\left(M_{F,B}(q_j)\right)}{\exp\left(M_B(q_j)\right) + \exp\left(M_F(q_j)\right)}$$

Where, in input image I, j-th pixel is represented as q_j. For every pixel q_j, scores $\widetilde{M}_B\left(q_j\right)$and ~$\widetilde{M}_F\left(q_j\right)$ are added using this formulation. This indicates probability of q_j belonging to that class. With respect to actual flower boundaries, obtained CNN predictions are coarse. So, image I and score map are given to RGR refinement module, instead of directly thresholding$\widetilde{M}_F$. Two high-confidence classification regions R_F and R_B defines this refinement algorithm and is given by,

$$R_{F,B} = \{q_j | \widetilde{M}_{F,B}(q_j) > \tau_{F,B}\}$$

Where, high-confidence background threshold is represented as τ_B and high-confidence foreground threshold is represented as τ_F. Multiple Monte Carlo region growing steps are performed using Region Growing Refinement (RGR) algorithm with high-confidence regions as starting points for forming clusters using similar pixels. Every generated cluster is classified into background or foreground by RGR via conduction of majority voting procedure.

Coarse Segmentation

Refined Segmentation

4. RESULTS AND DISCUSSION

Two public datasets namely, AppleB and AppleA are used for evaluating the propose method [27]. In diverse controlled environment with various capturing angle, collected the images of various fruit flower species, which are presented in table 1. In USAD orchard, on a sunny day, collected apple tree images for forming AppleB and AppleA dataset. Trellises are used for supporting trees and in rows they are planted in both datasets. A hand-held camera is used for acquiring 147 image collection to form AppleA dataset.

Training set is constructed by selecting 100 images from this collection in a random manner and which is used for training AECNN. Resolution of images in AppleA dataset is $4.3\times$ greater than other dataset images. Rather than splitting images into 321×321 pixels portraits, AppleA dataset images are split as 155×155 pixels portraits. At pixel-level, metrics like Intersection-over-Union (IoU), F-Score(F_1), Recall(R) and Precision(P), are computed which defines segmentation accuracies quantitative analysis in proposed method [27]. These metrics are computed at super-pixels level in existing methods.

TABLE 2: PERFORMANCE COMPARISON METRICS VS. FRUIT FLOWER DETECTION METHODS

Dataset Name	Methods/Metrics	Precision(P) (%)	Recall(R) (%)	F-Score(F_1) (%)	Intersection-over-Union (IoU) (%)
AppleA	**CNN**	39.7	60.5	50.7	40.5
	AECNN	67.1	77.2	73.8	55.3
	HGAECNN	85.4	91.7	88.3	78.4
AppleB	**CNN**	68.1	63.9	72.5	54.3
	AECNN	70.1	72.4	73.2	60.6
	HGAECNN	77.8	95.2	80.3	70.0

