INFORMATICA, 2015 Vol. 0, No. 0, 1

EXPLOITING SPATIO-TEMPORAL INFORMATION FOR LIGHT-PLANE LABELING IN DEPTH-IMAGE SENSORS USING PROBABILISTIC GRAPHICAL MODELS *

Jaka KRAVANJA[1], Mario ŽGANEC[1], Jerneja ŽGANEC-GROS[1],
Simon DOBRIŠEK[2], Vitomir ŠTRUC[2],

[1]Alpineon d.o.o., Ulica Iga Grudna 15,SI-1000 Ljubljana, Slovenia
[2]Faculty of Electrical Engineering, University of Ljubljana, Tržaška cesta 25, SI-1000 Ljubljana, Slovenia
E-mail: jaka.kravanja@alpineon.si

Abstract. This paper proposes a novel approach to light plane labeling in depth-image sensors relying on "uncoded" structured light. The proposed approach adopts probabilistic graphical models (PGMs) to solve the correspondence problem between the projected and the detected light patterns. The procedure for solving the correspondence problem is designed to take the spatial relations between the parts of the projected pattern and prior knowledge about the structure of the pattern into account, but it also exploits temporal information to achieve reliable light-plane labeling. The procedure is assessed on a database of light patterns detected with a specially developed imaging sensor that, unlike most existing solutions on the market, was shown to work reliably in outdoor environments as well as in the presence of other identical (active) sensors directed at the same scene. The results of our experiments show that the proposed approach is able to reliably solve the correspondence problem and assign light-plane labels to the detected pattern with a high accuracy, even when large spatial discontinuities are present in the observed scene.

Key words: Depth images, structured light, probabilistic graphical models, spatio-temporal information.

1. Introduction Depth-image acquisition is a field with many areas of application that range from reconstructing the shapes of statues, medical imaging and biometry to obstacle detection and other similar areas. Particularly interesting and simple to implement are the active-triangulation techniques that rely on the projections of structured light. Much research effort is being directed at improving structured-light approaches, especially with respect to outdoor usage (see, e.g., (Mertz, 2012)), where most of the existing solutions struggle with their performance.

Structured-light approaches to acquiring depth images typically work by projecting a pattern of structured light onto the observed scene and analyzing the

*This research was supported by the European Union, European Social Fund, within the scope of the framework of the Operational Programme for Development of Human Resources in the Period 2007–2013, contract no. 3211-11-000492 (ATRIS).

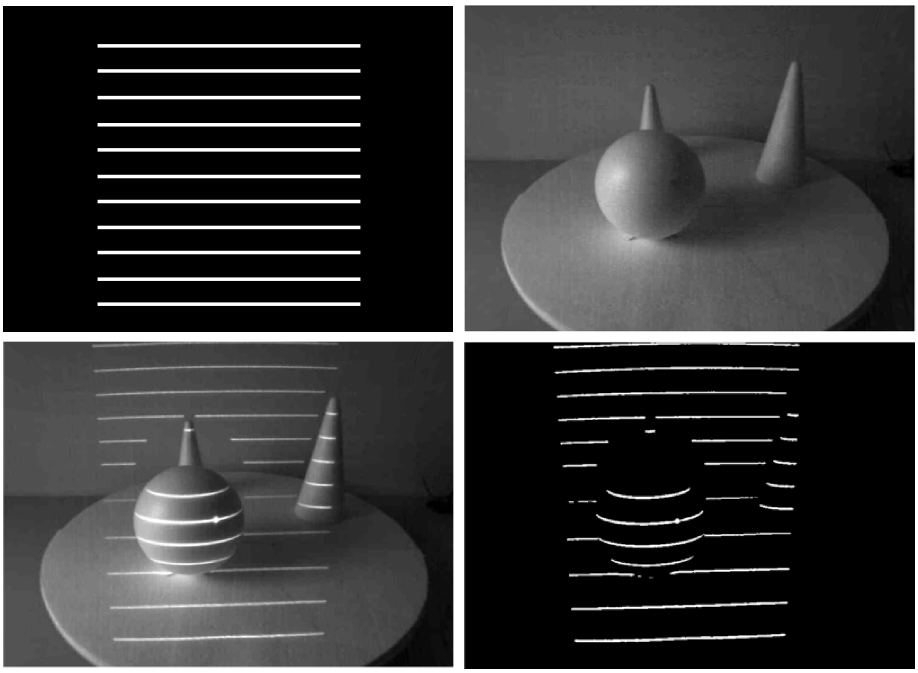


Fig. 1. Illustration of the correspondence problem: shape of the projected pattern (upper left), target scene (upper right), projected pattern superimposed on the target scene (lower left), deformed detected pattern (lower right).

deformations of the projected pattern to recover the shape (and reveal the depth information) of the observed scene. Clearly, an integral part of such an approach is the procedure for establishing the correspondence between the projected pattern and the pattern that is detected by the imaging sensor due to interactions with the observed scene. The difficulty associated with this problem is illustrated in Fig. 1. Here, the upper-left image depicts the structure of the projected pattern, the upper-right image shows the observed scene, the lower-left image depicts the scene with the superimposed light pattern and the lower-left image shows the deformed pattern detected by the imaging sensor. To be able to recover the shape of the observed scene, a one-to-one correspondence between all the parts of the projected and detected patterns need to be established.

