Confidence measure as fuzzy measure in color edge detection

Aureli Soria-Frisch, Abderrahim Kassid

Abstract—A framework for the detection of edges on color edges, which is based on the application of the fuzzy integral, is proposed herein. The framework makes use of the confidence measure of [1] in order to automate the construction of the fuzzy measure coefficients. The computation of the confidence measure is achieved by applying a competitive learning algorithm on the input images, whereby the adaptability of the mentioned algorithm is increased. This is not the only advance attained herein, since the automation of the fuzzy measure's construction furthers the application of the fuzzy integral in computer vision. The framework has been applied in the feasibility study of a system for the reconstruction of frescos. The results in this industrial application are shown together with the performance evaluation on some benchmark images.

I. INTRODUCTION

The simultaneous advances in computing capability and in sensor technology, in terms of cost and reliability of the acquisition devices, allow the successful implementation of multi-sensory computer vision systems. Color systems are a particular case of multi-sensory computer vision systems [2], which process the information delivered by more than one image sensor. The special characteristics of color vision in human beings [3], including e.g. independence of the illumination conditions or hue mutual contrast, challenges the employment of multi-sensory fusion operators in color image processing. In spite of this fact, fuzzy fusion operators, which present a greater flexibility than its hard counterparts [2], is seldom taken into account in the implementation of color processing tasks, and particularly in color edge detection.

Edge detection is an important image transformation involved in numerous applications. Paradigms for image segmentation, pattern recognition, image understanding, edge preserving smoothing, and image compression include the edge detection as a processing stage. In this context, most of the research work has been undertaken on grayvalue edge detection. The interested reader can find a review of proposed approaches in [4].

Usually the detection of edges on color images in real applications is based on the application of an edge detector operating in the grayvalue domain. The first approach takes into account the detection of edges in the three color channels plus the application of a simple fusion operator, e.g. the maximum operator, on the resulting grayvalue edge maps. The other approach takes the initial computation of the intensity image of the three color channels into account. Thence the selected fusion operator is applied on the intensity image. In contrast with these approaches different frameworks based on regularization theory [5], mathematical morphology [6], orthogonal polynomials [7], statistical moments [8], and statistical vector fields [9] have been developed for color edge detection. These methodologies make use of complex mathematical concepts in order to cope with the vectorial nature of color information and simultaneously to reflect the special features of human color vision. On the other hand the here presented framework follows the first mentioned strategy for color edge detection, namely the application of a fusion operator on the grayvalue edge maps of the color channels. However the maximum operator is substituted by a softer fusion operator [2], namely the fuzzy integral. In this context, the framework presented herein is based on a procedure already presented [10].

The fuzzy integral is a fusion operator introduced by Sugeno [11] to simulate the subjective fusion of multi-criteria in human beings. In spite of the extended use of the fuzzy integral in other fields of research, i.e. decision making under uncertainty, the usage of the fuzzy integral in computer vision is very limited. Hence it has been employed as classification procedure in recognition tasks [12][13][14][15][16], as non-linear filter [17][18][19][20], and as operator for the segmentation of color images [21][22][23][24]. This limited usage is due to the lack of methodologies for the automated assessment of the fuzzy measures [25], whose coefficients are used as weights in this fusion operator. For instance, the fuzzy measure coefficients were heuristically set up in the color edge detection procedure based on the fuzzy integral presented in [10]. This general shortcoming, which underplay the properties of the theoretical framework of fuzzy integrals for the implementation of multi-sensory systems, has been partially overcome by the proposal of few methodologies. While some supervised procedures based on genetic algorithms [26][25][27] and perceptron-like networks [12][28] have been successfully proposed for real applications, the definition of unsupervised ones is still an open research question. To the best of our knowledge, entropy-based methodologies [29] and self-organizing networks [24] are the only computational intelligence paradigms within this category that have been proposed.