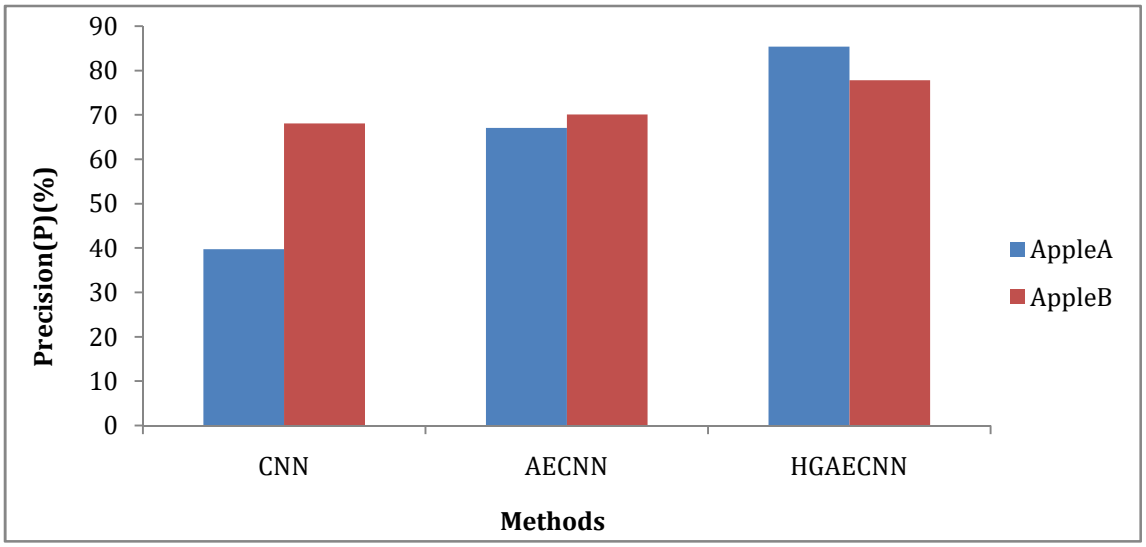

FIGURE 4: PRECISION (P)COMPARISON OF FRUIT FLOWER DETECTION METHODS

Precision metric comparison of various deep learning techniques are demonstrated in figure 4. High Precision results are produced by proposed HGAECNN deep learning techniques on AppleB and AppleA datasets, whereas, less results are produced by AECNN and CNN as demonstrated by this results.

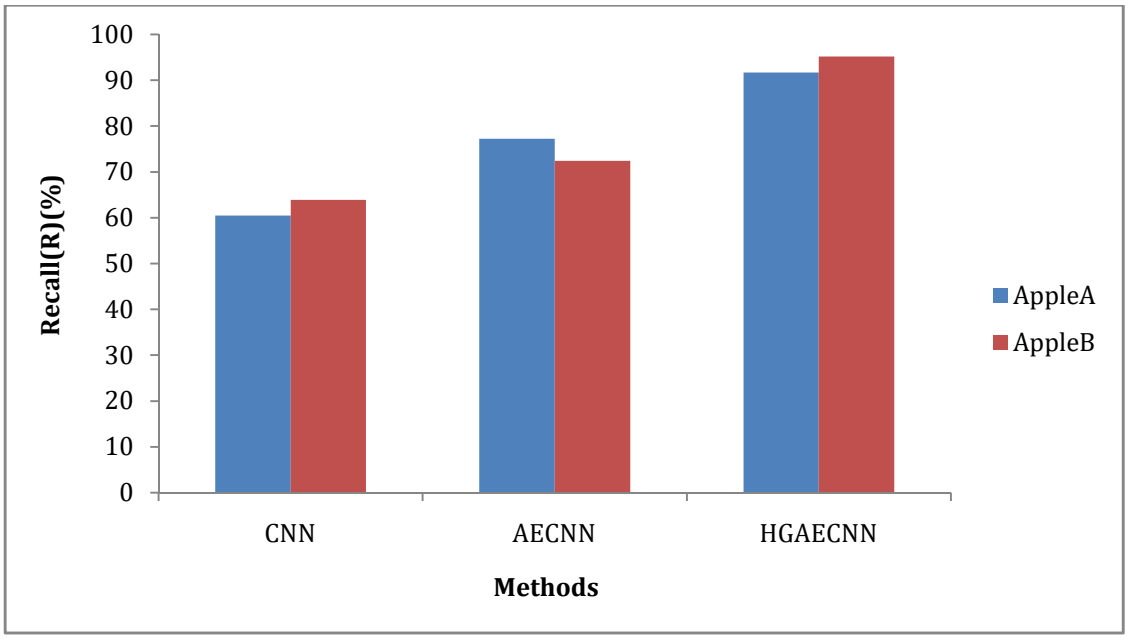

FIGURE 5: RECALL (R) COMPARISON OF FRUIT FLOWER DETECTION METHODS

Recall metric comparison of various deep learning techniques are demonstrated in figure 5. High recall results are produced by proposed HGAECNN deep learning techniques on AppleB and AppleA datasets, whereas, less results are produced by AECNN and CNN as demonstrated by this results.