To make the outlined problem simpler, the majority of structured-light approaches rely either on color-coded projection patterns (Albitar, 2009), (Boyer, 1987), (Koninckx, 2004), (Ulusoy, 2010), (Zhang, 2002) width-coded projection patterns (Beumier, 1999), time-dependant sequences (Curless, 1995), (Daley, 1998) or other coding techniques (Salvi, 2004). As argued in (Brink, 2008) and (Robinson, 2004), the presented approaches also exhibit some limitations: color cannot be applied consistently to surfaces with weak or ambiguous re-

flectance and limits the use of potentially useful optical filters on the camera side, for width coding the resolution is less than for uniform parallel stripes and so forth. Considering the presented limitations, it seems appealing to use projection patterns consisting of uncoded (i.e., homogeneous) parallel stripes, for which the term uncoded structured light is often used in the literature (Brink, 2008), (Robinson, 2004).

In this paper we present a technique for solving the correspondence problem between the projected and detected patterns of uncoded structured light. As will be shown in the paper, the problem corresponds to a light-plane-labeling problem that can effectively be solved using probabilistic graphical models. With the proposed approach a graphical model is first constructed by defining local (spatio-temporal) relationships between the parts of the detected light pattern. Inference on the graph is then conducted through fractional belief propagation.

The developed labeling technique is applied to images of detected light patterns generated with our imaging sensor (Volkov, 2013), (Žganec, 2009), which was already shown to be capable of robustly working in outdoor environments and, therefore, represents a major step towards robustifying structured-light approaches for outdoor use. Note that even existing commercial products, such as Microsoft's Kinect, which relies on speckle-pattern projection, have difficulties operating in outdoor environments.

To summarize, the following novelties are presented in the paper:

- a technique for building graphical models from the detected, uncoded, structured-light patterns,
- a technique for solving the problem of correspondence between the projected and detected patterns based on probabilistic graphical models,
- the inclusion of temporal information into the process of solving the correspondence problem.

The rest of the paper is structured as follows: In Section 2 we briefly present the imaging sensor used in the paper, formally define the correspondence problem and describe the preprocessing technique applied to the detected light pattern before constructing the graphical model. In Section 2.2 we introduce the new labeling procedure that relies on spatio-temporal information and probabilistic graphical models and show how it can be used to solve the correspondence problem. We evaluate the proposed approach in Section 4 and conclude the paper in Section 5.

2. Prerequisites In this section we present the background needed to understand the proposed labeling procedure that is presented in the remainder of the paper. We commence the section by formally defining the labeling problem we are trying to solve. We then proceed by briefly describing the depth-image sensor that is used to generate images of the projected patterns that form the basis for our work and conclude the section by presenting the basic pre-processing techniques

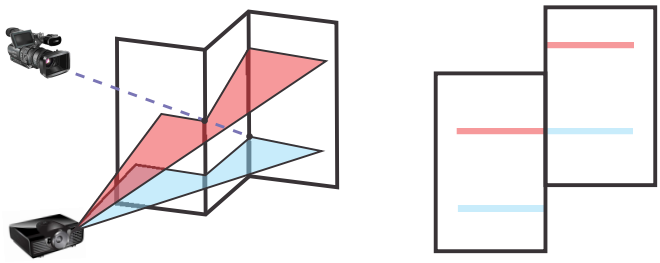


Fig. 2. Illustration of the problems typically encountered when solving the correspondence problem between the light planes of the projected pattern and the line segments of the detected pattern.

applied to the detected light patterns and defining some of the terminology used in the paper.

2.1. The labeling problem Structured-light approaches to depth-image acquisition work by projecting a structured light pattern onto a scene and analyzing the deformations of the projected pattern due to interactions with the scene. Based on these deformations a depth map of the observed scene can be reconstructed, but only under the condition that a correct correspondence between the projected and detected patterns is established (Salzmann, 2007). In our case, where the (uncoded) structured-light pattern consists of parallel light planes, this corresponds to finding the correct light-plane label for each part of the detected light pattern. While this task seems trivial at first, it is in fact quite complex, especially when depth discontinuities are presented in the observed scene. Fig. 2 illustrates the difficulties that are often encountered when trying to establish the correspondence. If the scene contains depth discontinuities (left side of Fig. 2), the projected pattern is deformed and what is detected is a light pattern where the line segments belonging to different projected light planes may form a single line or other (similar) ambiguities are introduced into the detected pattern.

Formally, the correspondence problem can be defined as follows. Assume that the detected light pattern (shown in Fig. 3) is represented as a binary image I; that is, pixels representing scene points illuminated by the projected pattern are encoded with a value of one, while all the other pixels are encoded with a value of zero. Let us denote with $\mathcal{B} = \{b_1, b_2, b_3, ..., b_N\}$ the set of all non-zero pixels in image I and let N stand for the number of such pixels.[2] Furthermore, let us denote the set of indices of the light planes constituting our light pattern as $\mathcal{I} = \{1, 2, ..., k, ..., M\}$, where M stands for the number of parallel light planes, and the index 1 denotes the plane that is the lowest in the projected pattern. The correspondence problem can then be defined as a mapping that assigns each pixel

[2]Note that in our case the set $\mathcal{B}$ actually represents a set of pixel-clusters.

from $\mathcal{B}$ an index from $\mathcal{I}$; that is:

$$\psi : b_i \rightarrow \mathcal{I}, \text{ for } i = 1, 2, ..., N. \tag{1}$$

It is clear that the correspondence problem actually represents a labeling problem, where each non-zero pixel in the detected light pattern needs to be assigned a light-plane label.