Taking all these facts into consideration, this paper furthers the framework presented in [10] by developing an unsupervised procedure for the construction of fuzzy measures in color edge detection. This procedure makes use of a so-called confidence measure of a framework for edge detection on grayvalue images [1] in order to determine the values of the fuzzy measure coefficients. The framework presented herein not only makes use of the work in [1] but in some sense generalizes it for its application on color edge detection, as

Aureli Soria-Frisch is with the Technology Dept., Pompeu Fabra University, Pg. Circumval·làció 8, 08003 Barcelona (email: aureli.soria-frisch@ieee.org).

Abderrahim Kassid was with the Dept. Computer Vision and Remote Sensing, Technical University Berlin, Str. des 17. Juni 135, 10623 Berlin, Germany.
it will be shown.

The paper can be outlined as follows. The theoretical background needed in order to follow, which includes a brief description of [1] and of the precedent framework based on the fuzzy integral [10], can be found in section 2. In section 3 the development of the new framework with the procedure for the unsupervised construction of fuzzy measures will be described. The results of its application on some benchmark images and thence on a real application will be presented in section 4. Finally, in section 5, a summary and the conclusions on the presented scheme, and some prospective work will be submitted.

II. USED METHODOLOGIES

The theoretical background of the methodologies, in which the framework presented herein are inspired, is described in this section. The framework results from the automation of the fuzzy measure construction in [10]. For this purpose the confidence measure presented in [1] is used. Not only this aspect but the post-processing proposed in this last framework is undertaken.

A. Edge detection with a confidence measure

As stated by [1] edge detection is one of the basic low-level processing tasks. Therefore its value depends on the capability for the embedding processing system of adapting the results of the edge detection stage. In this context [1] presents a framework where the embedding system can undertake a post-processing of the resulting edge maps by using an associated $\rho \eta$-diagram into account. Once the edge map of the grayvalue input image has been computed, this $\rho \eta$-diagram is generated, where $\rho$ states for the gradient magnitude sorted in ascending order and $\eta$, for the formerly mentioned confidence measure.

The framework in [1] is developed in order to detect edges on grayvalue images by computing the gradient of the pixels. The gradient computation is undertaken by applying as usual some mask-operatots on the input image. Although special differentiation masks are proposed, in our opinion any edge operator that allows the computation of the gradient magnitude $\hat{g}$ and orientation $\theta$ can be adapted.

The generation of the $\rho \eta$-diagram, which represents a two-dimensional histogram, succeeds as follows. The obtained values of $\hat{g}$ are first sorted in ascending order. This becomes the abscise axis $\rho$ of the two-dimensional histogram. Setting the ordinate axis is based on the computation of the $\eta$ values for each pixel. Hence, a $n \times n$ window is extracted from each pixel of the input image an rearranged in vector form as $A$. These pixel sets are thence compared with some edge models, which have been a priori defined, in order to find out the confidence measure $\eta$ of the edge in each pixel. The edge models are defined as $n \times n$ image templates $t$ following the expected grayvalue distribution of an edge in the grayvalue image domain (see Fig. 1 for some examples). The value of $n$ is usually 5 and $M$ templates with angles $\phi \in [-180^\circ, 180^\circ]$ with $1^\circ$ quantization steps are usually used. The comparison succeeds in form of a dot product, which can be mathematically expressed as [1]:

$$\eta = |\text{trace}(A_{ref}A)|,$$

where $A_{ref}$ corresponds to the vector rearrangement of the template $t$. Once the values of the confidence measure are computed, they are sorted in ascending order in the ordinate axis. The $\rho \eta$-diagram is eventually generated by increasing the value of each bin in the two-dimensional histogram when a pair $(\rho, \eta)$ appears in the edge map.

The last stage of the framework [1] implements two post-processing operations based on the computed $\rho \eta$-diagram. These operations, which are called non-maxima suppression and hysteresis thresholding, are undertaken on the resulting edge map in order to filter out the non-significant edges. In this context the non-significance is established by the embedding processing system, i.e. by the later stages of the computer vision application that uses the edge detection procedure. The non-maxima suppression gets rid of those edge pixels, which do not present the maximal value with two pixels of its 8-neighborhood. These two pixels lay in the perpendicular line to the edge direction. The hysteresis thresholding succeeds after defining four different thresholds on the $\rho \eta$-diagram respectively denoted herein as $\rho_{LO}, \rho_{HI}, \eta_{LO}, \eta_{HI}$. The procedure constitutes an extension of the usual hysteresis used in edge post-processing. The interested reader is referred to [1] for a detailed description of the framework implementation, which has not been extensively described herein for the sake of compactness.

