Neural Computing and Applications (2020) 32:6843–6855
https://doi.org/10.1007/s00521-019-04154-4

ORIGINAL ARTICLE

Fully automatic alpha matte extraction using artificial neural networks

Roberto Rosas-Romero[1] · Omar Lopez-Rincon[1] · Oleg Starostenko[1]

Received: 1 June 2018 / Accepted: 15 March 2019 / Published online: 21 March 2019

Abstract

The alpha matte is a two-dimensional map that is used to combine two images, one containing a foreground and the other containing a background. Alpha matte extraction is performed on green-screen images and requires user interaction to tune parameters in different preprocessing and postprocessing stages to refine an alpha matte. This paper tackles the problem of fully automatic extraction of the foreground on green-screen images with extraction of the corresponding alpha matte. The method is based on a multi-layer perceptron that assigns an alpha value, from a discrete set of ten alpha values, to each patch on a green-screen image. The approach for assigning an alpha value to an image patch is based on a set of features that enhance discrimination between foreground and background. The classifier is trained to learn to separate foreground objects from green-screen backgrounds as well as to generate the corresponding alpha matte map required for subsequent digital compositing. To test how the proposed approach handles alpha matte extraction under unsuitable conditions, a 64-image dataset was generated. The main contribution is that our method overcomes two challenges publicly posed within a dataset of green-screen image sequences, donated by *Hollywood Camera Work LLC*. Tests with this dataset generate high-quality visual results for those two cases. These results are confirmed by comparing the proposed fully automatic alpha matte extraction with that based on the use of *Adobe After Effects Creative Cloud*, an application which heavily depends on user interaction.

Keywords Alpha matting · Digital compositing · Green screen · Machine learning · Backpropagation algorithm

1 Introduction

Green-screen matting is widely used in applications for digital compositing and editing of images and videos, particularly, for generation of special visual effects. In these tasks, elements, known as foregrounds, are imposed over a background, generating a single image or video sequence. Despite many solutions for accurate foreground extraction, high-precision automatic matting is still a challenging task [1–4].

A pixel on a composite image combines information from both, foreground and background pixels, extending the RGB primary color model to RGBα, where α refers to the alpha channel or transparency. Therefore, for digital compositing, the most important process is alpha matte (transparency or opacity map for foreground objects in a compositing image) extraction. For example, if the background is characterized by a constant green color, values of $\alpha \in [0,1]$ are estimated by computing a color difference matte

$$\alpha = 1 - [G - \max(R, B)] \tag{1}$$

According to Eq. 1, background alpha values represent relatively high brightness, while foreground values are nearly zero. Brightness differences between background and foreground, as a result of applying Eq. 1, are illustrated in Fig. 1.

The alpha matte is a map, which specifies how to place the foreground over a background by introducing a weighting parameter to control the mixing of both on a digital composite. Consider the RGB components of a foreground pixel $\mathcal{P}_\mathrm{f} = [R_\mathrm{f}, G_\mathrm{f}, B_\mathrm{f}]^T$ and the RGB components of a background pixel $\mathcal{P}_\mathrm{b} = [R_\mathrm{b}, G_\mathrm{b}, B_\mathrm{b}]^T$; then, a pixel on a digital composite is a function that combines pixel values from background and foreground, according to

✉ Oleg Starostenko
oleg.starostenko@udlap.mx

[1] Department of Electrical and Computer Engineering, Universidad de las Americas-Puebla, Sta. Catarina Martir, 72810 Cholula, Pue, Mexico

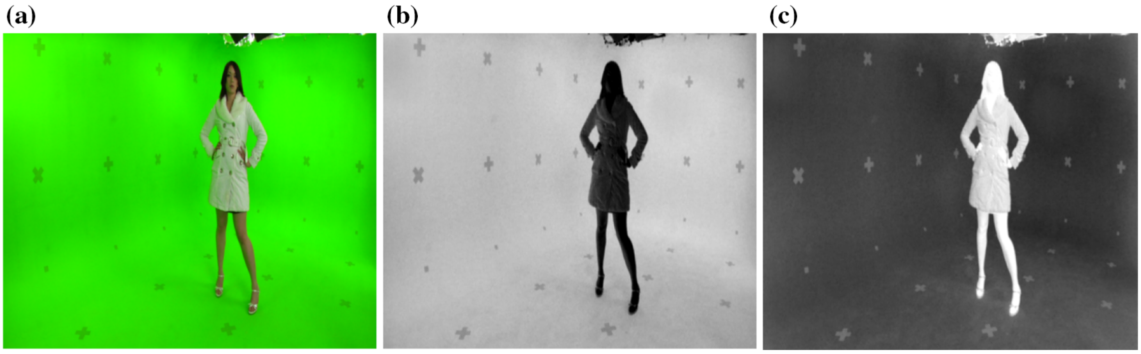


Fig. 1 **a** Green-screen image; **b** difference between the green channel and the maximum of other two channels; **c** the corresponding color difference matte (color figure online)

$$\mathcal{P} = \alpha\mathcal{P}_{\mathrm{f}} + (1 - \alpha)\mathcal{P}_{\mathrm{b}}; \quad \alpha \in [0, 1], \tag{2}$$

where α is called the alpha channel or transparency. If the transparency is zero, then the foreground objects are zero or transparent, while if alpha is one then the foreground objects are opaque. The fact that the transparency takes values between 0 and 1 allows for a smooth transition at edges between foreground and background.

There are different issues that prevent from accurate alpha matte extraction:

1. Estimation of accurate correlation between edges of the combined foreground and background pixels.
2. Lightning conditions without an evenly lit screen, which introduce overexposure, underexposure or impure screen color.
3. Incorrect color discrimination for neighboring pixels.
4. Influence of green-screen artifacts (seams, folds, tears, patches, tapes) which cause a deviation from a pure and uniform background color.
5. Foregrounds with contamination from green background. This contamination is magnified when green values on the foreground are higher than the other color components. This introduces unwanted holes in the extracted alpha matte.
6. The need for heavy user interaction during digital compositing of images and videos.

To overcome these limitations, our first work [5] introduced an approach for alpha matte extraction from green-screen images under unsuitable conditions. We reported that quality of alpha matte extraction is related to suitable conditions such as right color for green screen, the screen should be evenly lit, and there should be a significant separation distance between the subject and the green screen. Besides working on automatic alpha matte extraction, our first effort effectively handled scenarios where the aforementioned conditions were not given. We showed results where the proposed approach handled objects with intricate boundaries such as earrings, hair or filaments, blonde hair or foregrounds with some green in its composition. The new theoretical and practical contribution of this paper is a modified framework for fully automatic alpha matte extraction which provides high-quality visual results for two specific challenges within the public dataset *Hollywood Camera Work LCC*. Solutions to these cases have not been achieved in works reported in the scientific literature. There are no previous works discussing results based on the use of the Hollywood Camera Work dataset, and few works are focused on green-screen images.

The paper is organized as follows: Existed relevant solutions for alpha matte extraction, in related works, are analyzed in the second section. The third section describes the proposed approaches for fully automatic extraction of foreground and alpha matte in green-screen images. Experimental results are discussed in the fourth section as well as the evaluation of the proposed approaches on one public dataset. Finally, conclusions are presented in the last session of the paper.

2 Related works for alpha matte extraction

There are many scientific reports about methods for extraction of foreground and alpha matte in different environments and image conditions. Most authors address the general problem of segmenting a foreground from a background, which is unknown a priori. We are working on the more specific problem of extracting a foreground from a background with a color, which is known a priori. The following literature review is useful for future work. Almost all the authors adopt the basic classification for existing matting methods subdivided into sampling-based and affinity-based approaches and combination of these two techniques [5–9]. Particularly, affinity-based approaches are subdivided into (1) methods that are focused on estimating alpha values and subsequent extraction of true foreground colors for unknown pixels based on precomputed alphas and (2) methods where alpha matte values are computed by propagation from known pixels to unknown ones, thus small errors could be propagated and accumulated to produce higher errors [4, 10, 11].