2.2. The imaging sensor The imaging sensor used in this work consists of two parts, i.e., a pattern projector in the form of a laser and a high-speed camera, of which both are attached to a rigid structure and are capable of working in synchronization. The optical center of the camera and projector lie on the same vertical plane. The camera produces a special pseudo-random binary (on/off) sequence that drives the laser. When the sequence is in the "on" state, the laser projects the light pattern onto the scene and, similarly, when the sequence is in the "off" state, the laser emits no pattern. The camera acquires (sub-)images of the scene on each clock signal, regardless of the state of the pseudo-random sequence (on or off). Using the sequence of sub-images acquired by the camera, the final output image is produced by de-multiplexing the sequence of sub-images with the multiplexing signal. In other words, the final image is constructed as a normalized superposition of the entire sequence of sub-images, where the sign in the sum is determined by the multiplexing signal: 1 (on) is a positive sign and 0 (off) is a negative sign. In this way, the images that contain the projected pattern will have a positive sign and the images without the pattern will have a negative sign (see Fig. 3). After the summation, the projected pattern on the image is emphasized and the background is suppressed (Žganec, 2009). A similar background-substraction imaging approach was proposed in (Mertz, 2012), with the difference being that the acquisition step is not modulated, which can be prone to disturbances, especially if two such sensors are deployed in the same environment. Note that different implementations are possible to speed up the background subtraction, as for example (Hung, 2014).

As demonstrated in (Volkov, 2013), the described sensor has several merits, such as the capability to work in the presence of other identical sensors as well as in outdoor environments. Note that an experimental demonstration of the merits of the sensor is beyond the scope of this paper. The interested reader is referred to (Volkov, 2013) for more information.

2.3. Image pre-processing The problem formulation presented in Section 2.1 assumes that the detected light pattern represents a binary image. However, our sensor produces a grey-scale image that needs to be pre-processed to emphasize the projected patterns and to make them more suitable for the task of solving the correspondence problem. A local thresholding procedure is, therefore, first applied to the pattern image to produce its binary version (Gonzales, 2008). The generated binary image is then processed to generate what we refer to here as

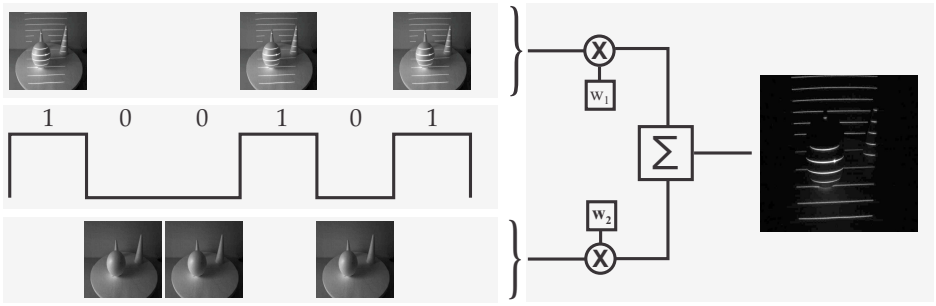


Fig. 3. Image demultiplexing. Upper part: Sub-images acquired when the illumination pattern is projected onto a scene. Middle part: Multiplexing signal. Lower part: Sub-images acquired when the illumination pattern is not projected onto the scene. Right side: Schematics of constructing the final image. Sub-images containing the projected pattern are multiplied by $w_1 = 1$, whereas sub-images without the projected pattern are multiplied by $w_2 = -1$.

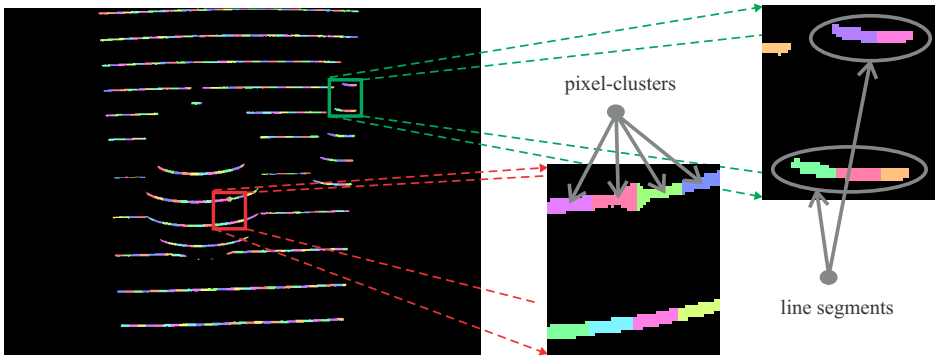


Fig. 4. Color-coded binary image after preprocessing (best viewed in color).

"pixel-clusters", which in our case are nothing more than line segments in the binary image partitioned into smaller pieces, each spanning a certain number of columns. Some examples of these pixel-clusters are shown in Fig. 4.

In the next section we show how the pixel-clusters can be assigned random variables and used to build a graphical model. Based on the constructed graphical model, the mapping ψ from Eq. (1) can then be easily determined.

3. Light-plane labeling with probabilistic graphical models In this section we build on the ideas introduced in (Ulusoy, 2010) and present the labeling problem in the form of a probabilistic graphical model (PGM). The use of PGMs for solving the labeling problem is reasonable, as the theoretical framework and formalism associated with PGMs allows us to describe complex problems in a concise way by partitioning them into smaller and simpler parts (Vesnicer, 2008). In the case of

our labeling problem, this translates to describing the relationships and dependencies between pairs of pixel-clusters based on spatial/temporal/prior information, and using the constructed dependency chain for inferring the labels of all the parts of the (global) projected and deformed light pattern.

3.1. Problem formulation with PGMs In general, graphical models consist of a set of vertices $\mathcal{V}$ and a set of edges $\mathcal{E}$ connecting those vertices, which together form a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$. In our case, each pixel-cluster in the detected pattern is represented as a vertex $v \in \mathcal{V}$, while the interdependencies between the pixel-clusters are represented as the edges $e \in \mathcal{E}$ of the graph.

