Smart Agricultural Technology 5 (2023) 100288

Contents lists available at ScienceDirect

Smart Agricultural Technology

journal homepage: www.journals.elsevier.com/smart-agricultural-technology

Developing an extreme learning machine based approach to weed segmentation in pastures ☆

Jonathan Ford *, Edmund Sadgrove, David Paul

School of Science and Technology, University of New England, Armidale, Australia

ARTICLE INFO

Editor: Stephen Symons

Keywords:
Extreme learning machine
Semantic segmentation
Weed detection
Computer vision
Precision agriculture
Feature extraction

ABSTRACT

Effective weed management in pastures is critical for maintaining the productivity of grazing land. Autonomous ground vehicles (AGVs) are increasingly being considered for weed localization and treatment in agricultural land. Weeds, however, can be difficult to distinguish from background plants, due to similarities in colour, shape and texture. While deep learning approaches can be used to solve the localization issue, they are computationally expensive, and require a large volume of training images in order to combat overfitting. In this paper we present a novel Extreme Learning Machine based network for segmenting weeds from the background pasture. The proposed method utilizes a combination of LBP, HOG and colour features, and is tested on four small datasets, achieving a high mean Intersection over Union of 87.1, 79.5, 81.6 and 87.6 for Bathurst burr, horehound, thistle and serrated tussock respectively.

1. Introduction

The world population is projected to reach 9.7 billion by 2050 [1]. With the continued growth in population, it is becoming increasingly important to maximize agricultural yield using the resources available, including arable land, chemicals, and manpower. Precision agriculture, driven by the artificial intelligence revolution, is an essential component in the solution to ensuring that world agriculture maintains sufficient output to feed a growing population [44]. One important application of artificial intelligence in agriculture is the use of computer vision to detect and classify weeds in crop fields and pastures [27]. Effective weed detection will allow the selective spraying of undesirable plants, while avoiding damage to other plants in the field, and reducing the amount of herbicide used, which reduces both cost and the potential harmful effects of chemicals on human health and the environment [27].

The majority of weed detection studies found in the literature involve crop fields, including sugarbeet [39,30,29], soy [14,47], lettuce [38], carrot [25,22,32], and grain crops [26,49]. In such an environment, the segmentation of vegetation from background soil and crop stubble can be carried out using a vegetation index, such as Excess Green (ExG). Such segmented vegetation then needs to be analysed to determine whether it is part of the crop plant, or a weed.

Fewer studies have been carried out on the detection of weeds in pastures. However, there has been considerable research on the task of locating dockleaf in pastures in Europe [5,45,51,24,11], with an accuracy of 83.4% achieved using conventional machine learning [5], and 95.6% [45] to 96.9% [51] for deep learning. It should be noted that the above studies each used different datasets, making it difficult to make a meaningful comparison between them. In Australia, one study looked at the detection of serrated tussock, a weed of national interest, on a dataset collected in farmland in northern NSW, using unsupervised feature learning, k-means clustering and a linear classifier to achieve an accuracy of 92.9% [20]. In another study a fusion of several features was passed to a SVM to classify images of serrated tussock, Chilean needlegrass and native grass clumps from south east Australia, with a best accuracy of 80.9% achieved using a combination of LBP and HOG features [16].

In other work, Olsen et al. [37] developed an extensive dataset comprising images of eight weed species of national significance from North-East Australia, and produced a baseline accuracy of 95.1% and 95.7% using Inception-v3 [46] and ResNet-50 [18] deep learning mod-

☆ The first author was supported by a Commonwealth of Australia RTP scholarship, with a top-up scholarship from the "Novel Autonomous Robotic Weed Control to Maximise Agricultural Productivity" CRC. The authors would like to thank the NSW Department of Primary Industries and Agent Oriented Software for providing the dataset.

* Corresponding author.
E-mail addresses: jford3@myune.edu.au (J. Ford), esadgro2@une.edu.au (E. Sadgrove), dpaul4@une.edu.au (D. Paul).

https://doi.org/10.1016/j.atech.2023.100288
Received 28 March 2023; Received in revised form 13 July 2023; Accepted 14 July 2023
Available online 25 July 2023

els respectively. These results were further improved by Hu et al. [19] who used a graph-based deep learning network to achieve an accuracy of 98.1%. On the other hand, Calvert et al. achieved an accuracy of 98.1% on a dataset of Harrisia cactus growing in dry Eastern Australian rangelands, using ResNet-50 [8]. They went on to test their robotic spot spraying vehicle, AutoWeed, in the field, achieving a weed knock-down rate of 95.0% to 96.3% compared to a rate of 98.3% that was achieved using manual application of herbicide [9].

In contrast, Sadgrove et al. [40–43] utilised a shallow ANN with a partially connected hidden layer, which was trained using a variation of the Extreme Learning Machine algorithm (ELM, discussed in the Methods section) called the Colour Feature ELM (CF-ELM), to classify images of thistle and horehound. These models were able to obtain a precision of 82% - 98% and recall of 84% - 94% for thistle, and a precision of 84% and recall of 97% for horehound. In other agricultural studies, Maimaitijiang et al. [31] used an ELM to estimate biochemical and biophysical parameters, such as chlorophyll content, nitrogen content and biomass from aerial images, and Aqel et al. [4] use an ELM to classify four different plant leaf diseases.

The ELM has also been used in a variety of computer vision problems aside from weed detection. In one study [2], a multilayer, ELM-based autoencoder was used to segment hyperspectral satellite imagery into land-use categories. Dixit and Hedge [13], also working with hyperspectral satellite imagery, developed an ELM model to classify vegetation cover as tree, shrub or grassland. Dewi et al. [12] used an ELM model to classify leaves of different varieties of Patchouli. Turkoglu and Hanbay [50], similarly used an ELM model to classify leaves of different species of plant, testing their approach on multiple publicly available datasets.

In spite of the studies discussed in the previous paragraphs, the use of ELMs for weed classification in agriculture remains largely unexplored, and those studies that have been done have avoided the more complex problem of segmentation, in which the specific location of weeds within an image needs to be determined. In order to explore this space further, we designed and implemented an ELM-based model which inputs hand crafted image features, and outputs a fine-grained weed heat-map. So that the model can handle large, and arbitrarily sized images, the model is passed, like a convolutional filter, across the feature cube extracted from the image. This generates multiple heatmaps for different image regions, which are then combined into a single segmentation result for the entire image. Using the ELM as a convolutional filter results in a significant reduction in the computational cost of feature extraction, as the extraction need only be completed in a single pass (for the entire image), as opposed to separately for multiple, overlapping frames, as would be required in the sliding window approach. This model has been developed to be part of the detect-and-spray pipeline of an autonomous ground vehicle, allowing site-specific management of weeds in pastures, thereby reducing herbicide use and the manpower required to control weeds.

The main contributions of this paper are:

1) Use of a novel ELM-based network to produce a fine-grained segmentation of several datasets containing weeds of interest in southeast Australian pastures.
2) Presentation of the results of a rigorous comparison of different LBP and HOG hyperparameters, used on several weed datasets collected from southeast Australia.

The remainder of the paper will be organised as follows. Section 2 provides a brief review of related research. Section 3 describes the approach used in this study. Section 4 gives the results of the study. Section 5 provides an explanation of how this study fits into the broader research landscape. Finally section 6 provides some possible future directions leading on from this work.

2. Related work

In this section, we will discuss previous studies on weed detection in crop-lands and pastures, focusing on studies which have used Local Binary Patterns (LBPs), Histogram of Oriented Gradients (HOG) and colour features. We will also present a review of studies involving the fine-grained segmentation of pictures into regions of crops and weeds.

2.1. Feature extraction

LBPs have been at the core of many conventional classifiers used for the classification of weeds in crop fields [28,14,35,26] and pastures [5,51,16]. While some studies either do not specify their choice of hyperparameters or use default values,[1] there are a few cases where a more extensive comparison of different LBP descriptors has been carried out, as summarized in Table 1. In one study [30], the LBP features were extracted from multiple colour bands including a vegetation index (NDVI), and the gradient and double gradient of each band was also considered. In another study [16], LBP features were extracted from a total of 27 bands from both the near infrared and visible spectrum, and concatenated to form the final feature vector. On the other hand, two studies considered multiple combinations of radius and number of neighbours in their analysis [5,35].

HOG features are also commonly used in plant classification algorithms [38,14,16,23]. The above cited papers, however, do not specify their choice of hyperparameters for HOG features, such as the number of orientation bins, or whether signed or unsigned orientation is used. Based on the original HOG paper of Dalal and Triggs [10], performance of HOG features for pedestrian identification improves as the number of orientation bins is added, up to a total of roughly 9 bins for unsigned orientation, however the optimal number of bins for a weed classification problem may be different to this.