B. Fusion for edge detection

In [10] the computation of the color edge maps results from the application of some standard grayvalue edge detectors on the channels of the input color image. After an optional fuzzification, which undertakes the filtering of non-significant edges in an automated manner, the one-channel edge maps are fused through the application of a fuzzy integral. The fuzzy measure coefficients were heuristically established (see Fig. 2).

In that framework the fuzzy integral was used in the $EFus$ module in order to cope with the complexity and uncertainty related to color vision. In this context it is worth mentioning the generalization property of the fuzzy integral, which makes from it a softer fusion operator than those traditionally used [2]. This fact can be exemplary observed in the expression of the Choquet fuzzy integral:

$$C_\mu(x) = \sum_{i=1}^{n} \left[ h_{t(i)}(x_i) - h_{t(i+1)}(x_{i+1}) \right] \cdot \mu(A_{t(i)}).$$

Fig. 1. Image templates $t$ for orientations of $-60^\circ$ (left) and $120^\circ$ (right) as given by [1].
The data from the different information sources, \( x = x_1, \ldots, x_n \), are fuzzified previous to the fusion, where \( h_i \) denote the fuzzifying functions. The fuzzified data is weighted through the fuzzy measure coefficients, \( \mu(A_i) \), where \( A_i \), \( i = 0, \ldots, 2^{n-1} \), stand for the different subsets that can be established over the set of information sources. The parenthesis in the subindices indicate a sorting operation previous to the aggregation, whereby the fuzzy measure coefficients are selected. The interested reader is referred to [30] for a detailed description on this fuzzy fusion operator and to [15][2] for a description of different aspects of its application in computer vision.

As mentioned in the former paragraph, the fusion of the channel edge maps is attained after weighting its importance. The coefficients of the fuzzy measure \( \mu(A_i) \) are thence accordingly adapted. If the optimal result is obtained through the application of a maximum fusion operator, like in the standard procedures (see the Introduction), this result can be achieved as well. This is due to the fact that the maximum operator is a particular case of the fuzzy integral as it can be seen on eq. (2).

One of the shortcomings of the framework presented in [10] is the heuristic construction of the fuzzy measures. Hence, the goodness of the weighting and therefore of the resulting procedure depends on the ability of the user to set the right value of the fuzzy measure coefficients. It is worth mentioning at this point that this operation is not trivial [25]. This shortcoming has been overcome by making use of some concepts proposed in [1].

III. FRAMEWORK WITH UNSUPERVISED FUSION FOR COLOR EDGE DETECTION

The confidence measure as defined by eq. (1), which was presented in [1], is mapped in the framework proposed herein to the fuzzy measure. The realization of this mapping is described in the following section. Not only the automation of the fuzzy measure construction is achieved but the implementation of a post-processing stage as well. This is attained by changing the generation of the \( \rho \eta - \text{diagram} \) in order to allow working with color images. The block diagram of the resulting framework is depicted in Fig. 3.

The functionality of the proposed framework can be identified in this block diagram. \( EDet \) implements an edge detection on the grayscale value image domain through traditionally employed mask-operators, e.g. Haar, Laplace. The edge detection is undertaken on each color channel. The resulting edge maps can be fuzzified in \( Fuzzif \) in order to filter the spurious edges. Thence these edge maps are fused through a fuzzy integral in \( EFus \). For this purpose the previous construction of the applied fuzzy measure succeeds in \( DetFM \) by taking the confidence measure of [1] into account. This procedure is detailed later in this section. On the other hand, the module \( DiagComp \) receives the color image channels and the resulting color edge map. It generates the confidence measure \( \rho \eta - \text{diagram} \). This diagram is eventually used in \( PostProc \) in order to post-process the color edge map. The generalization of the edge detection with confidence and therefore the functionality of \( DetFM, DiagComp, \) and \( PostProc \) are detailed in the following paragraphs.