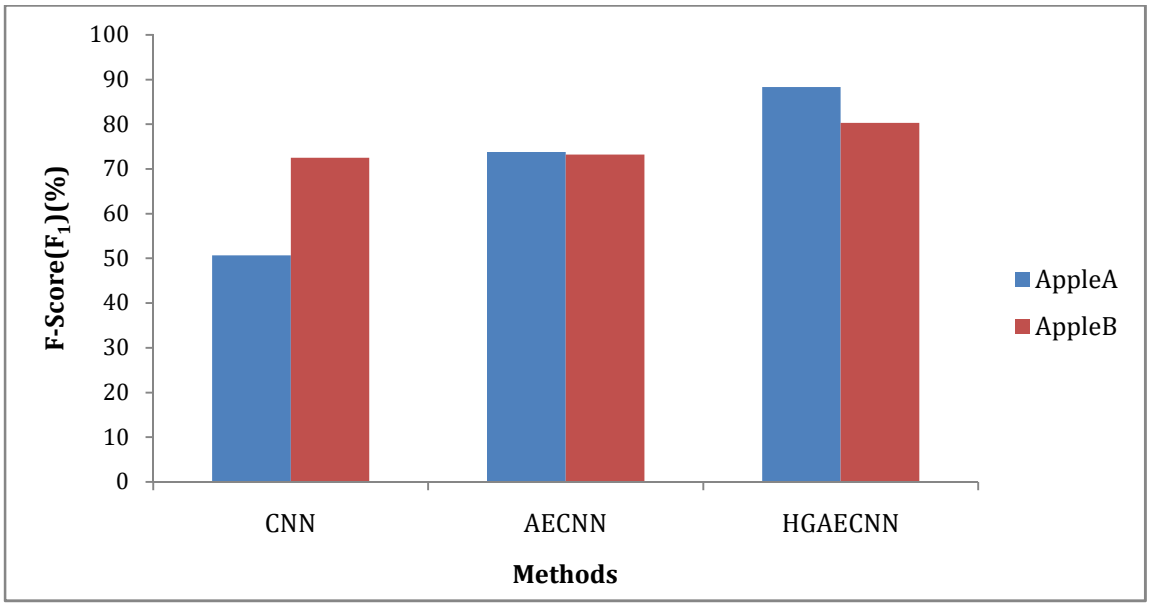

FIGURE 6: F-SCORE(F_1)COMPARISON OF FRUIT FLOWER DETECTION METHODS

F-scoremetric comparison of various deep learning techniques are demonstrated in figure 6. High F-score results are produced by proposed HGAECNN deep learning techniques on AppleB and AppleA datasets, whereas, less results are produced by AECNN and CNN as demonstrated by this results.