A range of colour features has been used in previous studies on weed classification and segmentation in agriculture. These features include the colour histogram [3], as well as statistical features such as the mean, standard deviation, minimum, maximum, range, median, skewness, kurtosis and entropy, either applied to different colour channels [14], or a mixture of colour channels and vegetation indices [30,21].

2.2. Image segmentation

The majority of studies on the fine-grained segmentation of agricultural images use convolutional neural networks, such as SegNet [6], fully convolutional networks [34,29], or custom encoder/decoder networks [22,33,32,39].

The use of conventional machine learning in the segmentation of agricultural imagery is uncommon. However, in one study, images of sugarbeet seedlings and weeds were broken up into key-points, corresponding to 10×10 pixel squares, and each keypoint was classified, based on features extracted from the 80×80 pixel neighbourhood, and passed to a Random Forest [28]. The same authors went on to consider an object-based approach, where an object corresponds to an individual connected component of plant pixels, and a combination of the object and key-point based approaches [30]. Where the connected component corresponds to a single class, the object-based approach can be used for classification, but if the connected component consists of both classes, the object-based method will return an intermediate result (i.e. part way between crop and weed). In the latter case, the key-point based approach can then be used to further segment the component into regions consisting of crop plants, and regions consisting of weeds.

The method discussed in the preceding paragraph, may work well in crop fields, where plants can easily be separated from background

[1] Here we take default values to be 8 neighbours and a radius of 1.

Table 1
LBP hyperparameters (r = radius, n = number of neighbours) and colour channels (Grey = greyscale; NIR = near infrared; R, G and B from the RGB colour space; NDVI = normalized difference vegetation index; H, S and L from the HSL colour space where the value for R is replaced by NDVI) used in previous studies on weed detection.

Study	Hyperparameters	Colour Channel	Other
Dockleaf in pasture [5]	$r = 1 \dots 16$ $n = 2 \dots 30$	Grey	
Canola, radish and barley [35]	$r = 1, n = 8$ $r = 2, n = 16$ $r = 3, n = 24$	Grey	Upright, and Uniform
Tussock and needlegrass [16]	$r = 2, n = 8$	16 visible and 11 NIR bands	
Sugarbeet [30]	$r = 1, n = 8$	NDVI, G, B, H, S and L	Also uses the gradient and Laplacian of each channel

Table 2
List of hyperparameters considered for features used in this study.

Feature	Hyperparameter	Range of Values	MATLAB argument
HOG	Number of Bins*	4,6,9,12,15	`NumBins`
HOG	Signed Orientation	T/F	`UseSignedOrientation`
LBP	Number of Neighbours	8, 16, 24**	`NumNeighbors`
LBP	Radius	1, 2, 3, 4, 6, 8, 10	`Radius`
LBP	Rotational Invariance	T/F	`Upright`

* When signed orientation is used, the number of bins is effectively doubled.
** The following combinations of radius (r) and number of neighbours (n) were not considered: (r = 1,n = 16), (r = 1,n = 24), (r = 2,n = 24).

Table 3
List of colour spaces/indices used in this study.

Name	Description	Formula
Grey	Greyscale	$\frac{1}{3}(R+G+B)$
ExG	Excess Green	$2\times G - R - B$
ExR	Excess Red	$1.4\times R - G$
ExGR	Excess Green minus Excess Red	$ExG - ExR$

soil using a vegetation index, such as NDVI. In the more complex environment encountered in pastures, where we have green weeds against a backdrop of green pasture elements (the so called green on green problem), another approach is needed.

3. Methods

All coding was implemented in MATLAB using built in functions wherever possible, including `extractHOGFeatures` for HOG features, and `extractLBPFeatures` for LBP features.[2] The above MATLAB functions allow for a number of different customizable options. Accordingly, tests were carried out to determine the appropriate hyperparameters to use. The list of all hyperparameters tested in this work is given in Table 2. In addition to this, we used a range of vegetation indices as well as grayscale to produce a single colour channel image to be inputted into the feature extraction functions, as shown in Table 3. All tests were carried out on one of two Centos 7 systems with 192 Gb of memory and 32 Intel 2.5 Ghz CPU (hyperthreaded) cores.

[2] Documentation for these functions can be found at https://au.mathworks.com/help/vision/ref/extracthogfeatures.html and https://au.mathworks.com/help/vision/ref/extractlbpfeatures.html respectively.

3.1. Local binary patterns

LBP features [36] are a texture analysis method in which the intensity of a pixel is compared with its neighbours. Each neighbour is either classified as darker or brighter than the central pixel, meaning that the algorithm generates one bit of information per neighbour, as shown in equations (1) and (2).

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

$$LBP(g^*) = \sum_{p=0}^{P-1} s(g_p - g^*)2^p \tag{2}$$

Here, $s(x)$ is a sign function, $LBP(g^*)$ is the LBP descriptor for pixel g^*, P is the number of neighbours, and g is the grey level of the pixel. The image can then be broken down into rectangular blocks, and an LBP histogram (LBPH) calculated for each block. If P is set to 8, then the LBPH will have 256 bins, one for each of the possible descriptor values. In MATLAB, the `extractLBPFeatures` method automatically combines all patterns that are not "uniform", as described by [36], into a single bin, meaning that the number of bins will be $P*(P-1)+3$, or $P+2$ depending on whether rotational invariance is used. For this study, both rotationally invariant and rotationally variant features were considered, and a range of different values for the number of neighbours, and the radius of the neighbours were tested. Fig. 1 gives an example of LBP features extracted from image patches of an image taken from the tussock dataset.

3.2. Histogram of oriented gradients

HOG descriptors were originally designed as a solution to the problem of detecting pedestrians in images [10]. In HOG feature extraction, the image is divided into overlapping blocks of cells, and the occurrence of gradient (or edge) orientation is calculated. The orientation is classified into equal bins from 0° to 180° for unsigned, and 0° to 360° for

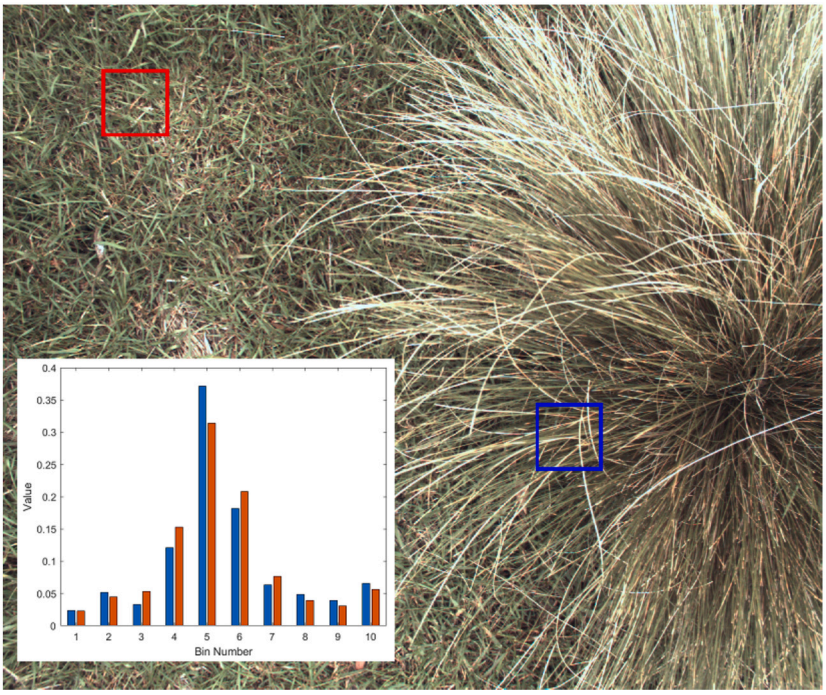

Fig. 1. Example LBP features extracted from image patches containing tussock (blue), and background pasture (red). Hyperparameters used were radius = 1, number of neighbours = 8, rotational invariance = True.

signed orientation. The number of orientation bins can be customized in MATLAB, and so this study considered a range of possible bin numbers, as well as testing both signed and unsigned orientation.

3.3. Colour features

In addition to the textural features discussed in the previous two subsections, colour features were also considered, including the mean, standard deviation, minimum, maximum, range, median, skewness and kurtosis of each channel, using either the RGB or YCbCr colour spaces.

3.4. Datasets

RGB Images were collected by the NSW Department of Primary Industries (DPI) from pastures in a range of locations across south east Australia. Four separate datasets were chosen for testing as shown in Table 4. Images were taken using a Mako G507 camera mounted on an unmanned ground vehicle, and facing straight downwards. The camera was at a height of 1200 mm. Images were annotated under supervision of experts from DPI, by drawing a polygon around the weeds of interest, using the VGG image annotator [15]. This provided a good compromise between the accuracy of the annotation, and the time taken to annotate each image. Care was taken to ensure that there was no overlap between any two images in a dataset, in order to avoid polluting the test set with data from the training set.

