Neurocomputing 194 (2016) 308–316

Contents lists available at ScienceDirect

Neurocomputing

journal homepage: www.elsevier.com/locate/neucom

Stereo matching based on classification of materials

Menglong Yang [a], Yiguang Liu [b,*], Ying Cai [b,c], Zhisheng You [b]

[a] School of Aeronautics and Astronautics, Sichuan University, Chengdu 610064, PR China
[b] School of Computer Science and Engineering, Sichuan University, Chengdu 610064, PR China
[c] School of Information Engineering, Sichuan Agricultural University, Yaan 625014, PR China

ARTICLE INFO

Article history:
Received 9 July 2015
Received in revised form
1 February 2016
Accepted 17 February 2016
Available online 4 March 2016

Keywords:
Stereo matching
Classification of materials
Fusion move

ABSTRACT

Stereo matching is one of the most important and fundamental topics in computer vision. Encouraging self-similar pixels to be assigned to the same label has been proved to be effective for stereo. A typical way of taking advantage of self-similarity is performing a color segmentation on the image and motivating the pixels within each segment to share an identical label. However, some cases cannot be handled by image segmentation, such as the pixels in disconnected regions. This paper proposes a stereo method based on the assumption, that a 3D scene is a collection of a few smooth surfaces and a few classes of reflective materials, such that the 3D points belonging to an identical material are likely to lie on a small number of surfaces and the 3D points lying on a single surface belong to a few classes of reflective materials. Each material is expected to have specific albedo properties. This paper presents two methods for classifying the albedo properties depending on whether the illumination environment is known, without recovering the albedo parameters. The proposed model is formulated as an energy function incorporating some new priors, that is optimized via fusion move algorithm.

1. Introduction

Stereo is one of the most fundamental topics in computer vision. Stereo vision attracts many researchers and has been widely researched since there was the publicly available performance testing such as the Middlebury [1] and KITTI stereo benchmark [2], which allow researchers to compare their algorithms against all the state-of-the-art algorithms.

Different from the feature matching [3,4], which matches sparse feature points in two images, the stereo matching can densely match the pixels. The stereo algorithm presented in this paper uses the popular energy minimization framework that is the basis for most high-performance algorithms such as graph cuts [5–7] and belief propagation (BP) [8–12]. The stereo is achieved in these algorithms, essentially, by solving a Markov Random Field (MRF) model, including the assumptions of photo-consistency (i.e. a 3D point should look similar in both views) and smoothness (i.e. the neighborhood pixels have close disparities).

Besides these assumptions, we believe that a 3D scene is a collection of a few classes of reflective materials, such that each class of the material has specific albedo properties, and that 3D points, which belong to the same material, are likely to lie on a small number of surfaces, even on a single 3D surface. Moreover, there are a few 3D smooth surfaces in the scene, such that 3D points, which lie on the same surface, belong to a few classes of materials. The energy function and the optimization algorithm in this work is formalized based on this idea.

Nearly all of the top-ranked methods [8,13–15] in the Middlebury benchmark use image segmentation to encourage the pixels within a single segment to share an identical label or have smooth disparities. However, the case that the pixels in disconnected regions, like Fig. 1, cannot be handled by image segmentation. In contrast, in this work, we use the prior, that the pixels belonging to the same material are preferred to lie on a small number of surfaces to motivate these separated pixels to have a small number of labels, or even a single label.

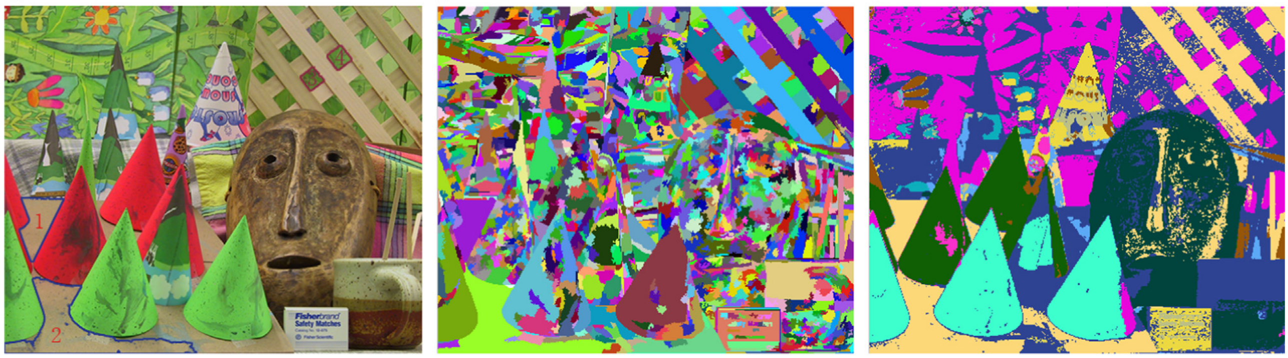
2. Related work and contributions

Encouraging self-similar pixels to be assigned to the same label is effective for stereo. A typical way of taking advantage of self-similarity is performing a color segmentation on the image and regarding the pixels within each segment as self-similar pixels. Some segmentation-based algorithms use a hard constraint that the pixels within a single segment must be assigned to the same plane [16,8,17], and some others use a soft constraint that the neighboring two pixels sharing the same segment are only encouraged to lie on the same plane [18,15]. The scale of segmentation is important for these algorithms. A large scale of segmentation over-constrains the