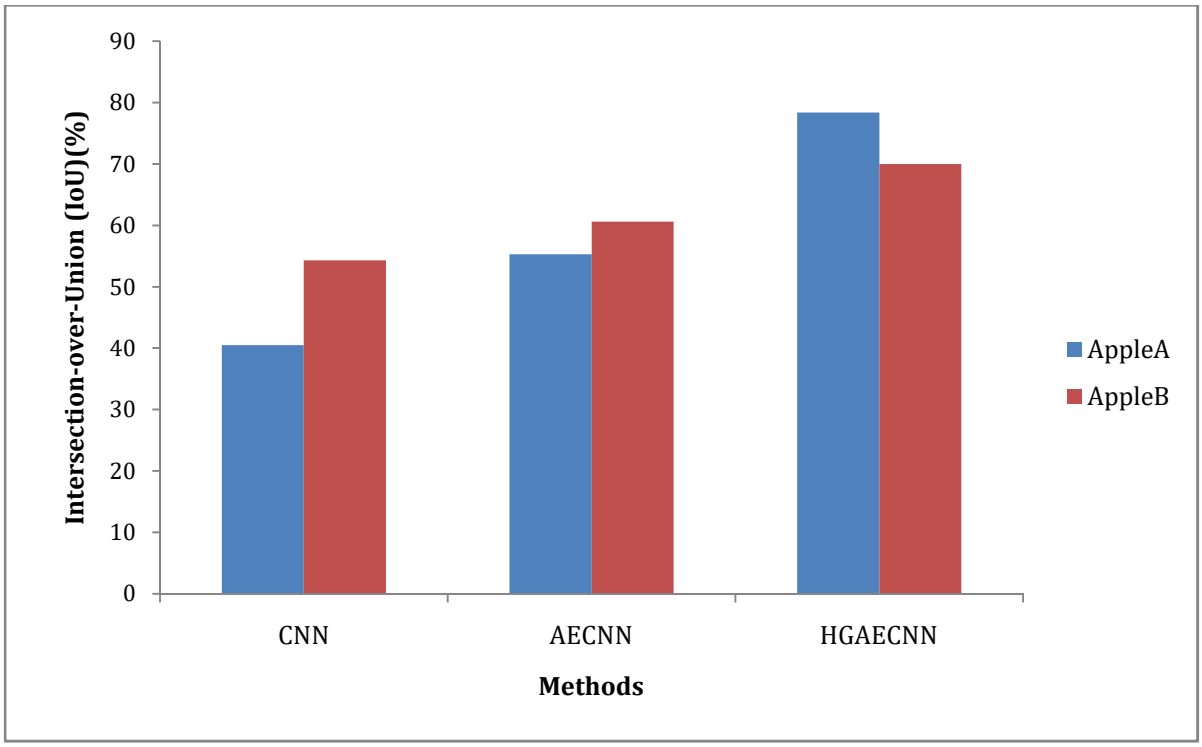

FIGURE 7: INTERSECTION-OVER-UNION (IoU) COMPARISON OF FRUIT FLOWER DETECTION METHODS

Intersection-over-Union (IoU) metric comparison of various deep learning techniques are demonstrated in figure 7. High Intersection-over-Union (IoU) results are produced by proposed HGAECNN deep learning techniques on AppleB and AppleA datasets, whereas, less results are produced by AECNN and CNN as demonstrated by this results.

5. CONCLUSION AND FUTURE WORK

In agriculture sector, fruit yield's advance prediction is more important. It is used for fruit yield prediction per orchard as well as for fixing and computing cost of the product. Intensity of flower is used for this detection. For semantic segmentation of images, state-of-the-art deep learning techniques are exploited in novel flower detection methods. In dataset where high variation with respect to flower species, flower density, image resolution, background composition and illumination conditions, better results are produced by proposed method.

There are four main stages in this work. They are, deep FCN network training, pre-processing, extraction of feature, reduction of dimensionality and segmentation. On commercial GPU, deep FCN is used for high-resolution evaluation. GPU memory space is needed for computation of fully convolution and based on resolution of image, this requirement increases exponentially. Neighboring information are used for spotting noises in images and they should be reduced or removed. Without affecting quality of an image, better filtering method is used for removing it and reinforce image smoothness.

For extraction of feature, LBP is used in third stage. For all pixels of image, it needs to be computed. According to LBP histogram distribution shape, computed texture regularity. In a sliding window manner, smaller patches are formed by splitting high resolution image. From original dataset, pertinent features subset are computed using a process called ILDA with some selection parameter. For AECNN hyper-parameters values are optimized for maximizing segmentation accuracy of genetic algorithm model.

In auto encoder, bias (b) and weight (W) which are the parameters of AECNN plays a crucial role. So, genetic algorithm is used for tuning AECNN parameters. Every generated cluster is classified into background or foreground by RGR via conduction of majority voting procedure. Final mask of segmentation is computed using obtained score –map with the application of refinement algorithm.

For further enhancing this model's generalization ability, multi-species flower datasets can be used in future for training and evaluation. Complete autonomous online bloom intensity estimation system can be developed.