For the Bathurst burr, horehound and thistle datasets, tests were carried out using 5×10 fold cross-validation. For the tussock dataset, there were sufficient images to produce 10 entirely independent train/test split, with each image used exactly once, either in a single training set or a single test set. Statistical significance was determined using a corrected repeated k-fold cv test [7] for the Bathurst Burr, Horehound and Thistle datasets. For the Tussock dataset, the train/test sets were entirely independent, so a standard t-test was used. During each iteration of cross validation, the train/test splits were identical for each tested feature, meaning that a two-tailed, paired t-value was used to determine significance, with $\alpha = 0.05$, and 49 (or 9 for the Tussock dataset) degrees of freedom.

For all datasets, stratification was used when determining the splits. Specifically, each image was placed in one of two pseudo-classes, based on the percentage coverage of weed in that image. For composite datasets, with multiple sub-sets taken at different locations and/or times, each sub-set was split into pseudo-classes separately, yielding $2 \times N$ pseudo-classes, where N is the number of sub-sets in the dataset.

In order to see if a fusion of features could improve on the use of a single feature, a set of features was generated for each dataset, consisting of the two colour features, the best LBP feature for each choice of radius, and the best HOG feature for each choice of number of bins, giving a total of 14 features. Starting with the best individual feature, as determined by the individual feature tests, additional features were added incrementally, choosing the feature that produced the best results during each iteration.

3.5. Extreme learning machine

The Extreme Learning Machine (ELM) is a fast learning algorithm proposed by Huang and Seiw [17] for a single hidden layer feed forward neural network. In the ELM, the hidden layer weights are randomly assigned, and remain unchanged during the training process. The hidden layer outputs **H**, can be found using equation (3), where **W** (**b**) is the hidden layer weights matrix (biases vector), and g is the activation function.

$$\mathbf{H} = g(\mathbf{WX} + \mathbf{b}) \tag{3}$$

The final layer weights $\hat{\beta}$ are then calculated using the closed form solution given in equation (4), where † is the Moore-Penrose pseudo-inverse.

$$\hat{\beta} = \mathbf{H}^{\dagger}\mathbf{T} \tag{4}$$

While the ELM algorithm described in the paragraph above uses random hidden layer weights, further studies indicated that hidden layer weights which had been computed from the training data in some way, such as the constrained ELM [52] and computed input weights [48], produced superior results. In this study, we use the constrained differ-

Table 4
List of datasets used in this study.

Dataset	Weed Species	Location	Date of Collection
Bathurst Burr	Bathurst Burr (*Xanthium spinosum*)	Ferndale (34.636S 147.833E)	4-2-2022
Horehound	Horehound (*Marrubium vulgare*)	Yellangelo (34.861S 149.169E)	13-10-2021 23-11-2021
Thistle	Scottish Thistle (*Onopordum acanthium*)	Jugiong (34.860S 148.297E)	19-10-2021
	Saffron Thistle (*Carthamus lanatus*)	Binalong (34.690S 148.644E)	22-11-2021
		Yellangelo (34.861S 149.168E)	23-11-2021
		Clarkefield (37.480S 144.765E)	20-01-2022
Tussock	Serrated Tussock (*Nassella trichotoma*)	Yellangelo (34.861S 149.169E)	13-10-2021

Table 5
Syntax and operations used in this paper.

Syntax and methods	Explanation
$*$	Convolution
$[\mathbf{X}, \mathbf{x}]$	Join matrix with vector in horizontal direction
$\mathbf{X}[:, k]$	Submatrix of X with all rows, and only the kth column(s).
$\oslash$	Element-wise (Hadamard) division
$\texttt{sample}^{+}_{-}$	Random sampling of positive and negative pair
`init`	Method for initializing weights
`flatten`	Operation for converting matrix into vector
`unflatten`	Operation for converting vector into square matrix
`rand`	Operation for generating a random number

ence (CD-weights) method of [52], and the sigmoid activation function, which is shown in equation (5).

$$g(x) = \frac{1}{1 + e^{-x}} \tag{5}$$

3.6. Proposed model

Conceptually, it is beneficial for us to treat both the feature extractor and the ELM classifier as convolutional kernels. The proposed workflow is given below in a "procedural" format. In MATLAB, however, most steps can be simplified using matrix (or tensor) manipulations. We start with the following components:

- $\mathbf{F_f}$: the convolutional kernel containing the feature extractor, parameterized by the feature extractor $\mathbf{f}$
- $\mathbf{E}_{\mathbf{W},\mathbf{b},\hat{\beta}}$: the convolutional kernel containing the ELM classifier, parameterized by the weights and biases of the ELM.
- $\mathbf{M}$: the operation for unflattening and combining the output from $\mathbf{E}_{\mathbf{W},\mathbf{b},\hat{\beta}}$

We also have a set of input images and corresponding ground truth weed heatmaps: $(\mathbf{I}, \mathbf{L}) \in \mathcal{I}$, and a training and validation set, $\mathcal{I}_t$ and $\mathcal{I}_v$, where $\mathcal{I}_t \cup \mathcal{I}_v = \mathcal{I}$ and $\mathcal{I}_t \cap \mathcal{I}_v = \emptyset$. An explanation of the syntax used in this section is provided in Table 5.

To train the ELM, we first calculate the input and target matrices, X and T, using Algorithm 1.

Algorithm 1 Generate training matrices.

Require: $\mathcal{I}_t$
Require: $\mathbf{F_f}$

$\mathbf{X} \leftarrow \mathbf{0}_{0\times 0}$
$\mathbf{T} \leftarrow \mathbf{0}_{0\times 0}$
for $(\mathbf{I}, \mathbf{L}) \in \mathcal{I}_t$ **do**
  $\mathbf{J} = \mathbf{I} * \mathbf{F_f}$
  for $\mathbf{J_p} \in \mathbf{J}$ **do**
    $\mathbf{L_p} \leftarrow$ corresponding $\hat{\mathbf{L}} \in \mathbf{L}$
    $\mathbf{X} \leftarrow [\mathbf{X}, \texttt{flatten}(\mathbf{J_p})]$
    $\mathbf{T} \leftarrow [\mathbf{T}, \texttt{flatten}(\mathbf{L_p})]$
  end for
end for
return $(\mathbf{X}, \mathbf{T})$

Now we can calculate the parameters of the ELM using Algorithm 2. With respect to hidden layer parameter initialization, each sample in $\mathbf{X}$ is considered positive (respectively negative), if its corresponding target in $\mathbf{T}$ is composed entirely of 1s (respectively 0s). If the target is composed of a mixture of 1s and 0s, then it is ignored. To combat imbalance in the dataset, negative samples (as defined in the preceding sentence) are randomly removed from the training matrices, with a probability of p equal to 0.3, 0.3, 0.7 and 0 for the Batthurst Burr, Horehound, Thistle and Tussock datasets respectively.

Algorithm 2 Generate ELM weights.

Require: $\mathbf{X}$
Require: $\mathbf{T}$
Require: $p \in [0, 1]$

$\mathbf{W} \leftarrow \mathbf{0}_{0\times 0}$
$\mathbf{b} \leftarrow \mathbf{0}_{0\times 0}$
for $i \leftarrow 1$ to m **do**
  $(x^+, t^+), (x^-, t^-) \leftarrow \texttt{sample}^{+}_{-}(\mathbf{X}, \mathbf{T})$
  $(\mathbf{w}, b) \leftarrow \texttt{init}((x^+, t^+), (x^-, t^-))$
  $\mathbf{W} \leftarrow [\mathbf{W}, \mathbf{w}]$
  $\mathbf{b} \leftarrow [\mathbf{b}, b]$
end for
keep $\leftarrow [\mathbf{t} \neq \mathbf{0} \vee \texttt{rand}(0, 1) > p]$ for $\mathbf{t} \in \mathbf{T}$
$\hat{\mathbf{X}} \leftarrow \mathbf{X}[:, \text{keep}]$
$\hat{\mathbf{T}} \leftarrow \mathbf{T}[:, \text{keep}]$
$\mathbf{H} \leftarrow g(\hat{\mathbf{X}}\mathbf{W} + \mathbf{b})$
$\hat{\beta} \leftarrow \mathbf{H}^{\dagger}\hat{\mathbf{T}}$
return $(\mathbf{W}, \mathbf{b}, \hat{\beta})$

Using the trained ELM, we can calculate the predicted heatmap for an image, using Algorithm 3. The unflattening operation $\mathbf{M}$ is given in Algorithm 4.

Table 6
All Datasets: Mean IoU for colour features. The best result for each dataset is in **bold**. Results shown are the mean ± standard deviation, for a 5×10 fold cross validation (Bathurst burr, horehound and thistle datasets), or for a set of 10 entirely independent train/test splits (tussock). Results which are not significantly different from the best result in the column are labelled with a †.