One of the parts of the edge detection with confidence that can not be generalized into the color image domain corresponds to the definition of the edge templates \( t \). In the grayscale image domain the model of an ideal edge is clear: a transition between the two domain extrema within a small spatial region\(^1\). But in the color image domain the questions spread. What is an edge? But as well, what are the extrema of the color image domain? Is an abrupt transition from red to blue an edge? How many color combinations can be considered to be an edge?

In this context a new approach is defined within the framework presented herein, whereby the edge templates can be generated from the input color image. Furthermore we believe that this adaptive approach is more useful than the definition of \textit{a priori} edge models. Therefore the templates extraction procedure (\( TempExt \)) belongs to both the computation of the confidence measure on the edge maps of the color channels within the module \( DetFM \) and the generation of the \( \rho \eta - \text{diagram} \) within the module \( DiagComp \) as it can be observed in Figs. 4 and 5. First the implementation of \( DetFM \) is elucidated. The extraction of the edge templates \( t (TempExt) \) begins with the random sampling of a set of \( L \) pixels on each of the color channels. This sampling procedure follows a random spatial distribution, which results from applying the Linear Pixel Shuffling (LPS) algorithm [31].

\(^1\)Or maybe not so clear? Definitely stuff for discussion.
Fig. 4. Block diagram for the construction of fuzzy measures in DetFM. 
TempExt: Templates extraction. ConfiComp: Confidence measure computation. The confidence measure on each channel will be used as fuzzy density by $EFus$.

LPS achieves the selection of $L = W \times H \times R/100$ pixels of the input image, which is $W \times H$ large, giving a particular percentage of pixels ($R$). Thence a $5 \times 5$ window centered in the resulting $L$ pixels is extracted and set as the initial pool of edge templates. The next step takes the selection of the $M$ most significant templates among the elements of this initial set into consideration by applying a competitive learning methodology.

The competitive learning attains the determination of the $M$ most significant templates. In order for them not to present any redundancy, the range $[-180^\circ, 180^\circ]$ is divided in $M$ intervals. For each interval the template with the largest magnitude value $\hat{g}$ and the nearest gradient orientation $\theta$ to the interval center is selected. Once the most significant templates have been computed, they are delivered to ConfiComp in order to compute the confidence measure of each edge pixel. This operation succeeds by applying the eq. (1) on each pixel of the input image. Since the operation is repeated on the three color channels, a confidence measure for each channel $(\eta_R, \eta_G, \eta_B)$ and each pixel is obtained. These values are eventually mapped to the fuzzy densities

$$(\eta_R, \eta_G, \eta_B) \rightarrow (\mu_R, \mu_G, \mu_B)$$

, what succeeds therefore for each pixel of the input image. The computation of the fuzzy integral ($EFus$), which is undertaken on each pixel, makes use of the resulting fuzzy measure.

Fig. 5. Block diagram for the computation of the $\rho\eta$ - diagram in DiagComp. TempExt: Templates extraction. ConfiComp: Confidence measure computation. DiagGen: Diagram generation. The diagram is eventually employed in PostProc.

The competitive learning procedure formerly described is applied in the generation of the $\rho\eta$ - diagram (DiagComp) as well. However the templates to be generated in this case are color templates. Therefore some light modifications are undertaken. The color gradient magnitude $\hat{g}$ is defined as:

$$\hat{g} = \sqrt{g_R^2 + g_G^2 + g_B^2},$$

and the color gradient orientation $\theta$ as the average of the gradient orientations on each color channel:

$$\theta = \frac{\theta_R + \theta_G + \theta_B}{3}.$$  

This modifications are a first approach and fuse the gradient features through traditionally used operators. Since both the color templates and the input images present three channels, one confidence measure for each color channels is computed in ConfiComp. Again the average operator is applied in order to find a unique confidence measure $\eta$:

$$\eta = \frac{\eta_R + \eta_G + \eta_B}{3}.$$  

This value is delivered to DiagGen together with $\hat{g}$ in order to generate the $\rho\eta$ - diagram. This succeeds by applying the same procedure described in Sec. II-A and proposed by [1]. The last module PostProc implements the post-processing based on the procedures non-maxima suppression and hysteresis thresholding described in [1] by taking the generated $\rho\eta$ - diagram into consideration. In this way a processed color edge map is computed.

