VOLUME - 11, ISSUE - 08, AUGUST - 2022 • PRINT ISSN No. 2277 - 8160 • DOI : 10.36106/gjra

Original Research Paper | Computer Science

AN IDENTIFICATION OF CROP AND WEED CLASSIFICATION IN SATILLIATE IMAGE USING NEURAL NETWORK

Mr. P. Sedhupathy — Research Scholar, Department of Computer Science, Gobi Arts & Science College, Karattatipalayam, Gobichettipalayam, Tamilnadu, India-638453

Dr. R. Prabahari — Assistant Professor, Department of Computer Science, Gobi Arts & Science College, Karattatipalayam, Gobichettipalayam, Tamilnadu, India-638453

ABSTRACT

Introduction: Usage of independent robotics to precision agriculture is emerging to be a hot topic among the research groups, with due credit to the best influence it may show in food security, sustainability and decrease in chemical treatments (Yahaya et al., 2018; Pantazi et al., 2016). It is important to use agrochemicals for controlling weeds effectively. But, they can result in adversely affecting the environment and human health too (Montalvo et al.,2013). Hence, precision agriculture faces one of the biggest challenging tasks of limiting the usage of agrochemicals like pesticides, herbicides, and fertilizers, when sustaining high crop productivity. However, manually monitoring weeds and crops is a time-consuming activity, making it unfeasible for large-scale agriculture (Shorewala et al.,2021). The Quad Histogram with Modified Convolutional Neural Network (MCNN) is utilized previously for categorizing crop as well as weed. Here this research exploits quad tree decomposition for feature extraction in addition Modified Convolutional Neural Network (MCNN) for classification. Still, color features are exploited for weed categorization. Classification rate is enhanced through several texture features which are also necessitated. In addition, single classifier is also sufficient to yield improved outcomes, henceforth ensemble learning is desired for prediction performance upgrading.

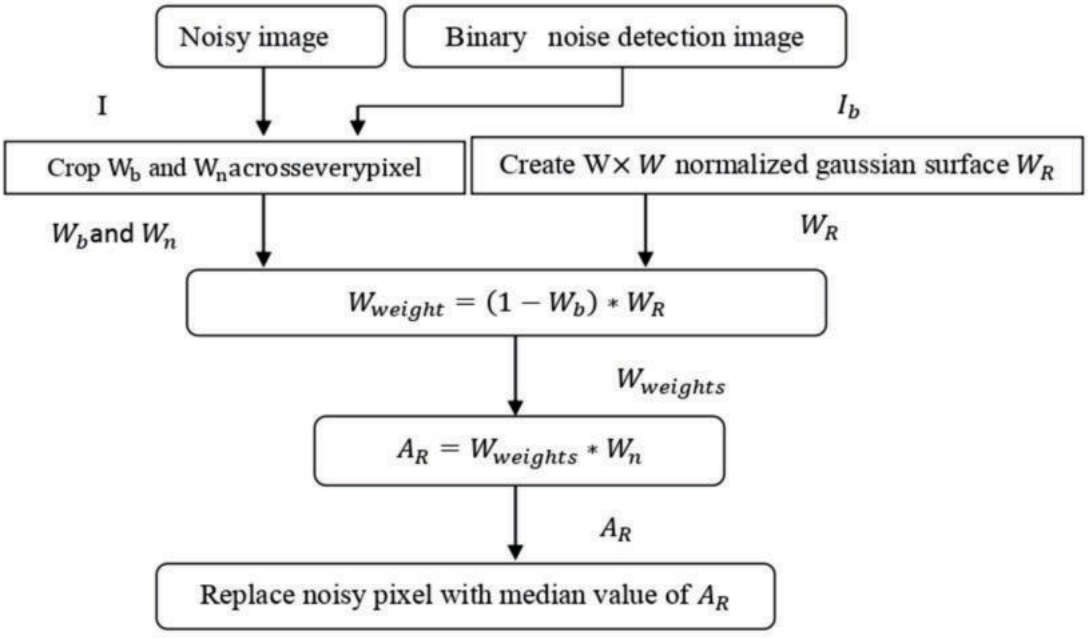
In this proposed work, Ensemble Modified Convolutional Neural Network (EMCNN) is a promising solution for categorizing crop as well as weed merely. Basically, the input images in this research are Near-Infrared (NIR) and red images. Normalized Difference Vegetation Index (NDVI) from NIR as well as red patch images extraction is accomplished through simple automated image processing techniques. Noise removal from images in a proficient way can be attained through dynamically weighted median filtering algorithm which is followed by color feature extraction via Quad Histogram.