The majority of the related works address the problem of soft-segmenting a foreground from a background that is unknown a priori. In chroma keying, however, the foreground is captured against a background whose color is known a priori. Yes indeed, we may come across challenges such as the shading of the foreground on the green background. Nevertheless, in my opinion, these challenges are specific to this "scenario" of alpha matting, the green-screen matting.

The main feature of the sampling-based approaches consists in sampling image background and foreground to find candidate colors used for computation of the alpha matte, which is finally used for digital compositing of images or video. Among sampling-based approaches, different methods provide quite accurate alpha matte extraction. For example, the algorithm, proposed in [12], measures alpha values along manifold connecting frontiers of each object color distribution, where each distribution is represented as a set of point masses found through vector quantization. Another approach provides global sampling using all samples available for foreground and background colors in image, attempting to generalize the search space of entire boundary using randomized Patch Match algorithm [13]. An original approach for solving the matting problem, by modeling the foreground and background color distributions with spatially varying sets of Gaussians, is proposed in [14], where foreground estimation is achieved assuming fractional blending of foreground and background colors using the maximum likelihood criterion.

The problem of extracting alpha matte may be solved by applying the following procedures, as it has been proposed in [15], (1) collecting a candidate set of potential foreground and background colors; (2) selecting high confidence samples from the candidate set; and (3) estimating a sparsity prior to remove blurry artifacts. The approach proposed in [16] exploits extended spectral segmentation techniques for extraction of soft matting components by computing a set of fuzzy matting components from the smallest eigenvectors by a suitably defined Laplacian matrix. Soft matting components may be used later as building blocks to easily reconstruct semantically meaningful foreground mattes.

Due to the fact that pixels in a small neighborhood often have high similar values, another proposed approach estimates real-time alpha matting taking into account that an initial collection of samples gathered by nearby pixels differs by a small number of elements and close-by pixels usually are characterized very similar pairs for their foreground and background colors [11]. An interesting matting algorithm proposed in [17] tackles the matting task as a statistical transductive inference, where it is assumed that user-marked pixels do not fully capture the statistical distributions of foreground and background colors in the unknown region of the given tri-map; therefore, new foreground and background colors are allowed to be recognized in the transductive labeling process. The particular problem of missing true foreground and background samples is solved using a new sampling strategy to build a comprehensive set of known samples by sampling from all color distributions in known regions so that this set includes highly correlated boundary samples as well as samples inside the foreground and background regions to capture all color variations [7].

Another interesting method proposed in [18] avoids usage of matting equations that restrict the estimate of α from a single pair of foreground and background samples. Thus, image matting is considered as sparse coding problem, where the sparse codes directly give the estimate of the alpha matte from more than just one pair of foreground and background samples. In contrast to well-known image matting techniques, a proposal in [2] attempts to reduce memory consumption and employs global optimization over whole set of image pixels. When the image is divided into small patches self-adaptively according to the distribution of pixels, then the matting algorithm is applied in patch level providing significant reduction of memory consumption.

The improvement in color difference method for green-screen matting is proposed in [3]. The conventional color difference method for controlling result matte uses two thresholds, which group pixels in blocks. However, the proposal allows user to adjust the thresholds associated with groups of pixels with similar luminance, clustering them into the same bin of histogram and as result, it takes a little more effort to obtain higher quality matte result.

The principal disadvantage of the sampling-based methods is their limitation to operations with linear combination of two colors for solution of the matting equation as well as color problem sensitive to situations, where foreground and background color distributions have large overlaps. Additionally, sampling-based methods provide high accurate results, when foreground and background color distributions are quite different; however, they fail when required conditions are not well satisfied. Finally, in order to achieve high performance, the sampling-based matting methods require applying additional operations that increase their complexity and computational cost [19–21].

In contrast to sampling-based, the affinity-based approaches have some advantages because they process local image statistics by defining various affinities between neighboring pixels instead of directly estimating the alpha value at each pixel. Therefore, affinities are always defined in small neighborhood of immediately connected pixels, where their correlations are usually strong providing more easy way for achieving the smoothness of transition between edges of the foreground and background after combining them and reducing the dependence from uneven lit conditions, even for quite complex images [9, 19].

Among propagation-based methods, the following approaches, proposed in the scientific literature, stand out. One interesting proposal operates directly on the gradient of matte followed by matte reconstruction through the solution of Poisson equations, unlike previous methods that optimize a pixel alpha and background and foreground colors with statistics. When global Poisson matting fails to produce high-quality mattes due to a complex background, local Poisson matting manipulates a continuous matting field in a local region [22].

Another iterative optimization approach proposed in [19] has been introduced to solve the matting problem for every pixel in the image based on a small amount of foreground and background pixels marked by the user and Markov Random Fields. An interactive framework for soft segmentation and matting of natural images and videos is presented in [23], where the proposed method is based on the optimal computation of weighted geodesic distances to the user-provided scribbles, from which the whole data are automatically segmented. The weights are based on spatial and/or temporal gradients, without explicit optical flow or any feature detectors. A localized refinement step follows this fast segmentation in order to accurately compute the corresponding matte function. An alpha matting method, introduced in [1], with local and nonlocal smooth priors is based on observing that the manifold preserving editing propagation introduces a nonlocal smooth prior on the alpha matte which is combined with the local smooth prior from matting Laplacian complement and a simple data term from color sampling for nature image matting.

A similar algorithm to solve a large kernel matting Laplacian has been proposed in [24], where kernel propagates information more quickly and may improve the matte quality. To further reduce running time, the adaptive KD-tree tri-map segmentation technique is used. Also, minimization of a matting Laplacian matrix is used in spectral methods for improving segmentation by a sparse set of affinity linear functions [16, 25].

To solve the problem of image matting, another approach uses appearance models by construction more compact traditional line color and point–point color models without the need for any additional user interaction [26]. The proposal presented in [11] combines local sampling and the KNN classifier propagation-based matting algorithm so that the corresponding feature space, according to the different components of image colors, is built to reduce the influence of overlapping between the foreground and background and then a KNN classifier is used for processing based on the pros and cons of the sample performance of unknown image areas.

Very satisfactory matting results in transparent foreground region are obtained in the proposal based on propagation matting algorithm, which involves two representative nonlocal propagation-based approaches: KNN matting and manifold preserving editing propagation method providing in this way strengthening the local smoothness as well as processing texture as an additional feature provides effectively discriminate the foreground and background regions [4].

Although affinity-based approaches are relatively insensitive to different user inputs and always generate smooth mattes, they may not be very accurate and sometimes fail when the foreground has long and thin structures or holes. As a result, mattes generated by affinity-based approaches are sometimes not so precise than those generated by sampling-based approaches.

A possible overcoming disadvantages and achieving a good compromise between accuracy and robustness of aforementioned approaches consist in combining sampling-based and propagation-based methodologies together, developing more advanced systems. It is important to mention that recently proposed improved image matting approaches have explored artificial intelligence. As usual, they do not directly learn an alpha matte given an image and tri-map. For example, in the proposal introduced in [9], a deep convolutional encoder–decoder network is used to predict a tri-map from an input image. Additionally, a small convolutional network refines the alpha matte predictions from the first network to have more accurate alpha values and sharper edges.

Very satisfactory results are reported in [27], where the automatic image matting method consists in using deep learning for creating the tri-map of a person in portrait images by an end-to-end convolutional neural network that considers, not only image semantic prediction, but also pixel-level image matte optimization.

Similarly, the proposal to use deep convolutional neural networks for natural image matting is presented in [28]. The method provides visually and quantitatively high-quality alpha mattes taking as inputs (1) results of the closed form matting, (2) results of the k-nearest neighbors matting and (3) normalized RGB color images and directly learns end-to-end mapping between inputs and reconstructed alpha mattes.