* Corresponding author.
E-mail address: lygpapers@aliyun.com (Y. Liu).

http://dx.doi.org/10.1016/j.neucom.2016.02.049
0925-2312/

a b c



Cones reference image | Color segmentation result | Our material classification result

Fig. 1. An example of disconnected self-similar regions, e.g. regions 1 and 2 of the table on the bottom left or 3 and 4 on the green background on the top right in (a). The traditional segmentation based methods (b) do not encourage these disconnected self-similar pixels to be assigned to the same label, while our work uses material segmentation (c) to motivate these pixels to lie on the same surface. (For interpretation of the references to color in this figure caption, the reader is referred to the web version of this paper.)

small objects and a small scale of segmentation is hard to constrain the large objects. Bleyer et al. [14] use a soft segmentation term to encourage the stereo result to consistent with a precomputed segmentation, but optimizing these higher-order cliques is difficult and time consuming. To make a tradeoff, we adopt a smoothness penalty for each pair of neighboring pixels that depends on multi-scale segmentations and classifications of materials. If two neighboring pixels belong to a single material, and they are always in the same segment with different segmentation scales, they are naturally expected to lie on the same surface.

Another way of handling self-similarity is employing color models. In [13], an oversegmentation of one input image is refined using single Gaussians to model color within each small segment, but the color models are not used during matching and the method cannot handle small disconnected background regions in the presence of complex occlusions. In [19], the authors present a method for bilayer segmentation of stereo input into foreground and background regions, where color models are also used. This work is extended in [15], where the authors segment the scene into an arbitrary number of 3D objects and not just two. Each object has a specific color model including a compact set of colors. The object segmentation and the color models rely on each other. Most pixels on the same object are spatially clustered. To handle small disconnected background regions in the presence of complex occlusions, they design an occlusion reasoning. It concerns a 3D connectivity constraint connectivity prior, which is hard to maintain.

However, in fact, the pixels on an object may have different colors (it is solved by Gaussian Mixture Model in [15] where the number of Gaussians need to be specified by users), and the pixels on different objects may have the same color. The most distinct difference to our work is that we do not use the one-to-one relationship between the objects and the color models. We segment the scene into a few surfaces and materials, where each surface contains a few materials, and each material occurs on a few surfaces. Moreover, the material classification results can provide robust cues for stereo than pixel color, because the pixel appearance is influenced by many aspects, such as albedo properties and illumination environment. There is great potentiality for material classification. In this work, we present two methods of material classification depending on whether the illumination environment is known. In addition, material classification provides the possibility for combining the stereo and shadow estimation.

The main contributions of this paper are summarized as follows. Firstly, to avoid use a higher-order prior like [14], a soft constraint that encourages spatially neighboring pixels to lie on the same 3D surface is achieved by simply defining the penalty constant depending on whether the neighboring pixels belong to the same material and whether they are in same segments with different segment scales.

Secondly, we use two global MDL (minimum description length) priors to penalize the number of surfaces and materials. Furthermore, we use a local MDL prior to motivate the pixels on the same surface belong to a small number of materials and the pixels belonging to the same material to lie on a small number of surfaces.

Thirdly, each material is expected to have specific albedo properties. This paper presents two methods for classifying the albedo properties depending on whether the illumination environment is known, without recovering the albedo parameters.

3. Model

In this paper, we represent the scene as a collection of surfaces like [14], in which some geometrical properties (such as the curvature and normal vector for each pixel) can be easily described without higher-MRF model. Following the notation in [14], let $\mathcal{I}$ denote the pixels of the reference image and $\mathcal{F}$ denote the set of all 3D surfaces. Further, let $\mathcal{M}$ denote the set of the possible reflectance materials. We need search two mappings: (1) $F: \mathcal{I} \rightarrow \mathcal{F}$ that assigns each pixel $p \in \mathcal{L}$ to a surface $f_p \in \mathcal{F}$, (2) $M: \mathcal{I} \rightarrow \mathcal{M}$ that assigns each pixel to a material.

In fact, the pixels belonging to each class of materials contain a specific group of similar albedo properties. In other words, assigning a pixel to a material implicitly assigns the pixel's albedo properties. Furthermore, assigning a pixel to a surface implicitly assigns the pixel's disparity, which can be computed according to the surface function, e.g., if a pixel p is assigned to a depth plane f_p, the corresponding disparity can be computed by $d_p = f_p[a] \cdot p_x + f_p[b] \cdot p_y + f_p[c]$ where a, b and c are plane parameters and p_x, p_y are p's image coordinates. Like [14], we use two types of surfaces, i.e. plane and thin-plate smoothing spline surface.[1]

The quality of a mapping F and M is evaluated by an energy function $E(F,M)$ defined as

$$E(F,M) = E_{data}(F) + E_{smooth}(F,M) + E_{ap}(M) + E_{smdl}(F) + E_{mmdl}(M) + E_{msmdl}(F,M) \quad (1)$$

where each term is described as follows.

[1] We use the Matlab command *tpaps* with the smoothing parameter $p = 0.5$ to generate the candidate spline surfaces.