Colour Channel	Dataset			
	Bathurst Burr	Horehound	Thistle	Tussock
RGB	74.9 ± 9.9	$\mathbf{70.8 \pm 6.5}^{\dagger}$	71.6 ± 2.5	$78.4 \pm 5.2^{\dagger}$
YCbCr	$\mathbf{79.5 \pm 7.7}^{\dagger}$	$69.6 \pm 6.4^{\dagger}$	$\mathbf{74.3 \pm 3.5}^{\dagger}$	$\mathbf{79.4 \pm 4.1}^{\dagger}$

Algorithm 3 Calculate predicted heatmap.

Require: $\mathbf{I}$
Require: $\mathbf{F_f}$
Require: $\mathbf{E_{W,b,\hat{\beta}}}$
$\mathbf{J} = \mathbf{I} * \mathbf{F_f}$
$\mathbf{K} = \mathbf{J} * \mathbf{E_{W,b,\hat{\beta}}}$
$\mathbf{Y} = \mathbf{M}(\mathbf{K})$
return $\mathbf{Y}$

Algorithm 4 Unflatten and combine output of ELM.

Require: $\mathbf{K}$
$\mathbf{Y} \leftarrow \mathbf{0}$
$\mathbf{C} \leftarrow \mathbf{0}$
for $\mathbf{k_p} \in \mathbf{K}$ **do**
  $\hat{\mathbf{k}}_\mathbf{p} \leftarrow \texttt{unflatten}(\mathbf{k_p})$
  $\mathbf{Y_p} \leftarrow$ corresponding $\hat{\mathbf{Y}} \in \mathbf{Y}$
  $\mathbf{C_p} \leftarrow$ corresponding $\hat{\mathbf{C}} \in \mathbf{C}$
  $\mathbf{Y_p} \leftarrow \mathbf{Y_p} + \hat{\mathbf{k}}_\mathbf{p}$
  $\mathbf{C_p} \leftarrow \mathbf{C_p} + \mathbf{1}$
  $\mathbf{Y} \leftarrow \mathbf{Y} \oslash \mathbf{C}$
  return $\mathbf{Y}$
end for

In this study, all images were of size (with height dimension first) $2016 \times 2400 \times 3$. The feature extraction kernel used a size of 48×48, and stride of 48×48. For HOG features, the cell size was set at 24×24, meaning that each block consisted of 2×2 cells. The ELM kernel used a size of 4×4 and stride of 1×1, corresponding to an image patch size of 192×192 and 48×48 respectively. The heatmap was downsampled by a factor of 16, meaning that the ELM outputted a 12×12 flattened heatmap vector. If a by-pixel segmentation was required, then the heatmap could further be upsampled, for example, using bilinear interpolation. However, for the purpose of determining when to switch herbicide spray nozzles on and off, a by-pixel segmentation is unnecessary, because the nozzles will have a limited accuracy. For this reason, we leave the heatmap downsampled in this study.

4. Results

The efficacy of individual features was first tested, according to the methodology in Section 3. The results are presented in Table 6 for colour features, Tables 7, 9, 11 and 13 for LBP features and Tables 8, 10, 12 and 14 for HOG features. For these tests, the number of hidden layer neurons was varied from 100 to 1000, with only the best result shown for each feature. In most, but not all cases, this corresponded to a hidden layer size of 1000. Extensions of the figures listed in this section are provided in the supplemental material, including the standard deviation of all results, and the number of neurons which produced the best result for each feature.

Out of the two colour spaces used for colour feature extraction, YCbCr performed best for three datasets (Bathurst Burr, Thistle and Tussock), and was not significantly difference from RGB in the other dataset (Horehound). For LBP features, the best choice of colour channel was dependent on the dataset. For Bathurst Burr, ExGR performed best for all choices of radius, except for $r = 1$. By contrast the best results for all choices of radius for horehound were achieved using the grey colour channel. For tussock and thistle, the grey and ExG colour channels produced the best results, depending on choice of radius. The choice of colour channel was also important for HOG features, with the best results for thistle and horehound being ExG, and for tussock being grey. By comparison, the best results for Bathurst burr was achieved using grey or ExGR, depending on the choice of number of bins. In terms of the choice of number of bins, 4 bins almost always produced inferior results, whereas there was often no statistically significant different between the choice of bin numbers ranging from 6 to 15.

It should be pointed out that there was considerable variance in the results for individual features, in particular for the Bathurst burr and horehound datasets (SD as high as 12.6% and 9.2% respectively). This is likely due to the fact that there were only a small number of images in these datasets, and some images were considerably harder than others to segment, due to plants being run over and flattened by the AGV, or heavily occluded by other vegetation. The use of identical train/test splits for each tested feature, however, allowed us to still demonstrate statistically significant results for some features, because the individual features tended to be good or bad at the same train/test splits.

The best result by dataset and feature type (colour, LBP or HOG) is presented in Table 15. As can be seen, LBP produced the best features for three out of four datasets, while colour features were best for the thistle dataset. By comparison, HOG features performed much worse for three out of four datasets (e.g., up to 22.2% worse than the best LBP feature, in the case of the Bathurst burr dataset). For the tussock dataset, the performance of the best HOG feature was only 7.6% worse than the best LBP feature. We suggest that the reason that HOG features performed better for this dataset is because the tussock plants are arranged with many blades of grass aligned in a rough radial pattern around the centre of the plant, resulting in a more pronounced feature value for particular bins, as compared to the more haphazard arrangement of grass blades in the background pasture.

Fig. 2 displays the results of the feature fusion tests. For these tests, the number of hidden layer neurons was varied from 1000 to 3000, with only the best result shown for each feature fusion. In most, but not all cases, this corresponded to a hidden layer size of 3000. As can be seen, the results improved as features were added, but plateaued quickly, and even started to go back down as the fusion became larger. The improvements achieved through the fusion, as compared to the best individual feature, varied from 4.1% for the tussock dataset, to 7.3% for the thistle dataset. Table 16 gives the order of features added to the fusion for each dataset. As can be seen, the best results were achieved using a combination of LBP, HOG and colour features for the Bathurst burr and horehound datasets. By comparison, the best result for the thistle and tussock datasets did not include any HOG or colour features respectively. The extraction of multiple features from an image can be computationally expensive, and so the smallest fusion which is not significantly different from the best result can be seen as a good candidate for a compromise between accuracy and speed. For Bathurst burr, horehound and thistle, this fusion consisted of two LBP features and one colour feature, whereas for tussock, it consisted of three LBP

Table 7
Bathurst Burr Dataset: Mean IoU for LBP features for various choices of colour channel, rotational invariance, number of neighbours, and radius. Feature parameters are given in format ch-r-nn, where ch is the colour channel or index, r is rotational [v]ariant or [i]nvariant, and nn is the number of neighbours. The best result for each set of three rows (which corresponds to fixing the colour channel and rotational invariance) is in **bold**, for each column (which corresponds to fixing the radius) is underlined, and overall is *italicized*. Results shown are the mean for a 5×10 fold cross validation. Results which are not significantly different from the best result in the set of three rows (column) are labelled with a * (†).

Parameters	Radius						
	1	2	3	4	6	8	10
grey-i-8	63.6†	67.1†	64.2	64.9	65.9†	63.4	61.5
grey-i-16	-	70.7†	70.4	71.6†	72.3*†	69.7†	65.9†
grey-i-24	-	-	72.3†	**74.2***†	73.8*†	72.5*†	68.1†
grey-v-8	64.8†	70.4*†	70.0	69.4*	66.7†	63.2	61.5
grey-v-16	-	71.7*†	**72.2***	70.5*†	64.8	60.2	58.9
grey-v-24	-	-	69.6	70.6*†	64.9	61.9	58.7
exg-i-8	**70.9***†	69.5*†	68.2*	67.5*†	63.4†	63.6†	64.4†
exg-i-16	-	68.7*†	67.2	67.7†	66.3†	66.5†	66.8†
exg-i-24	-	-	67.0	67.5†	66.6†	67.2†	66.4†
exg-v-8	**72.3***†	66.8†	64.9	66.4†	62.0	61.0†	61.3†
exg-v-16	-	68.2†	62.0	62.8	60.6	60.1	59.9
exg-v-24	-	-	61.8	64.4	60.4	60.3†	59.6
exr-i-8	67.1*†	60.8	62.9	68.4*	64.7	60.6	55.3
exr-i-16	-	62.1	67.9	70.2*	67.6†	66.4†	60.9
exr-i-24	-	-	68.8*	**72.0***†	68.4†	67.7†	63.1†
exr-v-8	67.2†	65.9	62.7	**72.1***†	65.3	61.0	56.5
exr-v-16	-	65.2	67.2	69.9*	67.1†	68.2†	60.0
exr-v-24	-	-	65.7	67.3	69.6*†	65.1†	64.7†
xgr-i-8	67.9†	68.7	68.2	71.9	66.3	63.1	57.9
xgr-i-16	-	74.1†	80.2*†	76.8†	73.2	73.4†	69.8
xgr-i-24	-	-	***81.6****†	78.7†	77.2†	73.0†	72.9†
xgr-v-8	69.4†	72.2	70.1	78.4*†	70.6	66.1	59.4
xgr-v-16	-	76.1†	**80.0***†	75.9	72.8	73.4†	69.3
xgr-v-24	-	-	79.7*	77.5†	75.4	72.2†	71.5†