One of the principal limitations of classifier-based approaches is the lack of public datasets with a significant number of images or videos for machine training and calibration. According to all the previous related work, our proposal is a simple neural network with very low computational cost. The number of multiplications at the first layer is in the order of the number of input features or patch dimensions n^2. It does not depend on multiple layers as deep learning machines. The number of input features is relatively low. The training of the proposed learning machine is based on one single image, a collage of patches from different foregrounds and patches from green backgrounds.

3 Proposed alpha matte extraction approach

3.1 Overview

An overview of the proposed alpha matte extraction is depicted in Fig. 2. Each patch (a two-dimensional array) on the green-screen image of interest is assigned an alpha value α by a classifier (trained neural network). The classifier is fed with a feature vector $\boldsymbol{x}$, extracted from a patch. The alpha value is discrete and takes one out of ten possible values, $\alpha \in \{0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9\}$. Each alpha value is associated with one output of the neural network. If an extracted feature vector $\boldsymbol{x}$ activates the second output of the neural network then the corresponding patch is assigned an alpha value of 0.1.

3.2 Training of the classifier

The classifier learns alpha matte extraction through supervised learning with a training set $\mathcal{T} = \{(\boldsymbol{x}_1, \boldsymbol{\alpha}_1), (\boldsymbol{x}_2, \boldsymbol{\alpha}_2), \ldots, (\boldsymbol{x}_k, \boldsymbol{\alpha}_k)\}$, where each set element $(\boldsymbol{x}_i, \alpha_i)$ consists of (1) one feature vector $\boldsymbol{x}_i$, extracted from an image patch and (2) one vector of desired values $\boldsymbol{\alpha}_i = \left[\alpha_{i,1}, \alpha_{i,2}, \ldots, \alpha_{i,10}\right]^T$ with ten entries. For the case of a vector of desired values $\boldsymbol{\alpha}_i = [0, 0, 0, 0, 0, 1, 0, 0, 0, 0]^T$, the alpha value assigned to the patch of interest is *0.5*.

The training set $\mathcal{T}$ is generated from a collage of images. This collage contains segments from multiple images, and it is carefully segmented in a manual fashion. A collage and its corresponding alpha matte are shown in Fig. 3. The right panel of Fig. 3 shows a zoomed out patch. At the middle panel in Fig. 3, it can be observed that an alpha matte contains smooth transitions around edges while pixels on the background have total transparency (alpha value of zero) and pixels on the foreground have partial opacity (alpha values from 0 to 1). The foreground object on the green-screen image includes a collage of fragments from other images. During training of the alpha matte extractor, patches are randomly picked from this collage along with their corresponding alpha values.

We are approaching alpha matte extraction as a classification problem, instead of a regression one. One reason is that generation of the training set heavily depends on manual assignment of alpha values to pixels, and the employment of a finite set of discrete values is very convenient for this manual task. One of our first efforts for alpha matte extraction was based on a linear regression model, where the model parameters are learned by using least squares. Even though the implementation of this approach is simple, extracted alpha values were not constrained in the range $y \in [0, 1]$. Another regression-based strategy was the modeling of alpha matte extraction with a logistic regression, where parameters are learned by using maximum likelihood. The last regression-based effort consisted of using a MLP with one output, which is trained with the backpropagation algorithm. In the last two cases, alpha values were constrained in the range $y \in [0, 1]$; however, visual results were not as good as those obtained with a classifier. An additional problem of using a regression was that there are situations where foreground pixels were assigned alpha values smaller than one.

3.3 Feature set

A feature vector $\boldsymbol{x}(\boldsymbol{s}_i) \in \mathbb{R}^N$ is extracted from an image patch $p(\boldsymbol{s}_i)$ at image position $\boldsymbol{s}_i$ (row and column). Information which is part of the N features is intensity color values contained in the three HSV (hue, saturation, value) planes from image patch $p(\boldsymbol{s}_i)$, where $N = 3 \times (n \times n)$. Instead of using RGB information, as in [5], HSV gives better results. The size of a patch on each color plane is $n \times n$. A patch from each color plane is stretched out as a column vector, and the three column vectors are concatenated to form

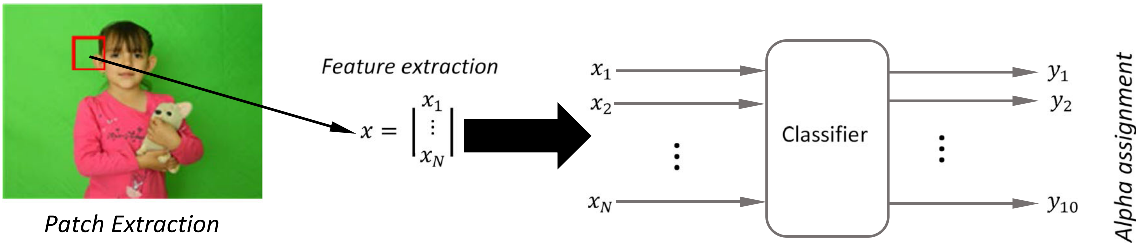


Fig. 2 General overview of the alpha matte extraction approach

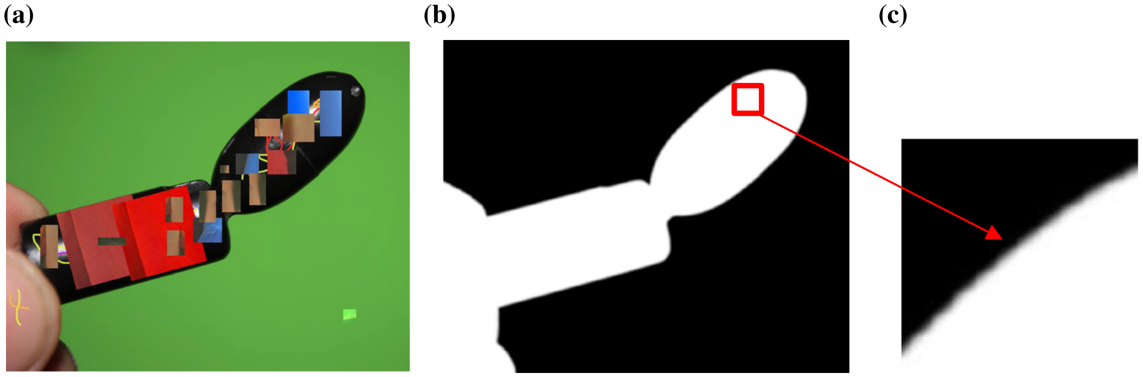


Fig. 3 **a** RGB image used as training set of the classifier; **b** image with corresponding target alpha values at each pixel location; **c** zoomed patch, from the alpha matte with smooth transitions around edges (color figure online)

$$x(s_i) = \left[\boldsymbol{x}_H(\boldsymbol{s}_i)^T \boldsymbol{x}_S(\boldsymbol{s}_i)^T \boldsymbol{x}_V(\boldsymbol{s}_i)^T\right]^T \in \mathbb{R}^{3\times n^2}. \quad (3)$$

Choosing patch of size, $n \times n = 5 \times 5$, accounts for $N = 75$ entries. Besides raw HSV pixel values, there are other relevant features which are useful to improve separation between foreground objects and the green background. One such feature, known as dissimilarity measure, is the square of the Euclidean distance between the color (red, green, blue) in the pixel of interest, $p_c = [R_P, G_P, B_P]^T$, and the pixel with the highest green intensity $p_g = [0, 1, 0]^T$,

$$x_{\text{Euclidean}} = \frac{1}{3} p_c - p_g^2 = \frac{1}{3}\left[R_P^2 + (G_P - 1)^2 + B_P^2\right], \quad (4)$$

where R_P, G_P and B_P are normalized values in the range [0, 1] and the square of the difference is multiplied by the factor to normalize this feature so that $0 \le x_{\text{Euclidean}} \le 1$. This relevant feature results in higher brightness for pixels on the foreground and nearly zero brightness for pixels on the background as it is observed in the middle part of Fig. 4.