Each pixel-cluster and thus also each vertex corresponds to a discrete random variable X, while the set of all N random variables of a given input image I at time instance t is defined as $\mathcal{X}^t = \{X_1^t, X_2^t, ..., X_N^t\}$. The domain of the random variables is defined by $\mathcal{I}$. It is trivial to see that determining the value of each random variable from $\mathcal{X}$ is equivalent to solving the labeling problem defined by Eq. (1).

To illustrate how the PGM framework is used for modeling the relationships between pixel-clusters in this paper, let us for a moment examine the two simplified patterns on the left hand side of Fig. 5, acquired at the time instances $t-1$ and t. Each of the two detected patterns consists of three pixel-clusters that belong to two distinct light planes. As can be seen from the right hand side of Fig. 5, where the PGM constructed on the basis of our modeling approach is presented, the state (or value) of each random variable (i.e., each pixel-cluster) is modeled to depend on its horizontal neighbors, vertical neighbors, temporal neighbors, and prior knowledge about the structure of the projected pattern. The dependencies between the neighboring pixel-clusters are defined by so-called factors, which model the relationships between the random variables, and for the horizontal, vertical, temporal and prior cases are denoted as ϕ_h, ϕ_v, ϕ_t, and ϕ_p, respectively.

For our modeling approach, the joint probability distribution of the PGM can thus be written as being proportional to the following factor product:

$$p(\mathcal{X}^{t-1}, \mathcal{X}^t) \propto \prod_{\substack{t'=t-1 \\ (i,j)\in\mathcal{E}_h}}^{t} \phi_h(X_i^{t'}, X_j^{t'}) \prod_{\substack{t'=t-1 \\ (i,j)\in\mathcal{E}_v}}^{t} \phi_v(X_i^{t'}, X_j^{t'}) \prod_{(i,j)\in\mathcal{E}_t} \phi_t(X_i^{t-1}, X_j^t) \prod_{\substack{t'=t-1 \\ i=1}}^{t,N} \phi_p(X_i^{t'}). \quad (2)$$

where N can, in general, take different values at different time instances and could also be written as $N^{t'}$ in the above equation. The sets $\mathcal{E}_h$, $\mathcal{E}_v$, and $\mathcal{E}_t$ correspond to the subsets of all the edges $\mathcal{E}$, on which the horizontal, vertical and temporal factors are defined, respectively. Note that the above joint distribution is defined

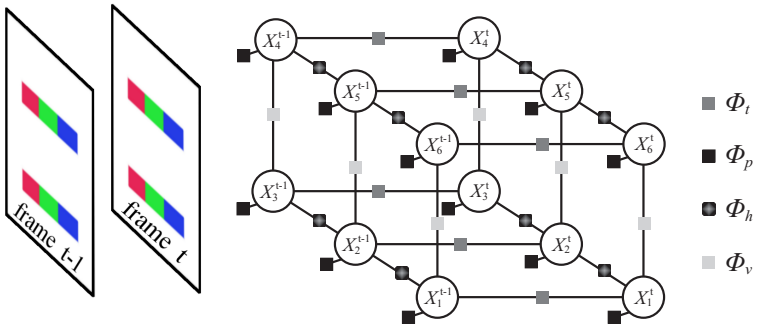


Fig. 5. Illustration of the construction procedure for the PGM: simplified light pattern (left), corresponding PGM (right).

for the case of two consecutive frames (at time instances t and $t-1$). The extension to a longer sequence is straightforward.

After the graph and the factors have been constructed, the optimal configuration of the graph can be found using various inference algorithms, such as a brute force search for exact inference, loopy belief propagation (Kschischang, 2001), or fractional belief propagation (Wiegerinck, 2003), which has also been used in our experiments (Mooij, 2010). Once the state of each random variable has been determined, the pixel-cluster-to-light-plane correspondence is known.

3.2. Building the graph The example shown in Fig. 4 represents a simple toy setting, where two line segments at time instances t and $t-1$ are perfectly aligned, as are the three pixel-clusters comprising the segments. In practice this is unfortunately very rarely the case. Since no specific topology is present in the projected pattern that could serve as a reference point for defining the spatial relationships between the pixel-clusters, we need to define the criteria for defining vertical, horizontal and temporal neighbors that can be used for building our graph.

Here, the simplest and most obvious criterion applies to the horizontal neighbors. In our modeling approach two pixel-clusters are considered to be horizontal neighbors if they are adjacent in the horizontal direction and they represent connected binary regions[3]. Horizontal neighbors are needed in our PGM to "encourage" horizontally adjacent pixel-clusters to take the same label. Most commonly, each pixel-cluster has two horizontal neighbors, but any other number is possible as well, even though it is less likely.

Vertical neighbors are defined as pairs of pixel-cluster that contain at least one pixel at the same x-coordinate (and different y-coordinates). There can again be several vertical neighbors for a given pixel-cluster, and having more than just two vertical neighbors is in fact the most common setting for a given pixel-cluster.

[3]Where a 8-adjacency is considered for the connectivity (Gonzales, 2008).

Vertical neighbors are needed in our PGM to ensure that the detected light planes tend to be labeled consecutively and as such are extremely important for our modeling approach.

Last but not least, temporal neighbors are defined as pixel-clusters from the pattern images of two consecutive time instances that share at least one non-zero pixel at the same spatial coordinates. This definition requires no tracking of the pixel-clusters over time and is, therefore, simple to implement. Temporal neighbors are included in our modeling approach to exploit the additional temporal information when labeling the light planes of the projected pattern and, as will be shown in the experimental section this is indeed useful for the labeling accuracy.