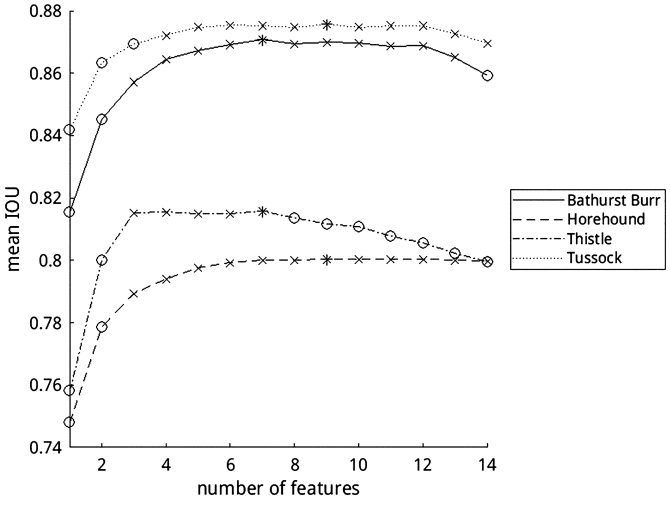

Fig. 2. Results for a fusion of features for all datasets. The data-point marked with a '$*$' is the best result for that dataset. Datapoints marked with an 'o' ('x') are (are not) significantly different from the best result for that dataset ($\alpha = 0.05$). Best result for Bathurst burr was 87.1 $\pm$3.9, for horehound was 79.5 $\pm$4.8, for thistle was 81.6 $\pm$2.1 and for tussock was 87.6 $\pm$1.9.

features, and one HOG feature. Fig. 3 gives example outputs of our model. In these images, the feature fusions which produced the best results for each dataset have been used. The model was able to segment some images very well, but other images posed significant challenges due to heavy occlusion (in the case of horehound), or the presence of many small weed plants against complex background elements (in the case of thistle).

Depending on the choice of hyperparameters, feature extraction for LBP feature took between 0.266 s and 0.848 s, for HOG features between 0.080 s and 0.090 s, and for colour features roughly 0.367 s. Based on CPU usage, it would appear that the matlab `extractLBPFeatures` function does not support parallelization, whereas the `extractHOGFeatures` function does, which could explain the difference in time taken for these methods. Excluding feature extraction, the time taken to process an image by the network and generate a mask varied depending on the dimensionality of the feature used, and the number of hidden layer neurons in the model. Times for a model with 100 neurons ranged from 0.001 s to 0.014 s, and for a model with 1000 neurons ranged from 0.002 s to 0.015 s. This means that the majority of the time was spent on feature extraction. In principle, it should be possible to speed up feature extraction through the development of a GPU implementation.

5. Discussion

This paper has presented a novel method of using a single hidden layer feedforward neural network trained with the ELM algorithm to produce a fine-grained segmentation of an image. The method has been tested on four datasets of pastoral lands in southeast Australia, and has demonstrated good performance, with a best mIOU of 79.5% to 87.6% achieved depending on the dataset. The horehound and thistle datasets proved to be the most difficult to segment. A likely explanation for this is that some plants in the horehound set were heavily occluded by other plants, whereas some of the thistle plants were small, and difficult to distinguish from the background. LBP features performed best for three out of four datasets. This strong performance compared to other feature extraction methods is consistent with the findings of other studies where multiple features are compared (see for example

Table 8
Bathurst Burr Dataset: Mean IoU for HOG features for various choices of colour channel, signed orientation and number of bins. The best result for each row is in **bold**, for each column is underlined, and overall is *italicized*. Results shown are the mean for a 5×10 fold cross validation. Results which are not significantly different from the best result in the row (column) are labelled with a * (†).

Parameters	Number of Bins				
	4	6	9	12	15
grey-u	**58.9***†	58.0*†	57.9*†	56.3†	55.8†
grey-s	53.8	***60.0****†	59.8*†	59.2*†	58.2†
exg-u	41.2	**46.3***	46.3*	45.6*	45.1*
exg-s	36.5	43.8*	**44.3***	44.0*	43.7*
exr-u	42.9	47.6	**51.1***	50.8*	50.5*
exr-s	40.0	46.1*	49.1*	50.2*	**50.6***
xgr-u	51.6†	57.9*†	59.5*†	**59.7***†	59.4*†
xgr-s	47.4	56.4†	59.1*†	59.6*†	**59.9***†

Table 9
Horehound Dataset: Mean IoU for LBP features for various choices of colour channel, rotational invariance, number of neighbours, and radius. Table has been annotated in the same way as Table 7.

Parameters	Radius						
	1	2	3	4	6	8	10
grey-i-8	65.1†	67.0†	68.2†	67.7	66.3	66.4	67.4*
grey-i-16	-	68.5*†	70.4*†	69.1	70.0*†	69.6*†	69.4*†
grey-i-24	-	-	70.3*†	***70.9****†	70.3*†	69.8*†	69.5*†
grey-v-8	64.9*†	67.7*†	68.4*†	67.7	66.0	66.2	68.0*†
grey-v-16	-	67.9*†	70.4*†	70.1*†	**70.8***†	69.2*†	67.9*†
grey-v-24	-	-	69.2*	70.6*†	70.1*†	68.5†	66.5
exg-i-8	60.3*†	61.0*	61.5*	61.5*	62.4*	61.8*	59.0*
exg-i-16	-	60.5*	62.3*†	**62.6***	61.7*	61.6*	60.6*
exg-i-24	-	-	61.6*†	61.3*	61.3*	59.9*	57.6*
exg-v-8	58.3*	60.3*	59.2*	59.0*	**60.4***	60.2*	59.5*
exg-v-16	-	56.1*	56.5*	56.7*	55.7*	55.5*	54.0
exg-v-24	-	-	56.3*	55.2*	55.8*	55.9*	52.1
exr-i-8	**60.2***†	59.6*	58.4*	59.1*	56.9*	55.4*	56.6*
exr-i-16	-	59.3*	59.1*	59.6*	58.1*	56.7*	57.0*
exr-i-24	-	-	58.4*	58.1*	56.4*	56.5*	59.9*
exr-v-8	**59.5***†	50.5	52.2	53.5	54.9*	54.7*	55.9*
exr-v-16	-	51.7	53.7	53.8	54.7	53.6	53.5
exr-v-24	-	-	52.4	52.2	53.6	56.0*	57.5*
xgr-i-8	63.1†	60.2	63.0†	63.0	62.9†	61.1	60.7
xgr-i-16	-	61.5	64.7*†	64.1†	64.7*†	64.5*†	64.2*†
xgr-i-24	-	-	64.1†	**67.2***†	65.2*†	64.1*†	64.1*†
xgr-v-8	**63.5***†	58.7*	59.7*	60.6*	61.1*	60.8*	59.7
xgr-v-16	-	56.1	62.5*†	60.5*	63.3*†	62.1*	60.8*
xgr-v-24	-	-	60.7*	63.4*†	62.3*†	62.9*†	60.7*

Table 10
Horehound Dataset: Mean IoU for HOG features for various choices of colour channel, signed orientation and number of bins. Table has been annotated in the same way as Table 8.

Parameters	Number of Bins				
	4	6	9	12	15
grey-u	48.8*†	49.5*	49.8*†	50.1*†	**50.4***†
grey-s	52.8†	***56.3****†	55.4*†	54.1*†	53.1*†
exg-u	53.0*†	54.1*†	54.9*†	**55.2***†	55.0*†
exg-s	45.2	54.2*†	54.4*†	**54.4***†	54.3*†
exr-u	44.9	48.2*	48.2*	48.0*†	**49.0***†
exr-s	45.9	**49.7***†	49.0*	48.2*†	47.6*†
xgr-u	51.0*†	51.1*†	51.6*†	51.8*†	**51.8***†
xgr-s	46.4	**54.6***†	53.3*†	52.8*†	52.5†

Table 11
Thistle Dataset: Mean IoU for LBP features for various choices of colour channel, rotational invariance, number of neighbours, and radius. Table has been annotated in the same way as Table 7.