A second relevant feature is a ratio between the sum of red and blue components over the green component

$$x_{\text{ratio}} = \frac{R_p + B_p}{2 \times 255 \times G_p} \quad (5)$$

where this feature results in high brightness on red and blue foregrounds and nearly zero brightness on a green background as it is shown in the right part of Fig. 4. The green component G_p takes values between 1 and 255 to avoid division by zero. The normalization of this feature, so that it takes values in range $0 \le x_{\text{ratio}} \le 1$, is possible with the inclusion of a constant factor 2 × 255 at the denominator. The total number of features N extracted from a patch is

$$3\,\text{HSV planes} \times (n \times n) + 2\,\text{special features}\; x_{\text{Euclidean}}\;\text{and}\; x_{\text{ratio}}$$

This feature set includes color intensity values and two relevant features. These two features highlight contrast between foreground and background, as it is shown in Fig. 4. Thus, these two features improve separation between foreground objects and green background when it comes to classify them.

3.4 Classifier

The classifier, shown in Fig. 5, assigns an alpha value to an N-entry feature vector $\boldsymbol{x} = [x_1, x_2, \ldots, x_N]^T$, extracted from

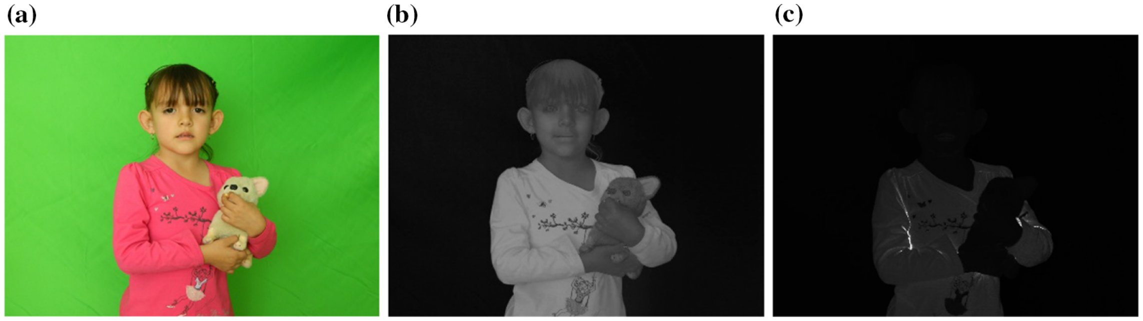


Fig. 4 **a** Green-screen patch from which two relevant features are extracted: **b** Euclidean distance between pixel RGB values and the highest intensity green color; **c** sum of red and blue components over green component (color figure online)

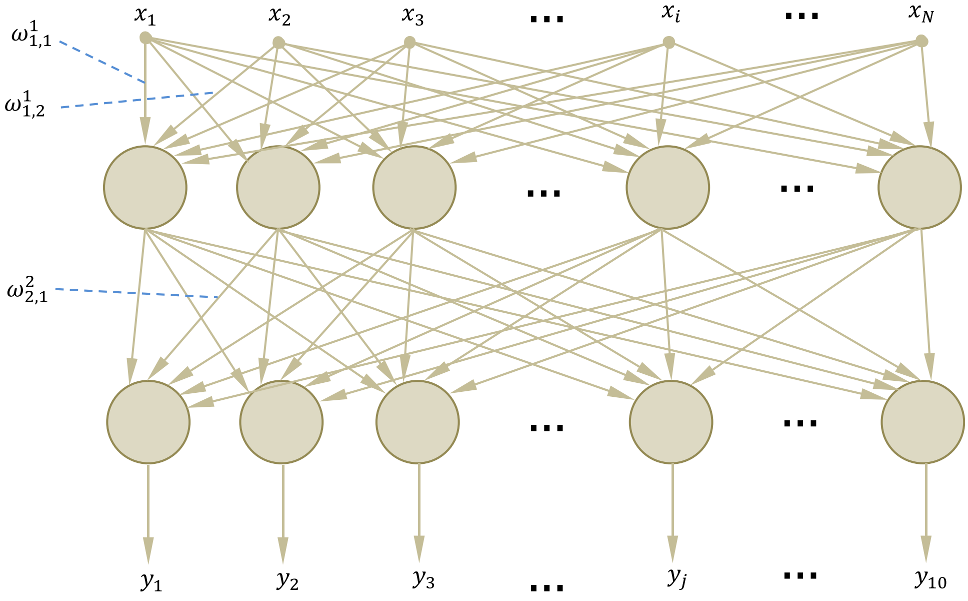


Fig. 5 Architecture of the input and output interfaces of the classifier used for alpha matte extraction

a green-screen patch. At each neural node incoming information $\boldsymbol{x}$ is processed by applying, two operations, a linear combination $v = \boldsymbol{w}^T\boldsymbol{x} + b$, where $\boldsymbol{w}$ is the set of *synaptic weights* and b is the *bias*; followed by an activation operation, defined by the logistic function $f(v) = \frac{1}{1+e^{-v}}$,

Neuron outputs $\boldsymbol{y} = [y_1, y_2, \ldots, y_{10}]^T$ correspond to a set of alpha values $\boldsymbol{\alpha} = [\alpha_1, \alpha_2, \ldots, \alpha_{10}]^T$. The output neuron y_i with the highest value determines the alpha value α_i assigned to input feature vector $\boldsymbol{x}$. According to Fig. 5, the architecture of classifier consists of a set of N input nodes, 10 output neurons and one hidden layer. Different multilayer perceptron (MLP) architectures were tested after variation of the number of hidden layers and neurons per layer to find an effective architecture.

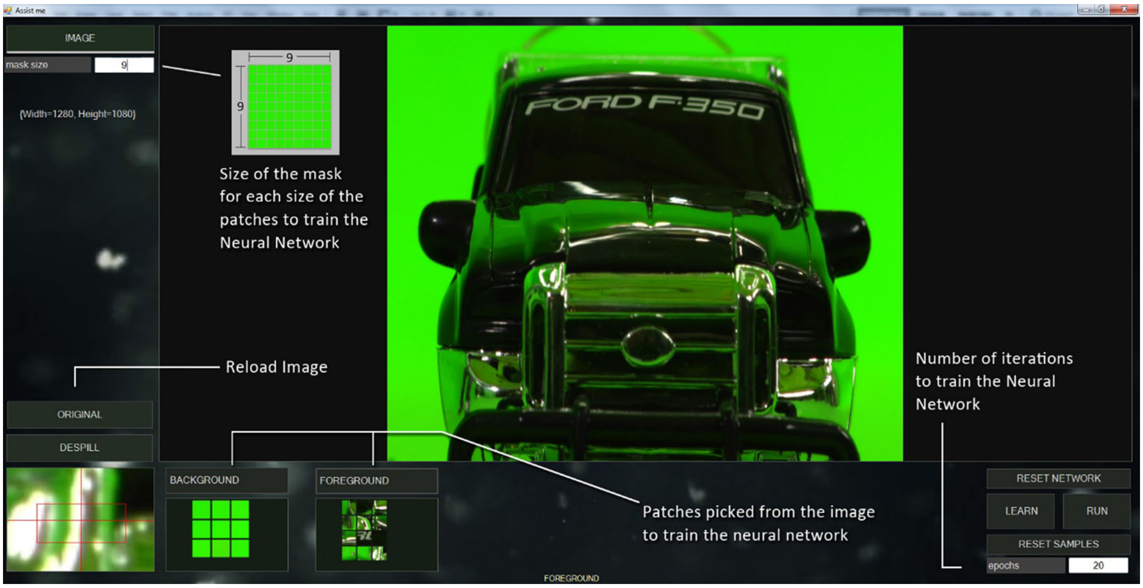
The training of the classifier is an iterative process and at each iteration a feature vector $\boldsymbol{x}_i$ is applied and the final output vector $\boldsymbol{y}_i$ is compared with the desired alpha matte vector $\boldsymbol{\alpha}_i$, according to the error function $E_i = \frac{1}{2}\boldsymbol{y}_i - \boldsymbol{\alpha}_{i2}^{2}$. Network parameters are adjusted at each iteration by adding to them the negative of the gradient of the error function according to $w_{m,n}(t+1) = w_{m,n}(t) - \eta \frac{\partial E_i}{\partial w_{m,n}}$, where η is the learning rate and $w_{m,n}$ is the synaptic weight of the connection between neuron n at layer ℓ and neuron m at the next layer, $\ell + 1$.