IV. SOME OBTAINED RESULTS

The framework proposed herein was implemented using the Java programming language. A detail of the GUI is depicted in Fig. 6. As it can be observed the application offers the user the possibility to set the following parameters:

- The type of employed fuzzy measure.
- The type of mask-operator for the grayvalue edge detection.
- The type of employed fuzzy measure.
- The number of edge templates $M$ for the computation of the confidence measures.
- The percentage of image pixels $R$ taken into consideration in the templates extraction.
- The threshold for the optional generation of edge binary maps.
- The parameters of the post-processing $\rho_{LO}, \rho_{HI}, \eta_{LO}, \eta_{HI}$.

The GUI allows to work with different images, which can be loaded through the menu.

Fig. 6. GUI of the framework proposed herein, which was implemented as a Java application.

Different tests have been undertaken with the implemented framework. Their results are given in this section. First
different benchmark images\textsuperscript{2} have been processed. Thence the images of a real application for the reconstruction of frescos have been used within the framework.

A. Benchmark images

Benchmark images have been employed in order to prove the general functionality of the framework and to facilitate the comparison of the obtained results with those of other frameworks for color edge detection. A Haar-operator has been used for the edge detection in the grayvalue image domain. Moreover values $M = 8$ and $R = 10\%$ were set in the computation of results. The variable parameterization of the obtained results, which refers to the hysteresis thresholding, is given in Tab. I.

<table>
<thead>
<tr>
<th>Fig.</th>
<th>$\rho_{LO}$</th>
<th>$\rho_{HI}$</th>
<th>$\eta_{LO}$</th>
<th>$\eta_{HI}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>house</td>
<td>0.46</td>
<td>0.42</td>
<td>0.95</td>
<td>0.92</td>
</tr>
<tr>
<td>lake</td>
<td>0.75</td>
<td>0.68</td>
<td>0.85</td>
<td>0.80</td>
</tr>
<tr>
<td>lenna</td>
<td>0.82</td>
<td>0.76</td>
<td>0.85</td>
<td>0.80</td>
</tr>
<tr>
<td>peppers</td>
<td>0.90</td>
<td>0.84</td>
<td>0.58</td>
<td>0.53</td>
</tr>
</tbody>
</table>

The obtained results are shown in the following figures together with the computed $\rho_{\eta}$ diagram for each image.

B. Edge detection in fresco’s reconstruction

The framework is finally applied on the images taken in order to solve a problem of image reconstruction. The depicted pieces of a fresco have to be virtually reconstruct in order to obtained a computer reconstruction of the original fresco. In this context the edge detection is one of the low-level operations needed in the embedding system. The obtained results are shown in Fig. 9. As it can be observed the pre-processing of the framework proposed herein increases the performance of the system. Hence, some spurious edges due to some effects of the protecting plastic and to shadows can be avoided through the applied post-processing. In this context the framework delivers some color edge maps, where just the edges among color regions are visible. This is very important for the embedding processing system.

V. Conclusions and Projective Work

A new framework has been proposed for the detection of color edges. The framework is based on the fuzzy integral. The framework successfully attains the automated construction of fuzzy measures by applying a confidence measure as the needed fuzzy densities. Furthermore the computation of the confidence measure is achieved through a competitive learning algorithm, which allows its computation on color images. The positive features of the resulting methodology have been shown on hand of two different types of images.

\textsuperscript{2}Available at the USC-SIPI Image Database \url{http://sipi.usc.edu/services/database/Database.html}.

Future works will attain the characterization of the framework and the analysis of its robustness w.r.t. the employed parameters. The application of different competitive learning algorithms will be analyzed within the framework proposed herein. Other applications will be implemented as well.

REFERENCES

Fig. 8. Results of the framework proposed herein on the lenna (left) and the peppers (right) images. (center) Plot of the computed \( \rho_n \) – diagram. (bottom) Resulting color edge map.