Parameters	Radius						
	1	2	3	4	6	8	10
grey-i-8	66.3	67.8	69.1†	68.7	68.4	68.7	68.6
grey-i-16	-	70.0†	70.5†	71.0	70.9*†	70.1	69.2
grey-i-24	-	-	70.7†	**71.6***†	71.0*†	70.4†	69.5†
grey-v-8	65.6	66.2	66.3	66.8	67.6	**68.3***	68.2*
grey-v-16	-	64.7	64.7	66.1	67.8*	68.0*	67.7
grey-v-24	-	-	63.8	65.4	67.3	67.7*	67.4
exg-i-8	***71.8****†	71.2*†	70.7†	68.0†	65.2	65.4	64.4
exg-i-16	-	71.1*†	71.0*†	67.9†	66.4	64.8	64.4
exg-i-24	-	-	70.8†	68.5†	66.2	65.4	64.1
exg-v-8	**71.7***†	70.0	69.1	67.8†	65.8	65.7	64.9
exg-v-16	-	69.3	67.8	66.4	64.5	62.8	62.7
exg-v-24	-	-	66.6	64.8	63.0	61.4	60.6
exr-i-8	65.4*	64.1	63.5	65.3*	59.7	59.7	59.4
exr-i-16	-	65.7*	**66.3***	65.5*	65.4*	64.0	60.5
exr-i-24	-	-	65.9*	66.1*	63.9	63.7	62.2
exr-v-8	**68.6***	63.9	62.6	64.9	60.0	61.3	60.6
exr-v-16	-	62.5	63.0	64.2	62.2	60.3	58.4
exr-v-24	-	-	62.8	62.4	60.6	59.5	57.9
xgr-i-8	66.5	66.4	67.2	67.7*	61.2	61.6	61.7
xgr-i-16	-	67.5*	**68.3***†	67.8*	68.0*	67.6*	64.9
xgr-i-24	-	-	67.9†	66.1	67.8*	66.9*	65.3
xgr-v-8	**68.5***	66.3	66.5	67.4*	61.3	62.5	61.4
xgr-v-16	-	66.1	65.8	64.4	63.7	63.4	60.6
xgr-v-24	-	-	65.2	63.3	63.2	62.4	60.2

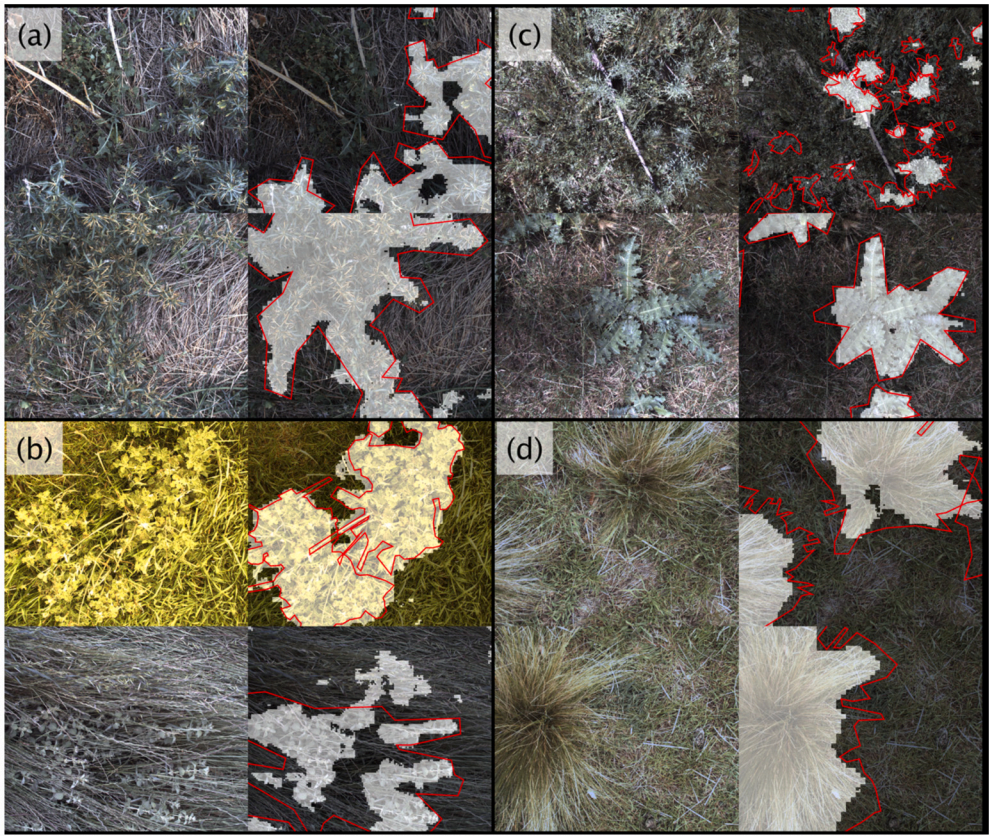

Fig. 3. Example outputs of model for (a) Bathurst burr, (b) horehound, (c) thistle and (d) tussock datasets. Polygon annotations drawn by a human annotator are shown in red, and regions classified as weed by the model are show in white.

[5,16,14]). Colour features also performed well, being the best feature for one dataset, and not significantly different from the best feature for another. By comparison, HOG features performed surprisingly poorly on all datasets tested.

Appropriate optimization of hyperparameters also proved to be important in designing the feature extractors. For LBP features, the best choice of hyperparameters leads to an improvement in mIOU of 4.8% to 18.4% over the default choice (we take this to be $r = 1$, $n = 8$, and

Table 12
Thistle Dataset: Mean IoU for HOG features for various choices of colour channel, signed orientation and number of bins. Table has been annotated in the same way as Table 8.

Parameters	Number of Bins				
	4	6	9	12	15
grey-u	46.3*	**46.9***	45.5	45.0	44.4
grey-s	53.4†	***57.6****†	57.1*†	55.8†	54.9†
exg-u	38.3	51.6	52.5*	52.3*	**53.0***†
exg-s	40.7	45.2	50.4*	**50.8***	50.3
exr-u	38.5	45.4	50.0	**51.1***	50.0
exr-s	47.0	49.7	50.1	50.1	**52.3***
xgr-u	41.7	46.2	48.1*	**48.6***	48.2
xgr-s	50.3	53.5*	54.9*	**54.9***†	54.6*†

Table 13
Tussock dataset: Mean IoU for LBP features for various choices of colour channel, rotational invariance, number of neighbours, and radius. Table has been annotated in the same way as Table 7.

Parameters	Radius						
	1	2	3	4	6	8	10
grey-i-8	75.5	75.6	76.6	78.8	79.6	76.1	70.1
grey-i-16	-	80.0	81.1	83.3*†	83.1*†	81.3†	79.7†
grey-i-24	-	-	82.3*†	83.2*†	**83.4***†	81.4†	79.9†
grey-v-8	82.4†	**83.1***†	82.4*	82.3*†	81.2	77.5	72.9
grey-v-16	-	82.6†	82.0	82.6*†	81.7	80.1†	79.2
grey-v-24	-	-	82.6*†	82.0	81.4	79.6	78.7
exg-i-8	76.8	79.7*	78.8	79.6	80.5	79.0	72.3
exg-i-16	-	79.5*	78.4	80.4	81.3*	**81.6***†	80.9†
exg-i-24	-	-	79.3	80.5	81.4*	80.9	80.6†
exg-v-8	81.3†	83.4*†	***83.4****†	82.8†	81.6	79.2	74.4
exg-v-16	-	82.4	81.8	81.7†	81.7	81.1†	80.3†
exg-v-24	-	-	80.8	81.1	81.0	80.8†	80.2†
exr-i-8	75.2*	70.9	69.7	69.1	73.8	68.4	60.1
exr-i-16	-	73.9	75.2*	73.7	**76.1***	74.5	74.2
exr-i-24	-	-	73.0	75.6*	74.7	74.5	73.9
exr-v-8	**79.5***	76.3	74.2	74.6	75.9	71.8	63.6
exr-v-16	-	75.9	76.4	77.2	76.1	73.2	71.7
exr-v-24	-	-	75.3	75.6	75.0	73.4	71.6
xgr-i-8	75.7	72.6	72.1	75.8	78.3	76.7	70.0
xgr-i-16	-	75.5	76.1	**79.8***	78.4	77.0	77.8
xgr-i-24	-	-	75.6	78.1	78.2	77.7	75.4
xgr-v-8	**81.1***†	79.7	80.1*	81.1*	80.3*	78.3	74.0
xgr-v-16	-	75.4	78.7	79.7*	78.2	77.8	77.4
xgr-v-24	-	-	76.9	78.7	77.1	75.4	73.9

Table 14
Tussock Dataset: Mean IoU for HOG features for various choices of colour channel, signed orientation and number of bins. Table has been annotated in the same way as Table 8.