4 Results and discussion

An application was developed to train the alpha matte extractor, as it is described in Sect. 3.2, and to extract alpha matte and foreground. For each sequence of HD green-screen images, the alpha matte extractor is trained with one single collage. The application runs by following simple steps (Fig. 6): (1) Selection of the image of interest through the "IMAGE" button. (2) Specification of the patch size by using button "MASK SIZE." (3) Samples from background or foreground are picked by dragging the pointer over the image of interest and double clicking on the selected area so that a training set (collage described in Sect. 3.2) is generated. (3) The network is trained by specifying number of epochs and clicking the "LEARNING" button. During training, the system randomly picks patches from background and/or foreground samples. A patch picked from foreground is assigned a desired alpha value of one while a patch on background is assigned zero. During training, each pixel is assigned an alpha value by considering neighboring pixels within a window.

The classifier is a multi-layer perceptron (MLP), which consists of *2* hidden layers with *13* neurons per hidden layer and an output layer with *10* neurons. Model selection was done by trying different network architectures while searching for the minimum error. The number of hidden layers (flexibility of the model) was gradually adjusted from 1 to 4, while the number of neurons per hidden layer was gradually adjusted with increments of 5 from 8 to 48.

In one set of experiments, the proposed alpha matte extractor was tested on two sequences of high-definition (HD) green-screen images from the public dataset Hollywood Camera Work LLC [29]. A HD image was shot with an HVX-200 at a speed of 100 Mbps. Each case presents limitations, which are not good enough during film production. These interesting image sequences are donated to the public and pose different challenges. One HD green-screen image sequence of interest is the TOY CAR ROTO, with 255 plates of 1080 × 1280 pixels. The challenge of



mask size 9

(Width=1280, Height=1080)

epochs 20

Size of the mask for each size of the patches to train the Neural Network

Reload Image

Number of iterations to train the Neural Network

Patches picked from the image to train the neural network

Fig. 6 GUI of designed application for alpha matte extraction from green-screen images (color figure online)

this sequence is to automatically key a car on a green screen, despite massive green light reflections on the black car as it is shown in the first row of Fig. 7. There are car parts which reflect green light, and alpha matte extraction fails at recognizing that those parts are not green background, introducing holes on the alpha matte. The word ROTO is related to rotoscoping, which is the only technique that can be applied to this case in order to create the corresponding matte without holes. In the visual effects industry, rotoscoping is a fully manual technique for matte extraction. The first row of Fig. 6 shows one TOY CAR ROTO HD plate with corresponding alpha matte and foreground, both generated by the proposed method.

Another Hollywood Camera Work LCC sequence of 250 HD green-screen plates, with 720 × 960 pixels per plate, is the GODIVA MEDIUM sequence [29], which shows a girl moving an almost transparent shawl with a green-screen behind her, as it is shown in the second row of Fig. 7. The goal is to achieve perfect extraction of the shawl and to avoid background dark green regions to show up on the foreground after alpha matte extraction.

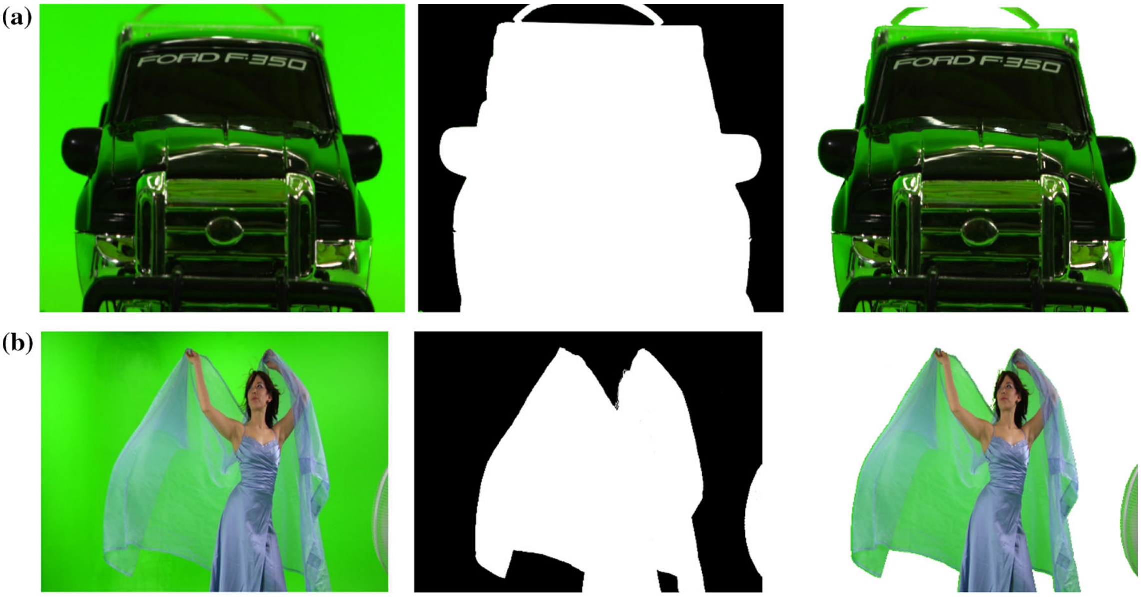


Fig. 7 **a** Extraction of alpha matte and foreground from HD green-screen plates for case TOY CAR ROTO in the first row; **b** extraction of alpha matte and foreground from GODIVA MEDIUM sequence in the second row (color figure online)

In a second group of experiments, 64 ad hoc images were generated to test alpha matte extraction under unsuitable conditions. There are situations, which pose challenges for alpha matte extraction. Examples of such situations are (1) unevenly lit green screens, (2) blurring due to motion, (3) people using eyeglasses, (4) short separation distance between subject and the green screen with the introduction of shadows around foreground contour, and (5) blonde hair, which is characterized by high green content. Figure 8 shows qualitative results of fully automatic alpha matte extraction on green-screen images under the aforementioned situations: (1) Unevenly lit green screen (Fig. 8a, e, f). (2) Subjects with blonde or light brown hair (Fig. 8a, e). (3) The presence of reflective and transparent objects, for example, glasses with thin rim (Fig. 8b). (4) Appearance of small objects in image (fingers, earring) (Fig. 8c, d). (5) Short separation distance between subject and green screen, which introduces dark green regions close to the foreground contour (Fig. 8f). For the case of the subject in Fig. 8c), green areas between the fingers of his right hand are correctly extracted. It is observed that foreground extraction handles objects with intricate boundaries such hair filaments (Fig. 8b, d, e), which are correctly separated, so that isolated hair is maintained. The proposed approach correctly estimates reflective, glossy and transparent objects. It completely extracts thin rim of glasses even though the thin rim is enclosed by green pixels in Fig. 8b). Very small objects, as the earring of the girl in Fig. 8e), are also maintained after foreground extraction. In all experiments, alpha matte extraction was fully automatic and postprocessing such as spill suppression was not required.