KEYWORDS : Classification, Image, Agriculture

PRE-PROCESSING

Image Alignment

Image correlation in addition to cropping is utilized for geometric transformation performing simple image processing for NIR and Red images. There is only negligible processing time for these processes due to single transformation computation for cameras involved firmly with respect to each other. The similarity lack is another major concern for misalignment with other image channels e.g. Green and Red Edge. The matching difficulty is another concern deprived of depth assessment of every pixel precisely. Consequently, camera baseline is considered to be smaller than distance from ground as well as camera (~two orders of magnitude). Camera intrinsics has its significance on system besides radiometric and atmospheric corrections applications does not relate to the system.



Dynamically Weighted Median Filtering Algorithm

Dynamically Weighted Median Filter (DWMF) is greatly involved for preprocessing in this research. Value 0 weightage assigning is done to those locations in a W*W window that are identified as noisy pixels through this DWMF. 2D Gaussian surface is yet another aspect utilized for weightage window automatic selection. The peculiar property of Gaussian function is that its intensity rises as migrate towards center.

The DWMF inputs are Binary image I_b as well as noisy image I. Noise detection algorithm is utilized for obtaining Binary image I_b .Factors such as W*W patches, W_n and W_b are chosen across complete detected noisy pixels in mutually noisy image I in addition binary image I_b correspondingly. The Weightage window $W_{weights}$of size W*W computation is done, however $W_{weights}$ location extraction is accomplished where W_b possess values of 1 (if complete entries of W_b are 1, $W_{weights}$ is substituted through W_b to elude exception). By this manner, 0 weightage is assigned for noisy pixel detection. Due to the gaps, allotted weights in $W_{weights}$ are shifted because of elements removal at noisy locations. For instance, if entire elements conforming to weightage of 4 are identified as noisy pixels, 4 weightage is detached from $W_{weights}$. Therefore, jump of 2 is perceived when migrating from weightage of 3 to 5 in W weights. Weights reassigning is done for diminishing number of repetitions.

Shape Feature Extraction

For every image patch of NDVI image created in the aforementioned step, set of shape features computation such as contour and skeleton features are done.

Dynamic Non-Linear Decreasing Strategy based Glowworm Swarm Optimization (DNDSGSO) based Feature Selection

Dynamic Non-linear Decreasing Strategy based Glowworm Swarm Optimization (DNDSGSO) is greatly utilized for extracted feature selection. Arbitrary deployment of glowworms swarm in GSO is performed in solution space (Oramus 2010);(Kaipaand Ghose 2017). It consists of luminescence quantity called luciferin in GSO agents similar to glowworms. Their current locations fitness is encoded through glowworms which are assessed through objective function, into a luciferin value that they transmist to their neighbors.The neighbors are identified by glowworm as well movement computation is done through adaptive neighborhood exploitation which is bounded above through its sensor range. A probabilistic mechanism is utilized for selecting every glowworm and there occurs movement towards if that neighbor has a luciferin value higher than its own. The glowworms swarm are enabled for partitioning into disjoint subgroups that converge on manifold optima of a specified multimodal function on the basis of local information and selective neighbor interactions.

The Process repetition takes place till termination condition is met through algorithm. The majority of individuals collect everywhere brighter glowworms now. It consist of six main phases in GSO: Glowworms' initialization, luciferin-update phase, neighborhood-select phase, moving probability-computer phase, movement phase, and decision radius update phase.