The presented definitions define the topology of the PGM (i.e., vertices and edges) constructed from the given input image I. To be able to conduct inference on the graph, we need to define the factors between pairs of neighboring vertices (or on a single vertex) that model the dependencies between the random variables of the vertices. The procedure for defining this factor used in this paper is described in the next section.

3.3. Defining the factors As emphasized in the previous section, factors represent functions of random variables and are typically used to model the dependencies between the neighboring vertices (i.e., random variables) or to include prior knowledge into the PGM, in which case they apply only to a single vertex. In the simple toy example in Fig. 5 the factors are represented by small squares. In our modeling procedure we defined the factors to model the relationships between horizontally, vertically and temporally neighboring pixel-clusters and added unary [4] prior factors to include prior knowledge about the pattern structure into the modeling procedure.

Horizontal factors ϕ_h are assigned between variables that correspond to horizontally neighboring pixel-clusters. Depending on the values of both random variables, the factor determines a fitness value. This fitness is high if the variables take the same value and small if their values differ, which forces the horizontal neighbors to tend to correspond to the same projected plane of light. The fitness assigned by the factor is defined by Eq. (3).

$$\phi_h(X_i^t = k, X_j^t = k') = \begin{cases} 1, & k = k', \\ f_c, & \text{else} \end{cases}, \tag{3}$$

where $k, k' \in \mathcal{I}$, and f_c $(0 < f_c < 1)$ denotes a fraction cost parameter that tells us how much the fitness is diminished if two horizontally neighboring pixel-clusters take a different value.

Vertical factors ϕ_v are assigned between the variables whose corresponding pixel-clusters overlap vertically. This factor tends to assign a high fitness when the two vertically neighboring pixel-clusters come from two neighboring light planes.

[4] Factors of a single random variable.

If this is not the case, the fitness drops accordingly. Eq. (4) summarizes the fitness determined by the vertical factors:

$$\phi_v(X_i^t = k, X_j^t = k') = \begin{cases} f(k-k'), & k > k' \\ 0, & \text{else} \end{cases}, \tag{4}$$

where $k, k' \in \mathcal{I}$ and f stands for a linear function between the difference of two label indices. The function f decreases monotonically with the difference:

$$f(\delta) = \begin{cases} g(1-(\delta-1)h), & \delta \neq 0 \\ o_c, & \text{else} \end{cases}. \tag{5}$$

The constant h determines the function drop rate, o_c (overlap cost) denotes a parameter penalizing two vertically overlapping pixel-clusters taking the same variable value, and the function $g(.)$ represents a truncating function that truncates all the negative values to zero. Ideally, pixel-clusters originating from the same projected light plane never overlap vertically, but due to system imperfections and lens distortions some vertically overlapping pixel-clusters can take the same value. This is why it should be discouraged, but allowed at high cost.

Temporal factors ϕ_t are assigned between the pixel-clusters belonging to two consecutive frames. Given a sufficiently high frame rate, the pixel-clusters in the temporal neighborhood should preserve the same correspondence and, hence, also the same variable state. The temporal factor is a function that assigns high fitness if the pixel-clusters take the same value and zero fitness if the values differ.

$$\phi_t(X_i^{t-1} = k, X_j^t = k') = \begin{cases} 1, & k = k' \\ 0, & \text{else} \end{cases}, \tag{6}$$

where $k, k' \in \mathcal{I}$.

The prior factors ϕ_p are assigned to all the vertices and operate on a single random variable. They are used to incorporate prior knowledge about the spatial structure of the projected pattern into the modeling procedure and, in a sense, carry information about the most likely range of values a random variable can take with respect to the vertical position of the pixel-clusters and the number of vertical neighbors above and below. The prior factors are determined based on the pseudo-procedure shown in Algorithm 1.

4. Experimental assessment

4.1. Experimental setup and performance measurement In order to evaluate the proposed approach we construct a database of 152 binary images of a dynamic scene. The scene is illuminated with the uncoded structured-light pattern generated by our sensor, which in the current form consists of eleven parallel light planes[5], i.e., $M = 11$. The scene contains three objects (two cones and a ball)

[5]Note that the number of parallel light planes the sensor emits can be adjusted depending on the needs of the application that uses the sensor. Our current prototype features eleven light

Algorithm 1: Determining prior factors

```
for all pixel-clusters (i.e., rand. var. X_i) in the image I do
    Init: Initialize p as an M-dimensional vector of all zeros;
    Result: Normalized distribution φ_p(X_i);
    for all columns in the pixel-cluster X_i do
        - find at most M biggest line segments overlapping with the given image
        column;
        - record the position k of the pixel-clusters among the pixel-clusters found
        counting from the bottom of the image up;
        if number of found line segments m equals M then
            -increase the k-th element of p by some positive constant ρ;
        else
            - increase all elements of p from position k to k + (M − m − 1) by
            some positive constant ρ;
        end
    end
    - normalize the vector p to unit L_1 norm; φ_p(X_i) = p;
end
```

that are positioned on a rotating table. When the table is turned, the objects change their position back and forth and left and right in a range of approximately 1m, thus creating different depth discontinuities. The sensor records the scene and stores the individual images in our database for subsequent experimentation. Some sample images from the acquired database are shown in Fig. 6. Here, the upper row depicts raw images of the constructed scene, while the lower row presents the corresponding detected patterns that serve as the input for the proposed labeling procedure.