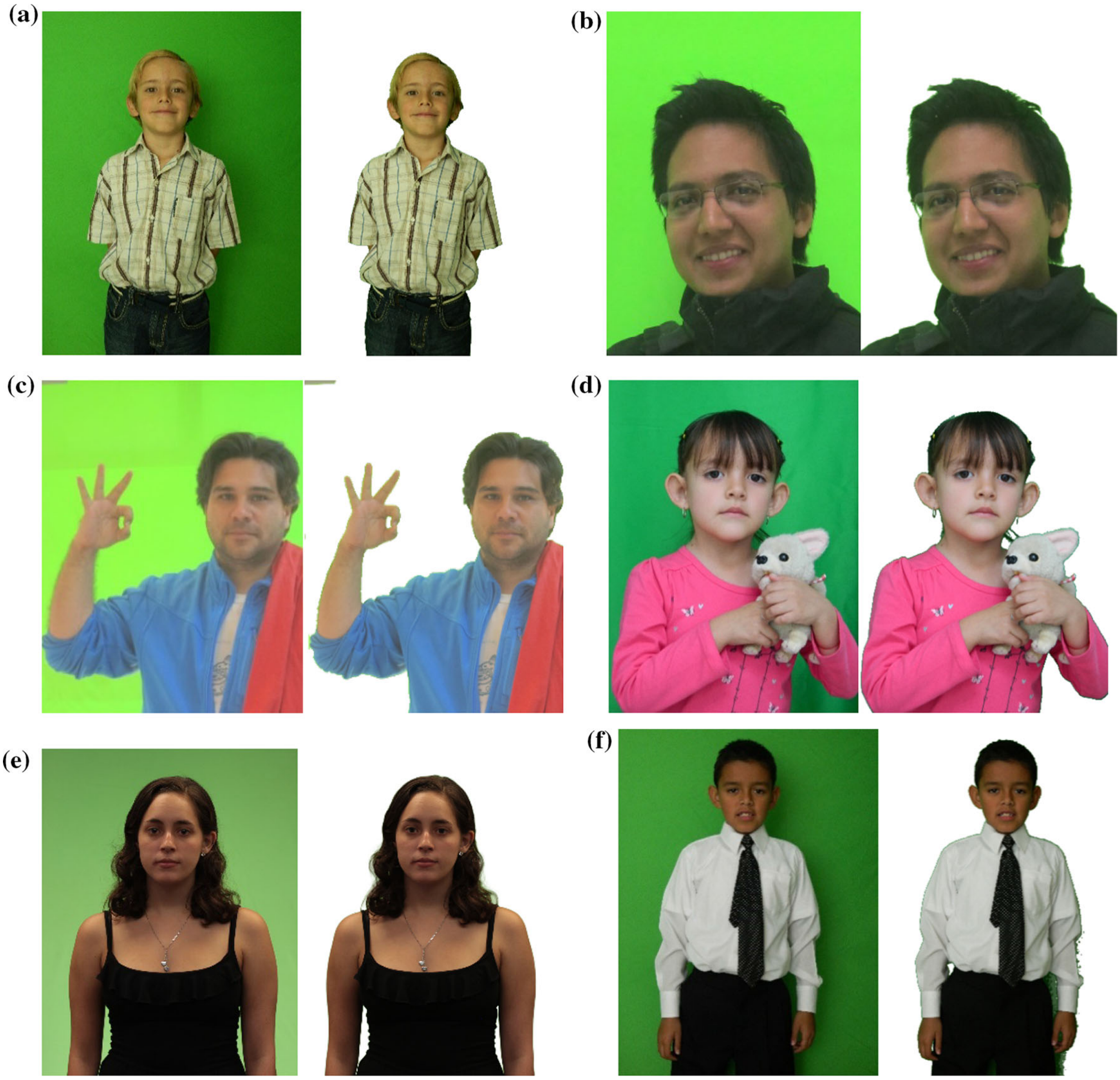


Fig. 8 Alpha matte extraction applied to images with some unsuitable conditions such as **a** blonde hair, **b** glasses with thin rim, **c**, **d** very small objects in image, **e**, **f** unevenly lit green-screen and short separation distance between subject and screen causing shadows

A quantitative comparison was performed by measuring the difference between extracted alpha values and ground truth values. The ground truth was obtained by using the commercial application Adobe After Effects CC. This application was developed by Adobe System for generation of visual effects and compositing during postproduction of films and TV programs. This application involves heavy manual efforts to generate good results. This process was executed on the 255 plates from the TOY CAR ROTO sequence and the 250 plates from the GODIVA MEDIUM sequence [29]. One metric is the average absolute difference (AAD),

$$\mathrm{AAD} = \frac{1}{\mathrm{rows} \times \mathrm{columns} \times \mathrm{frames}} \times \sum_{r=1}^{\mathrm{rows}} \sum_{c=1}^{\mathrm{columns}} \sum_{t=1}^{\mathrm{frames}} |\alpha_1(r,c,t) - \alpha_2(r,c,t)|$$

where r, c, t stand for row, column and frame indexes. The AAD value for the GODIVA MEDIUM video sequence is 41.7663 and 11.2317 for the TOY CAR ROTO video sequence. A second metric, that is used, is the mean squared difference (MSD),

$$\mathrm{MSD} = \frac{1}{\mathrm{rows} \times \mathrm{columns} \times \mathrm{frames}} \times \sum_{r=1}^{\mathrm{rows}} \sum_{c=1}^{\mathrm{columns}} \sum_{t=1}^{\mathrm{frames}} [\alpha_1(r,c,t) - \alpha_2(r,c,t)]^2$$

where the MSD value is 80.9213 for the GODIVA MEDIUM video sequence and 17.1054 for the TOY CAR ROTO video sequence. The first row in Fig. 9 shows the alpha matte of a single plate from the TOY CAR ROTO sequence and the second row shows results corresponding to the GODIVA MEDIUM sequence. The first column represents extracted alpha matte by the proposed approach and the second column shows the result obtained by Adobe After Effects CC application.

There are not previous works discussing results based on the use of the Hollywood Camera Work dataset and few works are focused on green-screen images. From our quantitative results, there are notable differences (AAD and MSD), between the proposed approach and the commercial application for the case of the TOY CAR ROTO sequence. These differences are visually revealed by comparing alpha mattes from both methods. For the CAR sequence, the commercial application has the disadvantage of misclassifying background and foreground because it assigns high alpha values to some patches on the background (lower part of plates) and labels foreground patches with relatively low alpha values, which introduces holes (upper part and right lower corner of the car). In the other hand, the challenge, of fully extracting the car without holes, is overcome by the proposed application. Differences, between the proposed method and the commercial application, are much higher for the case of the GODIVA MEDIUM sequence. For this sequence, the proposed application

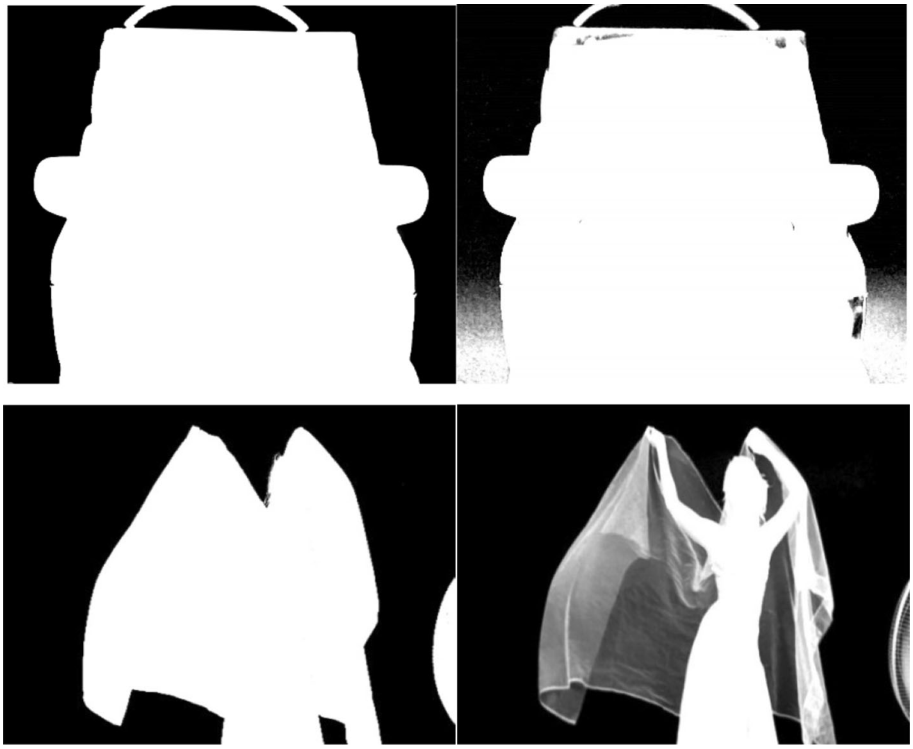


Fig. 9 Alpha mattes of single plate extracted from the TOY CAR ROTO sequence (first row) and the GODIVA MEDIUM sequence (second row) by using the proposed approach (first column) and Adobe After Effects CC (second column)

succeeds at smooth contour extraction of the shawl and both applications do not fail to misclassify background and foreground patches; however, the commercial application is better, when it comes to assign alpha values to the shawl on the foreground, emphasizing transparency of this object. This is because the commercial application most probably outputs fuzzy alpha values, while the proposed method produces discrete alpha values.