Glowworms' initialization:

Glowworms are regarded as an image features in this phase,which are primarily spread arbitrarily in the established fitness function space. The same amount of Lucifer is encompassed in glowworms. In addition, current iteration is fixed to 1. Here classification accuracy is regarded as fitness value.

Luciferin-update phase:

On the basis of fitness value (accuracy) besides preceding luciferin value, luciferin updating is done and its rule specified through

$$l_i(t+1) = (1-\rho)l_i(t) + \gamma Fitness x_i(t+1) \quad (4.3)$$

Where, $l_i(t)$ represents glowworm luciferin (feature) i at time t , ρ signifies luciferin decay constant ($0< \rho < 1$) , γ denotes luciferin enhancement constant, $x_i(t+1) \epsilon R^M$signifies glowworm (feature) location i at time in addition $Fitness x_i(t+1)$ characterizes fitness value at glowworm i's location at time $t+1$.

Neighborhood-Select Phase

Neighbors $N_i(t)$ of glowworm (feature) i at t time comprise of brighter ones which is given by

$$N_i(t) = \{j: \left|\left| d_{ij}(t) < r_d^i(t) \right|\right|;\ l_i(t) < l_j(t) \quad (4.4)$$

,$r_d^i(t)$ denotes variable local-decision domain, $d_{i,j}(t)$ signifies Euclidean distance amid features i in addition to j at time t.

Moving Probability-Computer Phase

A glowworm utilizes probability rule for moving towards other glowworms possessing higher luciferin level. The probability $p_{ij}(t)$ of glowworm (feature) i moving in the direction of its neighbor j can be quantified as trails:

$$p_{ij}(t) = \frac{l_j(t) - l_i(t)}{\sum_{k \in N_i(t)} l_k(t) - l_i(t)} \quad (4.5)$$

Movement Phase

Presume glowworm (feature)i picks glowworm (feature)j$\in N_i(t)$ with $p_{ij}(t)$; the discrete-time model of glowworm (feature) i movement is specified as

$$x_i(t+1) = x_i(t) + s(t)\left(\frac{x_j(t) - x_i(t)}{\left\| x_j(t) - x_i(t) \right\|}\right) \quad (4.6)$$

Here, s reprsents step size and $||.||$ denotes Euclidean norm operator

Decision Radius Update Phase

Glowworm (feature)i decision radius in every update is specified as trails:

$$r_d^j(t+1) = \min\{r_s, max\{0, r_d^j(t) + \beta(n_t - |N_i(t)|)\}\} (4.7)$$

Here, β represents constant, r_s denotes glowworm (feature) i sensory radius, andn_t signifies parameter for neighbor number control.

ENSEMBLE MODIFIED CONVOLUTIONAL NEURAL NETWORK (EMCNN)

Modified Convolutional Neural Network (MCNN) is one in which selected features are given as an input besides regarded as most potent deep networks encompassing multiple hidden layers accomplishing convolution in addition sub sampling for extracting low to high levels of input data features. Mostly, there are three layers in a network: convolution layers, subsampling or pooling layers, and full connection layers.

The features are given as input, output layer from where the system acquires trained output besides intermediate layers known as hidden layers. The proposed Modified Convolutional Neural Network (MCNN), optimization of features weight values for attaining precise outcomes.

Convolution layer

The convolution is performed for an input image with a kernel (filter) of size $a * a$. An output pixel is obtained through independent convolution of every input matrix block with kernel. In a similar way, output image features are produced through convolution of the input image in addition to kernel. Primarily convolution matrix kernel is termed as filter although output image features attained through convolution of kernel besides input images are termed as feature maps of size $i * i$.

CNN encompasses multiple convolutional layers, inputs in addition to outputs of next convolutional layers are the feature vector. It also comprises group of n filters in each convolution layer. These filter convolution is attained with input besides created feature maps (n*) depth is equivalent to number of filters. Every filter map is deliberated as explicit feature at a definite input image locality.