For each acquired image, the ground truth needed to estimate the efficiency of the proposed labeling approach is obtained by hand labeling all the pixels belonging to all the detected line segments. The labels of the pixel-clusters obtained with our PGM technique are then compared to the pre-annotated ground truth to estimate the accuracy of the assessed procedure. Here, the accuracy is measured in the form of the correct labeling rate (CLR), which is defined as the ratio between the number of correctly labeled (non-zero) pixels $n_{correct}$ and all the non-zero pixels n_{all}, i.e.:

$$\text{CLR} = \frac{n_{correct}}{n_{all}}, \tag{7}$$

where CLR$\in [0, 1]$. Obviously, a CLR value close to one indicates a good performance, whereas a CLR value close to zero indicates a poor performance of the assessed technique.

planes and was used in this experimental evaluation as such. Since hardware modifications are beyond the scope of this paper, we made no efforts to alter the number of light planes emitted by our sensor.

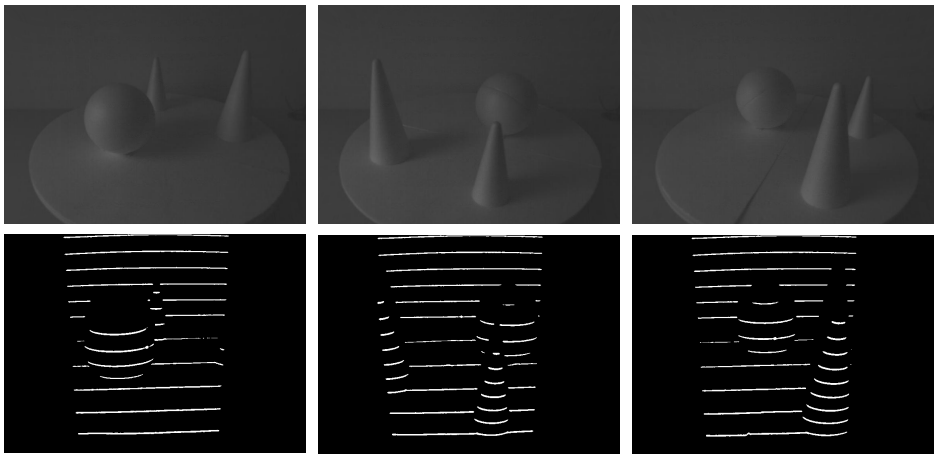
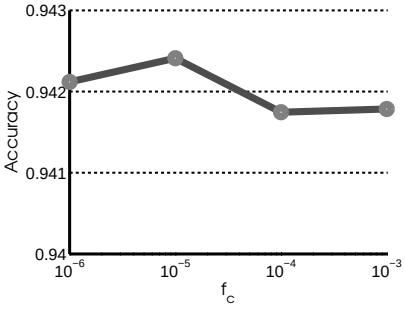


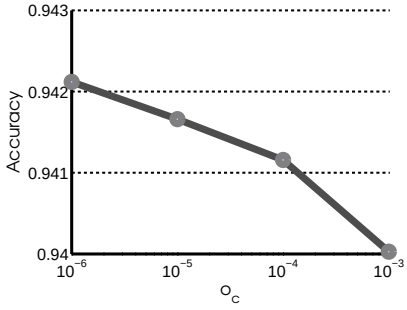
Fig. 6. Sample images from the constructed database: captured scene (upper row), detected pattern (lower row)

4.2. Experiments and results The first issue of interest with the proposed PGM labeling method is the impact of various parameters on the performance of the proposed method. To examine this issue, we fix all the open parameters to a default value, change one parameter at a time and observe how the labeling accuracy changes with respect to the varying parameter. Even though the parameters are, in general, not independent, we obtain a rough impression of the performance of the proposed method with respect to the changing parameter. For this series of experiments we only use a sequence length of one and, hence, rely for the moment only on spatial information. The results of the described experiments are presented in Fig. 7. Note that both the fraction cost f_c (see Eq. (3)) and the overlap cost o_c (see Eq. (5)) have only a little effect on the labeling performance, as long as they are kept sufficiently small. In contrast, the function drop rate h (Eq. (5)) has a significantly larger impact on the accuracy of the proposed method. All in all, our experiments suggest that a similar performance can be achieved over a wide range of parameter values. For the following experiments we fix the values of the fraction cost f_c, the overlap cost o_c and the function drop rate h to the values that resulted in the highest accuracy in this series of experiments.

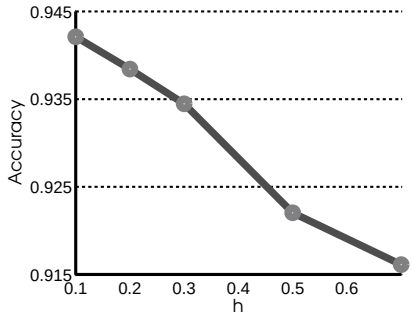
Another important observation that can be made on the basis of the presented results is the fact that the PGM labeling technique, in general, results in a relatively high labeling accuracy. This suggests that the spatial relationships defined between the line segments (and pixel-clusters) of the detected patterns are reasonable, despite the fact that in the vertical direction no obvious topology is present to define vertical neighbors. To summarize, the experiments show that it is possible to define the spatial relationships between neighboring line segments (and pixel clusters) with the proposed labeling technique without the need to project complex light patterns, as in (Ulusoy, 2010), and still produce a high labeling



(a) Effect of varying the fraction cost f_c



(b) Effect of varying the overlap cost o_c



(c) Effect of varying the function drop rate h

Fig. 7. Impact of: (a) fraction cost f_c , (b) overlap cost o_c , and (c) the function drop rate h on the labeling accuracy of the proposed approach. The accuracy is measured in terms of the CLR.

accuracy.