Parameters	Number of Bins				
	4	6	9	12	15
grey-u	75.4†	***75.8****†	75.5†	75.2†	75.1†
grey-s	61.8	**75.3***	75.2*†	74.8	74.5
exg-u	68.8	73.0	**73.6***	73.2	72.7
exg-s	51.9	71.8	**73.4***	73.1	72.6
exr-u	63.1	69.3	71.1	**71.6***	71.2*
exr-s	43.4	63.7	67.8	**69.7***	69.7*
xgr-u	70.8	73.7	**74.5***†	74.4*†	74.1†
xgr-s	51.7	72.1	**73.7***	73.4	73.1

grey channel), depending on the dataset. For HOG features, the difference in results for varying combinations of hyperparameters was not so pronounced, with the choice of colour channel being a far more important consideration. It was found that the use of 4 bins was almost always worse than other choices, except for an unsigned feature extractor working on the grey colour channel. There was often no significant difference between other choices of bins (i.e. 6 up to 15).

It is interesting to note that different colour channels were most effective for different datasets, which suggests that, if a multi-class segmenter was required, then features may need to be extracted from multiple channels/indices to produce acceptable results.

Table 15
All Datasets: Comparison of best feature for each feature type (colour, LBP and HOG). The best result for each dataset is in **bold**. Results shown are the mean ± standard deviation, for a 5×10 fold cross validation (Bathurst burr, horehound and thistle datasets), or for a set of 10 entirely independent train/test splits (tussock). Results which are not significantly different from the best result in the column are labelled with a †.

Feature Type	Dataset			
(Best)	Bathurst Burr	Horehound	Thistle	Tussock
Colour	79.5 ± 7.7†	70.8 ± 6.5	**74.3 ± 3.5**†	79.4 ± 4.1
LBP	**81.6 ± 7.7**†	**74.6 ± 5.3**†	71.8 ± 3.3	**83.4 ± 3.5**†
HOG	59.4 ± 6.6	58.8 ± 7.8	57.6 ± 3.6	75.8 ± 2.1

Table 16
All Datasets: order in which the different features were added to the feature fusion. At every iteration, the features were added to the existing fusion, one at a time, and the one which produced the best average mIOU was selected. The features are labelled as LBP-*r* for LBP features, where *r* is the radius, and HOG-*n* for HOG features, where *n* is the number of bins. RGB and yCbCr are the two colour features. The feature whose addition produced the best result for each dataset is labeled with a *, and the feature whose addition produced the smallest fusion whose result was not significantly different from the best result is labelled with a †. This table should be read as follows: the feature fusion for a dataset, consisting of *n* individual features was made up of the features given in rows $1, \ldots, n$ in the column for that dataset.

Feature Order	Bathurst Burr	Horehound	Thistle	Tussock
1	LBP-3	LBP-4	yCbCr	LBP-3
2	yCbCr	LBP-10	LBP-4	LBP-1
3	LBP-4 †	RGB †	LBP-3 †	LBP-6
4	HOG-9	LBP-1	LBP-1	HOG-15 †
5	LBP-10	LBP-3	RGB	LBP-4
6	RGB	LBP-8	LBP-2	LBP-8
7	HOG-6 *	HOG-4	LBP-6 *	LBP-2
8	HOG-4	yCbCr *	LBP-8	HOG-6
9	HOG-12	HOG-9	HOG-4	HOG-4 *
10	LBP-8	HOG-12	HOG-6	HOG-12
11	LBP-6	LBP-2	LBP-10	HOG-9
12	HOG-15	HOG-15	HOG-9	LBP-10
13	LBP-1	HOG-6	HOG-12	yCbCr
14	LBP-2	LBP-6	HOG-15	RGB

6. Conclusion

This work demonstrates the feasibility of using a shallow ANN trained by the ELM algorithm, and working on manually extracted features, to produce a fine-grained segmentation of agricultural images. Future work, however, should determine how attractive this approach is compared to other methods, such as the use of CNNs, which are the most common choice for segmentation problems.

This study also demonstrated that the choice of colour channel (or index) was important, and had a significant impact on the usefulness of extracted features, as measured using the mIOU metric. The use of a feature fusion was an effective method of improving results, resulting in an increase of 4.2 - 7.3% compared to the best individual feature, depending on the dataset considered. Future research could involve testing more colour channels/indices, or looking at novel ways of combining multiple features into a more effective model.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

The data that has been used is confidential.

Appendix A. Supplementary material

Supplementary material related to this article can be found online at https://doi.org/10.1016/j.atech.2023.100288.

References

[1] Growing at a slower pace, world population is expected to reach 9.7 billion in 2050 and could peak at nearly 11 billion around 2100, https://www.un.org/development/desa/en/news/population/world-population-prospects-2019.html.
[2] Muhammad Ahmad, Adil Mehmood Khan, Manuel Mazzara, Salvatore Distefano, Multi-layer extreme learning machine-based autoencoder for hyperspectral image classification, 2019.
[3] Mansoor Alam, Muhammad Shahab Alam, Muhammad Roman, Muhammad Tufail, Muhammad Umer Khan, Muhammad Tahir Khan, Real-time machine-learning based crop/weed detection and classification for variable-rate spraying in precision agriculture, in: 2020 7th International Conference on Electrical and Electronics Engineering (ICEEE), IEEE, 2020, pp. 273–280.
[4] Darah Aqel, Shadi Al-Zubi, Ala Mughaid, Yaser Jararweh, Correction to: extreme learning machine for plant diseases classification: a sustainable approach for smart agriculture, Clust. Comput. 25 (3) (jun 2022) 2303.

[5] A. Binch, C.W. Fox, Controlled comparison of machine vision algorithms for Rumex and Urtica detection in grassland, Comput. Electron. Agric. 140 (2017) 123–138.

[6] Petra Bosilj, Erchan Aptoula, Tom Duckett, Grzegorz Cielniak, Transfer learning between crop types for semantic segmentation of crops versus weeds in precision agriculture, J. Field Robot. 37 (1) (2020) 7–19.

[7] Remco R. Bouckaert, Eibe Frank, Evaluating the replicability of significance tests for comparing learning algorithms, in: Advances in Knowledge Discovery and Data Mining, Springer Berlin Heidelberg, Berlin, Heidelberg, 2004, pp. 3–12.

[8] Brendan Calvert, Alex Olsen, Bronson Philippa, Mostafa Rahimi Azghadi, AutoWeed: detecting Harrisia cactus in the Goondiwindi region for selective spot-spraying, in: Proceedings of the 1st Queensland Pest Animal and Weed Symposium, Page 52, Weed Society of Queensland Pty. Ltd., Toowoomba, QLD, Australia, 2019.

[9] Brendan Calvert, Alex Olsen, James Whinney, Mostafa Rahimi Azghadi, Robotic spot spraying of harrisia cactus (Harrisia martinii) in grazing pastures of the Australian rangelands, Plants 10 (10) (2021) 2054.

[10] N. Dalal, B. Triggs, Histograms of Oriented Gradients for Human Detection, in: IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), IEEE.

[11] Schori Damian, Anken Thomas, Seatovic Dejan, Using fully convolutional networks for rumex obtusifolius segmentation, a preliminary report, in: 2019 International Symposium ELMAR, IEEE, 2019, pp. 119–122.

[12] Candra Dewi, Wayan Firdaus Mahmudy, Rio Arifando, Yoke Kusuma Arbawa, Beryl Labique Ahmadie, Improve performance of extreme learning machine in classification of patchouli varieties with imbalanced class, in: Proceedings of the 5th International Conference on Sustainable Information Engineering and Technology, ACM, New York, NY, USA, nov 2020, pp. 16–22.

[13] Anita Dixit, Nagaratna Hedge, B. Eswara Reddy, Vegetation type classification system using pseudo Zernike moments and ELM, in: 2017 IEEE International Conference on Power, Control, Signals and Instrumentation Engineering (ICPCSI), IEEE, sep 2017, pp. 2502–2508.

[14] Alessandro dos Santos Ferreira, Daniel Matte Freitas, Gercina Gonçalves da Silva, Hemerson Pistori, Marcelo Theophilo Folhes, Weed detection in soybean crops using ConvNets, Comput. Electron. Agric. 143 (2017) 314–324.

[15] Abhishek Dutta, Andrew Zisserman, The VIA annotation software for images, audio and video, in: Proceedings of the 27th ACM International Conference on Multimedia, MM '19, ACM, New York, NY, USA, 2019.

[16] Adnan Farooq, Xiuping Jia, Jun Zhou, Texture and shape features for grass weed classification using hyperspectral remote sensing images, in: IGARSS 2019 - 2019 IEEE International Geoscience and Remote Sensing Symposium, IEEE, 2019, pp. 7208–7211.

[17] Guang-Bin Huang, Qin-Yu Zhu, Chee-Kheong Siew, Extreme learning machine: a new learning scheme of feedforward neural networks, in: 2004 IEEE International Joint Conference on Neural Networks (IEEE Cat. No. 04CH37541), IEEE, 2004.