The l-th convolution layer output, signified as $C_j^{(l)}$, encompassing feature maps which is estimated by $C_i^{(l)} = B_i^{(l)} + \sum_{j=1}^{a_i^{(l-1)}} K_{i,j}^{(l-1)} * C_j^{(l)}$ (4.12)

Where, $B_i^{(l)}$ represents bias matrix ,$K_{i,j}^{(l-1)}$signifies convolution filter or kernel of size $a * a$ relating j-th feature map in layer $(l-1)$ with the i-th feature map in the alike layer. The output $C_i^{(l)}$ layer comprises of feature maps. In equation (4.14), first convolutional layer$C_i^{(l-1)}$ is input space, that is,$C_i^{(0)} = X_i$.

Feature map s generated through kernel. Nonlinear transformation of convolutional layer ouput is obtained by activation function application after convolution layer.

$$Y_i^{(l)} = Y(C_i^{(l)}) \qquad (4.13)$$

Where, $Y_i^{(l)}$ signifies activation function output in addition to $C_i^{(l)}$is the input that it receives.

Generally sigmoid, tanh, besides rectified linear units (ReLUs) are the various activation functions utilized here. ReLUs signified as $Y_i^{(l)} = max(0, Y_i^{(l)})$ are exploited in deep learning models due to benefit of diminished interaction and nonlinear effects. ReLU transforms output to 0 if it takes a negative input, while it yields the same input value if it is positive. Fastest training is one of the prominent advantages of this activation function in contrast to other functions since error derivative tends to be very trivial in saturating region and hence vanishing of weights update takes place which is termed as vanishing gradient problem.

REFERENCES

1. Bakhshipour, A and Zareiforoush, H, (2020). Development of a fuzzy model for differentiating peanut plant from broadleaf weeds using image features. Plant Methods, vol. 16, no. 1, pp. 1-16.
2. Barbedo, J.G.A, (2016). A novel algorithm for semi-automatic segmentation of plant leaf disease symptoms using digital image processing. Tropical Plant Pathology, vol. 41, no. 4, pp. 210-224.
3. Chen, B, Xing, L, Zhao, H, Zheng, N and Prı, J.C, (2016). Generalized correntropy for robust adaptive filtering. IEEE Transactions on Signal Processing, vol. 64, no. 13, pp. 3376-3387.
4. Chetty, S and Adewumi, A.O, (2013). Studies in swarm intelligence techniques for annual crop planning problem in a new irrigation scheme: case study. South African Journal of Industrial Engineering, vol. 24, no. 3, pp. 205-226.
5. Chuanlei, Z, Shanwen, Z, Jucheng, Y, Yancui, S and Jia, C, (2017). Apple leaf disease identification using genetic algorithm and correlation based feature selection method. International Journal of Agricultural and Biological Engineering, vol. 10, no. 2, pp. 74-83.
6. Collobert, R and Weston, J, (2008). A unified architecture for natural language processing: Deep neural networks with multitask learning, in Proceedings of the 25th international conference on Machine learning, pp. 160-167.
7. Collobert, R, Weston, J, Bottou, L, Karlen, M, Kavukcuoglu, K and Kuksa, P, (2011). Natural language processing (almost) from scratch. Journal of machine learning research, 12(ARTICLE), pp. 2493-2537.
8. Dash, P.B, Naik, B, Nayak, J and Vimal, S, (2022). Socio-economic factor analysis for sustainable and smart precision agriculture: An ensemble learning approach. Computer Communications, vol. 182, pp. 72-87.
9. De Castro, A.I, Torres-Sánchez, J, Peña, J.M, Jiménez-Brenes, F.M, Csillik, O and López-Granados, F, (2018). An automatic random forest-OBIA algorithm for early weed mapping between and within crop rows using UAV imagery. Remote Sensing, vol. 10, no. 2, pp. 285.
10. Eerens, H, Haesen, D, Rembold, F, Urbano, F, Tote, C and Bydekerke, L, (2014). Image time series processing for agriculture monitoring. Environmental Modelling & Software, vol. 53, pp. 154-162.