With the open parameters fixed in the previous series of experiments, we can now evaluate the effect of temporal information on the labeling accuracy of our procedure. To this end, we construct sequences of different lengths from images of our database that were taken one after the other. We first build a new database of all the possible sequences of length two, then a database of all the possible sequences of length three, and so forth up until a sequence length of five. For each constructed database we compute the labeling accuracy by considering spatial as well as temporal information. The results of the described experiments are presented in Fig. 8. Note that the performance increases quite significantly when adding temporal information to the labeling techniques. The largest jump in performance is visible when the sequence length is increased from one (no temporal information used) to two, while any additional images in the sequence again add a little to the overall accuracy of the PGM labeling approach, but not as much as the first. The results suggest that temporal information is indeed important when labeling structured-light patterns and represents an important source of

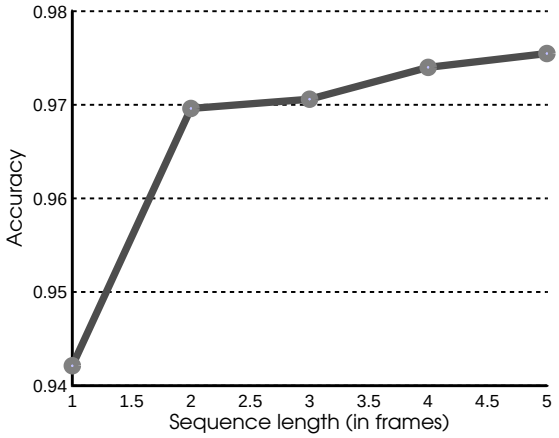


Fig. 8. Effect of varying the sequence length on the labeling accuracy. The accuracy is again measured in terms of the CLR.

information that can add to the overall accuracy of the labeling procedure.

Another important issue with the proposed labeling technique is the computational cost. In our experiments the average time needed to process sequences comprising a single image was around 0.3s. The computational cost increased linearly with the length of the sequence and prolonged the processing time by 1s (on average) for each image that was added to the sequence.

Up to this point, we have looked at the impact of open parameters and the effect of temporal information on the labeling accuracy of the proposed PGM technique. Hence, we observed only the relative improvements in the labeling accuracy, while it is also important to know how the proposed technique compares to other techniques that are applicable for the task of labeling uncoded structured-light patterns. To examine this issue we implement two reference techniques, namely

- *A naive labeling approach (NLA)*, which assigns light plane labels to the detected non-pixels in the order the non-zero pixels appear in the image. The first non-zero pixel at a given x-coordinate[6] looking from the bottom of the image up is assigned a label of one, the next detected non-zero pixel is assigned a label of two and so forth, until all eleven labels have been assigned;

- *The reference approach from (Ulusoy, 2010) (RPGM)*, which also relies on probabilistic graphical models, but makes no use of temporal information.

The comparison is presented in Fig. 9. Note that the proposed technique outperforms both reference techniques in terms of the labeling accuracy, again demon-

[6]The first non-zero pixel in this context refers to the pixel lowest in the image, or in other words, the pixel with the largest y-coordinate.

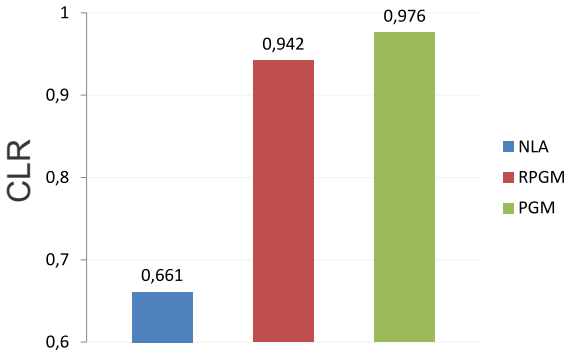


Fig. 9. Comparison with reference techniques. The performance is measured in the form of the CLR.

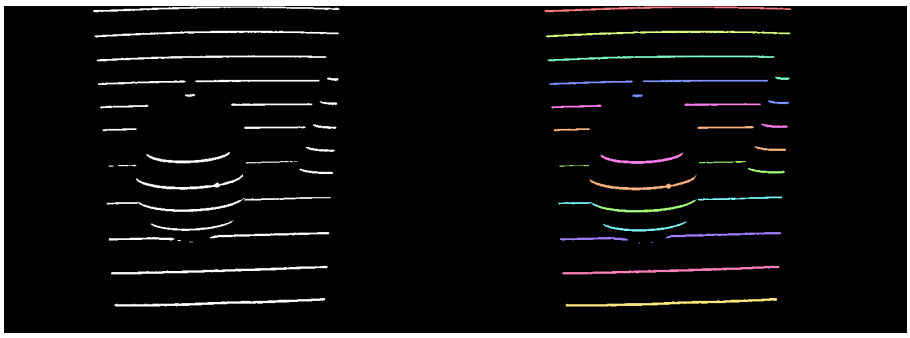


Fig. 10. Automatically labeled pixel-clusters. Left: Unlabeled pixel-clusters. Right: Labeled pixel-clusters (best viewed in color).

strating that spatial and temporal information are important for the labeling process.

All in all, the results of our experimental assessment suggest that considering spatio-temporal information for modeling dependencies among parts of the structured-light pattern represents a viable solution to the problem of light-plane labeling in our (depth-image) sensor. Our PGM approach yields good results on the experimental database and, as can be seen from Fig. 10, where a sample result image is presented, is capable of assigning the correct index to most of the pixel-clusters of the detected light pattern, even in difficult situations where large discontinuities are present in the scene. As can also be seen from the presented example, the proposed labeling approach easily handles spurious connections in the detected light planes and assigns them the correct labels. It is also relatively robust to discontinuities in the detected light planes that are caused either by discontinuities in the scene or failure to detect the projected patterns.