REFERENCES

1. J.P. Underwood, C. Hung, B. Whelan and S. Sukkarieh, Mapping almond orchard canopy volume, flowers, fruit and yield using lidar and vision sensors, Computers and electronics in agriculture, Vol.130, Pp.83-96, 2016.
2. M. Paustian and L. Theuvsen, Adoption of precision agriculture technologies by German crop farmers, Precision agriculture, Vol.18, No.5, Pp.701-716, 2017.
3. P.A. Dias, A. Tabb and H. Medeiros, Apple flower detection using deep convolutional networks, Computers in Industry, Vol.99, pp.17-28, 2018.
4. A. Gongal, A. Silwal, S. Amatya, M. Karkee, Q. Zhang and K. Lewis, Apple crop-load estimation with over-the-row machine vision system, Computers and Electronics in Agriculture, Vol.120, Pp. 26–35, 2016.
5. Y. Song, C.A. Glasbey, G.W. Horgan, G. Polder, J.A. Dieleman and G.W.A.M. Van der Heijden, Automatic fruit recognition and counting from multiple images, Biosystems Engineering, Vol.118, Pp.203-215, 2014.
6. U.O. Dorj, M. Lee and S. Han, A comparative study on tangerine detection, counting and yield estimation algorithm, International Journal of Security and Its Applications, Vol.7, No.3, Pp.405-412, 2013.
7. G.L. Grinblat, L.C. Uzal, M. G. Larese and P.M. Granitto, Deep learning for plant identification using vein morphological patterns, Computers and Electronics in Agriculture, Vol.127, Pp.418–424, 2016.
8. A. Fuentes, S. Yoon, S.C. Kim and D.S. Park, A robust deep-learning-based detector for real-time tomato plant diseases and pests recognition, Sensors, Vol.17, No.9, Pp.1-21, 2017.
9. U.O. Dorjand M. Lee, A new method for tangerine tree flower recognition, Computer Applications for Bio-technology, Multimedia, and Ubiquitous City, Springer, Berlin, Heidelberg, Pp.49-56, 2012.
10. M. Hočevar, B. Širok, T. Godeša and M. Stopar, Flowering estimation in apple orchards by image analysis, Precision agriculture, Vol.15, No.4, Pp.466-478, 2014.
11. M.P. Diago, A. Sanz-Garcia, B. Millan, J. Blascoand J. Tardaguila, Assessment of flower number per inflorescence in grapevine by image analysis under field conditions, Journal of the Science of Food and Agriculture, Vol.94, No.10, Pp.1981-1987, 2014.
12. Z. Wang, B. Verma, K.B. Walsh, P. Subediand A. Koirala, Automated mango flowering assessment via refinement segmentation, International Conference on Image and Vision Computing New Zealand (IVCNZ), Pp.1-6, 2016.
13. Z. Wang, J. Underwood and K.B. Walsh, Machine vision assessment of mango orchard flowering, Computers and Electronics in Agriculture, Vol.151, Pp.501-511, 2018.
14. P.A. Dias, A. Tabb and H. Medeiros, Multispecies fruit flower detection using a refined semantic segmentation network, IEEE Robotics and Automation Letters, Vol.3, No.4, Pp.3003-3010, 2018.
15. Y. Chen, W.S. Lee, H. Gan, N. Peres, C. Fraisse, Y. Zhang and Y. He, Strawberry Yield Prediction Based on a Deep Neural Network Using High-Resolution Aerial Orthoimages, Remote Sensing, Vol.11, No.13, Pp.1-21, 2019.
16. A. Koirala, K.B. Walsh, Z. Wang and N. Anderson, Deep Learning for Mango (Mangiferaindica) Panicle Stage Classification, Agronomy, Vol.10, No.1, Pp.1-22, 2020.
17. Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama and T. Darrell, Caffe: Convolutional architecture for fast feature embedding, Proceedings of the 22nd ACM International Conference on Multimedia, Pp.675–678, 2014.
18. N. Kumar and M. Nachamai, Noise removal and filtering techniques used in medical images, Oriental Journal of Computer Science & Technology, Vol.10, No.1, Pp.103-113, 2017.
19. A. Peng, S. Luo, H. Zeng and Y. Wu, Median filtering forensics using multiple models in residual domain, IEEE Access, Vol.7, Pp.28525-28538, 2019.

20. J. Tang, Y. Wang, W. Cao and J. Yang, Improved Adaptive Median Filtering for Structured Light Image Denoising, IEEE 7th International Conference on Information, Communication and Networks (ICICN), Pp.146-149, 2019.

21. Y. Li, An Improved Median Filtering Image Denoising Algorithm, 2nd IEEE Advanced Information Management, Communicates, Electronic and Automation Control Conference (IMCEC), Pp.2707-2710, 2018.

22. E. Prakasa, Texture feature extraction by using local binary pattern, INKOM Journal, Vol.9, No.2, Pp.45-48, 2016.

23. D. Chu, L.Z. Liao, M.K.P. Ng and X. Wang, Incremental linear discriminant analysis: a fast algorithm and comparisons, IEEE transactions on neural networks and learning systems, Vol.26, No.11, Pp.2716-2735, 2015.

24. B. Hou and R. Yan, Convolutional Auto-Encoder Based Deep Feature Learning for Finger-Vein Verification, IEEE International Symposium on Medical Measurements and Applications (MeMeA), Pp.1-5, 2018.

25. M.A. El-Shorbagy, M.A. Farag, A.A. Mousa and I.M. El-Desoky, A hybridization of sine cosine algorithm with steady state genetic algorithm for engineering design problems, International Conference on Advanced Machine Learning Technologies and Applications, Pp.143-155, 2019.

26. https://data.nal.usda.gov/dataset/data-multi-species-fruit-flower-detection-using-refined-semantic-segmentation-network

27. M. Everingham, S.A. Eslami, L. Van Gool, C.K. Williams, J. Winn and A. Zisserman, The pascal visual object classes challenge: A retrospective, International journal of computer vision, Vol.111, No.1, Pp.98-136, 2015.