Among 64 ad hoc images, with unsuitable conditions, there were four images, where regions were not correctly separated. For example, Fig. 8d) shows the presence of small green regions inside hair after foreground extraction. In Fig. 8f, a green region between boy's left arm and his torso remains. The reason for this is that the green region is very dark. On the other hand, the green region, between boy's right arm and his torso, is correctly separated. This problem does not occur when people are standing at a considerable separation distance from the green screen (suitable scenario). In the aforementioned image, the boy is standing right in front of the green screen, a condition, which causes shadows.

In another set of experiments, the proposed method was tested on a green-screen image downloaded from the online dataset from alphamatting.com [30]. One example of the original image and ground truth taken from [30] is shown in Fig. 10(top row). Alpha values were assigned to patches of different odd sizes: 3 × 3, 9 × 9, 11 × 11, 13 × 13, 15 × 15, 17 × 17, and the corresponding alpha mattes obtained by the proposed algorithm are shown in Fig. 10 in the last two rows. As it can be observed, the best visual results were obtained for the highest patch sizes.

According to the literature review, common practices in alpha matte extraction are based on handmade linear models, which combine steps and find parameters by hand. Usually, the process begins with the sampling of an image with the purpose of removing some colors and keeping others. The same process can be used interactively through nonlinear and associative computations of neural networks. After selecting samples from a desired image, a neural network learns which pixels are to be removed and which ones stay through association. According to traditional linear methods in the related work, green-screen reflections, inside the desired foreground, are unwillingly removed and then recovered with postproduction techniques. Within the proposed approach, this is solved since green content, reflected on foreground elements, is still kept while background samples are removed. The disadvantage of convolutional neural networks is their dependence on large datasets during training which shows how poorly it performs when training samples are not enough [31]. In the proposed method, sampling is executed with a sparse technique to let the network learn from randomly picked pieces of selected images, which increases variability and conveys generalization of the network.

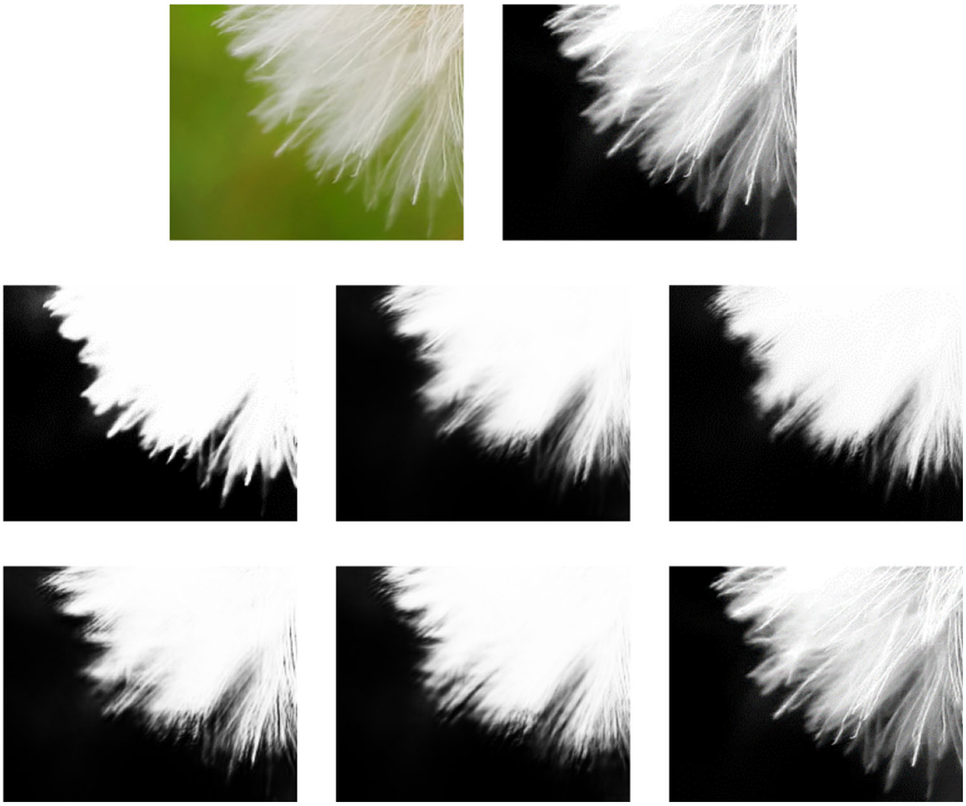


Fig. 10 Original image and ground truth from alphamatting.com (top row) and extracted alpha matte with six different patch sizes (last two rows)

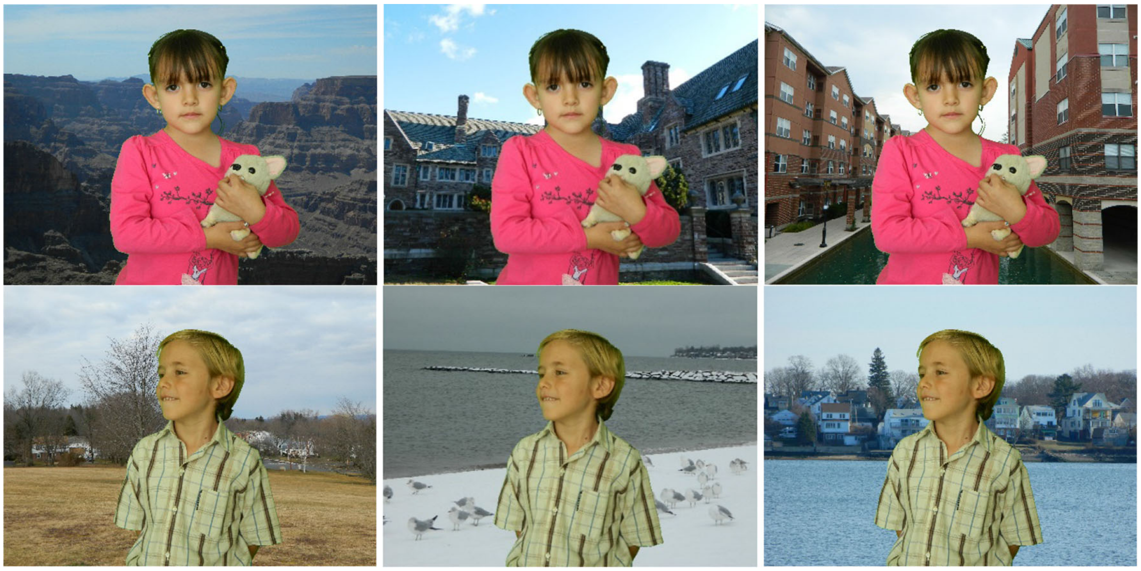


Fig. 11 Composition of extracted foreground over new background scenes without postprocessing

Figure 11 shows the results of compositing an extracted foreground over new background sceneries by using the extracted alpha matte, according to Eq. 1. The first row shows a little girl on three different places: the Grand Canyon, Princeton and the canal of Indianapolis. The second row of Fig. 11 shows a boy superimposed over different outdoor scenarios. Alpha matte extraction and digital compositing are executed automatically. Subsequent processing, such as spill suppression or histogram equalization, can be applied to enhance compositing quality if it is necessary.