[18] Kaiming He, Xiangyu Zhang, Shaoqing Ren, Jian Sun, Deep residual learning for image recognition, in: 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, 2016.

[19] Kun Hu, Guy Coleman, Shan Zeng, Zhiyong Wang, Michael Walsh, Graph weeds net: a graph-based deep learning method for weed recognition, Comput. Electron. Agric. 174 (2020) 105520.

[20] Calvin Hung, Zhe Xu, Salah Sukkarieh, Feature learning based approach for weed classification using high resolution aerial images from a digital camera mounted on a UAV, Remote Sens. 6 (12) (2014) 12037–12054.

[21] Wajahat Kazmi, Francisco Garcia-Ruiz, Jon Nielsen, Jesper Rasmussen, Hans Jørgen Andersen, Exploiting affine invariant regions and leaf edge shapes for weed detection, Comput. Electron. Agric. 118 (2015) 290–299.

[22] Abbas Khan, Talha Ilyas, Muhammad Umraiz, Zubaer Ibna Mannan, Hyongsuk Kim, CED-Net: crops and weeds segmentation for smart farming using a small cascaded encoder-decoder architecture, Electronics 9 (10) (2020) 1602.

[23] Tsampikos Kounalakis, Georgios A. Triantafyllidis, Lazaros Nalpantidis, Image-based recognition framework for robotic weed control systems, Multimed. Tools Appl. 77 (8) (2018) 9567–9594.

[24] Tsampikos Kounalakis, Georgios A. Triantafyllidis, Lazaros Nalpantidis, Deep learning-based visual recognition of rumex for robotic precision farming, Comput. Electron. Agric. 165 (2019) 104973.

[25] Petre Lameski, Eftim Zdravevski, Vladimir Trajkovik, Andrea Kulakov, Weed detection dataset with RGB images taken under variable light conditions, in: ICT Innovations 2017, Springer International Publishing, Cham, 2017, pp. 112–119.

[26] Vi Le Nguyen Thanh, Selam Ahderom, Kamal Alameh, Performances of the LBP based algorithm over CNN models for detecting crops and weeds with similar morphologies, Sensors 20 (8) (2020) 2193.

[27] Bo Liu, Ryan Bruch, Weed detection for selective spraying: a review, Curr. Robot. Rep. 1 (1) (2020) 19–26.

[28] P. Lottes, M. Hoeferlin, S. Sander, M. Muter, P. Schulze, Lammers C. Stachniss, An effective classification system for separating sugar beets and weeds for precision farming applications, in: 2016 IEEE International Conference on Robotics and Automation (ICRA), IEEE, 2016, pp. 5157–5163.

[29] Philipp Lottes, Jens Behley, Andres Milioto, Cyrill Stachniss, Fully convolutional networks with sequential information for robust crop and weed detection in precision farming, IEEE Robot. Autom. Lett. 3 (4) (2018) 2870–2877.

[30] Philipp Lottes, Markus Hörferlin, Slawomir Sander, Cyrill Stachniss, Effective vision-based classification for separating sugar beets and weeds for precision farming, J. Field Robot. 34 (6) (2017) 1160–1178.

[31] Maitiniyazi Maimaitijiang, Abduwasit Ghulam, Paheding Sidike, Sean Hartling, Matthew Maimaitiyiming, Kyle Peterson, Ethan Shavers, Jack Fishman, Jim Peterson, Suhas Kadam, Joel Burken, Felix Fritschi, Unmanned Aerial System (UAS)-based phenotyping of soybean using multi-sensor data fusion and extreme learning machine, ISPRS J. Photogramm. Remote Sens. 134 (dec 2017) 43–58.

[32] Chris McCool, Tristan Perez, Ben Upcroft, Mixtures of lightweight deep convolutional neural networks: applied to agricultural robotics, IEEE Robot. Autom. Lett. 2 (3) (2017) 1344–1351.

[33] Andres Milioto, Philipp Lottes, Cyrill Stachniss, Real-time semantic segmentation of crop and weed for precision agriculture robots leveraging background knowledge in CNNs, in: 2018 IEEE International Conference on Robotics and Automation (ICRA), IEEE, 2018, pp. 2229–2235.

[34] Anders Krogh Mortensen, Soren Skovsen, Henrik Karstoft, Rene Gislum, The oil radish growth dataset for semantic segmentation and yield estimation, in: 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), IEEE, 2019, pp. 2703–2710.

[35] Vi Nguyen Thanh Le, Beniamin Apopei, Kamal Alameh, Effective plant discrimination based on the combination of local binary pattern operators and multiclass support vector machine methods, Inf. Process. Agric. 6 (1) (2019) 116–131.

[36] T. Ojala, M. Pietikainen, T. Maenpaa, Multiresolution gray-scale and rotation invariant texture classification with local binary patterns, IEEE Trans. Pattern Anal. Mach. Intell. 24 (7) (2002) 971–987.

[37] Alex Olsen, Dmitry A. Konovalov, Bronson Philippa, Peter Ridd, Jake C. Wood, Jamie Johns, Wesley Banks, Benjamin Girgenti, Owen Kenny, James Whinney, Brendan Calvert, Mostafa Rahimi Azghadi, Ronald D. White, DeepWeeds: a multiclass weed species image dataset for deep learning, Sci. Rep. 9 (1) (2019).

[38] Kavir Osorio, Andrés Puerto, Cesar Pedraza, David Jamaica, Leonardo Rodríguez, A deep learning approach for weed detection in lettuce crops using multispectral images, AgriEngineering 2 (3) (2020) 471–488.

[39] Ciro Potena, Daniele Nardi, Alberto Pretto, Fast and accurate crop and weed identification with summarized train sets for precision agriculture, in: Intelligent Autonomous Systems 14, Springer International Publishing, Cham, 2017, pp. 105–121.

[40] Edmund Sadgrove, Greg Falzon, David Miron, David Lamb, Fast object detection in pastoral landscapes using a Colour Feature Extreme Learning Machine, Comput. Electron. Agric. 139 (2017).

[41] Edmund Sadgrove, Greg Falzon, David Miron, David Lamb, Fast object detection in pastoral landscapes using a multiple expert colour feature extreme learning machine, in: Asian-Australasian Conference on Precision Pastures and Livestock Farming, 2017.

[42] Edmund Sadgrove, Greg Falzon, David Miron, David Lamb, Real-time object detection in agricultural/remote environments using the multiple-expert colour feature extreme learning machine (MEC-ELM), Comput. Ind. 98 (2018).

[43] Edmund Sadgrove, Greg Falzon, David Miron, David Lamb, The segmented colour feature extreme learning machine: applications in agricultural robotics, Agronomy 11 (11) (2021) 2290.

[44] Abhinav Sharma, Arpit Jain, Prateek Gupta, Vinay Chowdary, Machine learning applications for precision agriculture: a comprehensive review, IEEE Access 9 (2021) 4843–4873.

[45] Lyndon N. Smith, Arlo Byrne, Mark F. Hansen, Wenhao Zhang, Melvyn L. Smith, Weed classification in grasslands using convolutional neural networks, in: Michael E. Zelinski, Tarek M. Taha, Jonathan Howe, Abdul A. Awwal, Khan M. Iftekharuddin (Eds.), Applications of Machine Learning, SPIE, 2019, p. 42.

[46] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, Zbigniew Wojna, Rethinking the inception architecture for computer vision, in: 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, 2016.

[47] JingLei Tang, ZhiGuang Zhang, Dong Wang, Jing Xin, LiJun He, Research on weeds identification based on K-means feature learning, Soft Comput. 22 (22) (2018) 7649–7658.

[48] Jonathan Tapson, Philip de Chazal, André van Schaik, Explicit Computation of Input Weights in Extreme Learning Machines, Proceedings of ELM-2014, vol. 1, Springer International Publishing, Cham, 2015, pp. 41–49.

[49] Vi Nguyen Thanh Le, Giang Truong, Kamal Alameh, Detecting weeds from crops under complex field environments based on faster RCNN, in: 2020 IEEE Eighth International Conference on Communications and Electronics (ICCE), IEEE, 2021, pp. 350–355.

[50] Muammer Turkoglu, Davut Hanbay, Leaf-based plant species recognition based on improved local binary pattern and extreme learning machine, Phys. A, Stat. Mech. Appl. 527 (2019) 121297.

[51] Wenhao Zhang, Mark F. Hansen, Timothy N. Volonakis, Melvyn Smith, Lyndon Smith, Jim Wilson, Graham Ralston, Laurence Broadbent, Glynn Wright, Broad-leaf weed detection in pasture, in: 2018 IEEE 3rd International Conference on Image, Vision and Computing (ICIVC), IEEE, 2018, pp. 101–105.

[52] Wentao Zhu, Jun Miao, Laiyun Qing, Constrained extreme learning machines: a study on classification cases, https://arxiv.org/abs/1501.06115, 2015.