5 Conclusions and future works

This paper presents a simple method for fully automatic alpha matte extraction from green-screen images from two different datasets. The method was tested with two high-definition green-screen plate sequences from the public dataset Hollywood Camera Work LLC. The alpha matte extractor successfully separates the foreground from plates on the TOY CAR ROTO sequence without introducing holes. A solution to this problem has only been possible through rotoscoping that is a manual technique to create mattes in the visual effects industry. The extractor also effectively handles alpha matte extraction on plates from the GODIVA MEDIUM sequence.

An ad hoc dataset with 64 images was generated to test the proposed application, which consists of green-screen images captured under conditions that make extraction of alpha matte a challenging task such as situations with uneven illumination, short separation distance between the subject and the green background, presence of eyeglasses, blonde hair and hair filaments. The method produces high-quality results for subsequent digital compositing.

The proposed alpha matte extractor is based on an artificial neural network where learning is achieved through the gradient descent method (backpropagation algorithm). Alpha values are extracted for each image patch. The set of features consists of raw HSV values from the patch and two additional relevant features, a dissimilarity measure and a ratio, both based on RGB values.

Our proposal is a simple learning machine of very low computational cost. It consists of few hidden layers, it is fed with a relatively low number of features, and it does not require a large dataset for training or calibration, and it is trained with one single image which is a collage of patches from one or different foregrounds and a green background.

Besides qualitative results, a quantitative comparison has been performed between alpha mattes extracted from two Hollywood Camera Work LLC HD TOY CAR ROTO and GODIVA MEDIUM green-screen plate sequences by applying the proposed fully automatic method versus the professional application Adobe After Effects CC, which, particularly, requires significant user interaction. Metrics to compare both applications are the average absolute difference and the mean squared difference.

Acknowledgements This research was partially supported by European Grant EU FP7 PEOPLE IRSES (CASES: Cloud computing-based Advisory Services towards Energy-efficient manufacturing Systems).

Compliance with ethical standards

Conflict of interest The authors declare that they have no conflict of interest.

References

1. Chen Q, Li D, Tang CK (2013) KNN matting. IEEE Trans Pattern Anal Mach Intell 35(9):2175–2188
2. Cao G, Li J, He Z, Chen X (2016) Divide and conquer: a self-adaptive approach for high-resolution image matting. In: Proceedings of international conference on virtual real and visualization, China. https://doi.org/10.1109/ICVRV.2016.13
3. Phoka T, Jariyawattanarat W, Sudsang A (2017) Fine tuning for green-screen matting. In: Proceedings of IEEE international conference on knowledge and smart technology, Thailand. https://doi.org/10.1109/KST.2017.7886106
4. Zhu X, Wang P, Huang Z (2017) Adaptive propagation matting based on transparency of image. Int J Multimed Tools Appl. https://doi.org/10.1007/s11042-017-5357-7
5. Rosas-Romero R, López-Rincón O, Rojas-Velázquez ED, Jacobo-Aispuro NP (2016) Learning matte extraction in green-screen images with MLP classifiers and the back-propagation algorithm. In: Proceedings of 26th international conference on electronics, communication and computers, Mexico
6. Gastal ES, Oliveira MM (2010) Shared sampling for realtime alpha matting. Comput Gr Forum 29:575–584
7. Shahrian E, Rajan D, Price B, Cohen S (2013) Improving image matting using comprehensive sampling sets. In: Proceedings of IEEE conference on computer vision and pattern recognition, USA. https://doi.org/10.1109/CVPR.2013.88
8. Al-Kabbany A, Dubois E (2016) Matting with sequential pair selection using graph transduction. In: Proceedings of international symposium on vision, model, and visualization, Germany. https://doi.org/10.2312/vmv.20161349
9. Xu N, Price B, Cohen S, Huang T (2017) Deep image matting. https://arxiv.org/abs/1703.03872
10. Wang J, Cohen MF (2008) Image and video matting: a survey. Found Trends Comput Gr Vis 3(2):97–175
11. Chen X, He F (2016) A propagation matting method based on the local sampling and knn classification with adaptive feature space. J Comput Aided Des Comput Gr. https://arxiv.org/ftp/arxiv/papers/1605/1605.00732.pdf
12. Ruzon MA, Tomasi C (2000) Alpha estimation in natural images. In: Proceedings of IEEE conference on computer vision and pattern recognition, USA. https://doi.org/10.1109/CVPR.2000.855793
13. He K, Rhemann C, Rother C, Tang X, Sun J (2011) A global sampling method for alpha matting. In: Proceedings of IEEE conference on computer vision and pattern recognition, USA. https://doi.org/10.1109/CVPR.2011.5995495
14. Chuang YY, Curless B, Salesin DH (2001) Bayesian approach to digital matting. In: Proceedings of IEEE conference on computer vision and pattern recognition, USA, vol 2, pp 264–271
15. Rhemann C, Rother C, Gelautz M (2008) Improving color modeling for alpha matting. In: Proceedings of British machine conference. https://doi.org/10.5244/C.22.115
16. Levin A, Rav AA (2008) Spectral matting. IEEE Trans Pattern Anal Mach Intell 30(10):1699–1712
17. Wang J (2013) Image matting with transductive inference. Comput Vis Comput Gr Collab Technol 6930:239–250
18. Johnson J, Rajan D, Cholakkal H (2014) Sparse codes as alpha mattes. In: Proceedings of British conference BMVC. https://doi.org/10.5244/C.28.74
19. Wang J, Cohen MF (2005) An iterative optimization approach for unified image segmentation and matting. In: Proceedings of IEEE conference on computer vision, China. https://doi.org/10.1109/ICCV.2005.37
20. Nath D, Chitra P (2015) Image matting based on weighted color and texture sample selection. Biomed Pharmacol J 8(1):331–335
21. Yao G, Zhao Z, Liu S (2017) A comprehensive survey on sampling-based image matting. Comput Gr Forum 36:613–628
22. Sun J, Jia J, Tang CK, Smith HY (2004) Poisson matting. J ACM Trans Gr 23(3):315–321
23. Bai X, Sapiro G (2007) A geodesic framework for fast interactive image and video segmentation and matting. In: Proceedings of IEEE conference on computer vision, Brazil, pp 1–8
24. He K, Sun J, Tang X (2010) Fast matting using large kernel Laplacian matrices. In: Proceedings of IEEE conference on computer vision and pattern recognition, USA. https://doi.org/10.1109/CVPR.2010.5539896
25. Levin A, Lischinski D, Weiss Y (2007) A closed-form solution to natural image matting. IEEE Trans Pattern Anal Mach Intell 30(2):228–242
26. Singaraju D, Rother C, Rhemann C (2009) New appearance models for natural image matting. In: Proceedings of IEEE conference on computer vision and pattern recognition, USA. https://doi.org/10.1109/CVPR.2009.5206491
27. Shen X, Tao X, Gao H, Zhou C, Jia J (2016) Deep automatic portrait matting. In: Proceedings of European conference on computer vision, Netherlands. https://doi.org/10.1007/978-3-319-46448-0_6
28. Cho D, Tai YW, Kweon I (2016) Natural image matting using deep convolutional neural networks. In: Leibe B, Matas J, Sebe N, Welling M (eds) Lecture notes in computer science, vol 9906. Springer, Berlin. https://doi.org/10.1007/978-3-319-46475-6_39
29. Hollywood Camera Work LLC (2017) Green-screen plates. https://www.hollywoodcamerawork.com/green-screen-plates.html Accessed 13 Mar 2018
30. Online benchmark dataset alpha matting (2009) https://www.alphamatting.com. Accessed 20 Feb 2019
31. Wagner R, Thom M, Schweiger R, Rothermel A (2013) Learning convolutional neural networks from few samples. In: Proceedings of joint conference on neural network, USA (2013). https://doi.org/10.1109/IJCNN.2013.6706969