5. Conclusion We have presented a novel technique for labeling light planes in depth-image sensors using probabilistic graphical models. We have shown that next to the spatial relationships between the parts of the projected pattern, temporal information can also be exploited to improve the labeling accuracy. The performance of the proposed approach was compared to reference techniques from the literature and demonstrated highly competitive results.

REFERENCES

Albitar, C., Graebling, P., Doignon, C. (2009). Fast 3D vision with robust structured light coding. in *Proc. SPIE 7261, Medical Imaging 2009: Visualization, Image-Guided Procedures, and Modeling*, pp. 8.

Beumier, C., Acheroy, M. (1999). 3D facial surface acquisition by structured light. in *International Workshop on Synthetic Natural Hybrid Coding and 3D Imaging*, , pp. 103–106.

Boyer, K. L., Kak, A. C. (1987). Color-encoded structured light for rapid active ranging. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, 9, 14–28.

Brink, W., Robinson, A., Rodrigues, M. (2008).Indexing Uncoded Stripe Patterns in Structured Light Systems by Maximum Spanning Trees, in *Proc. British Machine Vision Conference (BMVC)*.

Curless, B., Levoy, M. (1995). Better optical triangulation through spacetime analysis, in *Proc. International Conference on Computer Vision (ICCV)*, pp. 987–994.

Daley, R.C., Hassebrook, L.G. (1998). Channel capacity model of binary encoded structured lightstripe illumination, *Applied Optics*, 37, 3689–3696.

Gonzales, R.C., Woods, R.E., *Digital Image Processing, 3rd Ed.*, Prentice Hall, New Yersey, (2008).

Hung, M.H., Pan, J.S., Hsieh, C.H.A Fast Algorithm of Temporal Median Filter for Background Subtraction. *Journal of Information Hiding and Multimedia Signal Processing*, 5(1), 33-40.

Koninckx, T.P., Geys, I., Jaeggli, T., van Gool, L. (2004) A graph cut based adapt. structured light approach for real-time range acquisition, in *Proc. 3D Data Proc., Visualization and Transmission*, pp. 413–421.

Kschischang, F. R., Frey, B. J., Loeliger, H.A. (2001).Factor Graphs and the Sum-Product Algorithm, *IEEE Transactions on Information Theory*, 47, 498–519.

Mertz, C., Koppal, S.J., Sia, S., Narasimhan, S. (2012).A low-power structured light sensor for outdoor scene reconstruction and dominant material identification, in *Proc. IEEE Computer Vision and Pattern Recognition Workshops (CVPRW)*, pp. 15–22.

Mooij, J.M. (2010).libDAI: A free and open source C++ library for Discrete Approximate Inference in graphical models, *Journal of Machine Learning Research*, 11, 2169–2173.

Robinson, A., Alboul, L., Rodrigues, M. (2004).Methods for Indexing Stripes in Uncoded Structured Light Scanning Systems, *Journal of WSCG*, 12, 371–378.

Salvi, J., Pages, J., Batlle, J. (2004).Pattern codification strategies in structured light systems, *Pattern Recognition*, 37, 827–849.

Salzmann, M., Pilet, J., Batlle, J. (2007).Surface deformation models for non-rigid 3-D shape recovery, *IEEE Transaction on Pattern Analysis and Machine Intelligence*, 29, 1481–1487.

Ulusoy, A.O., Calakli, F., Taubin, G. (2010).Robust one-shot 3D scanning using loopy belief propagation, in *Proc. Computer Vision and Pattern Recognition Workshops (CVPRW)*, pp. 15–22.

Vesnicer, B., Mihelič, F. (2008).The likelihood ratio decision criterion for nuisance attribute projection in GMM speaker verification, *EURASIP Journal of Advances in Signal Processing*, 2008, 1–11.

Volkov, A., ŽganecGros, J., Žganec, M., Javornik, T., Švigelj, A. (2013). Modulated acquisition of spatial distortion maps, *Sensors*, 13, 11069–11084.

Wiegerinck, W., Heskes T. (2003). Fractional Belief Propagation, in *Proc. Neural Information Processing Systems (NIPS)*, pp. 438–445.

Zhang, L., Curless, B., Seitzy, S.M. (2002).Rapid Shape Acquisition Using Color Structured Light and Multi-pass Dynamic Programming, in *Proc. 3D Data Proc., Visualization and Transmission*, pp. 24–36.

Žganec M., Žganec-Gros, J. (2009). Active 3D triangulation-based imaging method and device, *US Patent No. 7483151*.

Jaka Kravanja studied at the Faculty of Electrical Engineering of University of Ljubljana, Slovenia. In the last years of studies he focused on automation and intelligent systems. After graduating in 2011 and obtaining a B.Sc. degree, he enrolled in the Ph.D. program at the same faculty and conducted research in the field of computer vision and optoelectronics at Alpineon d.o.o in parallel. He is now completing his Ph.D. His other research interests also include artificial intelligence and machine learning.

Mario Žganec is CTO at the RTD-performing SME Alpineon. He received his PhD from the University of Ljubljana. His research interests include image processing, pattern recognition, biometrics and smart control systems. He has authored and co-authored several patents and scientific papers on these topics. He has coordinated several national research projects including the ATRIS project on robust 3D image acquisition.

Jerneja Žganec-Gros is COO at Alpineon. She received her PhD from the University of Ljubljana. Her research interests include speech and image processing, pattern recognition, biometrics and machine translation. She has co-authored several patents and scientific papers on these topics. Jerneja coordinated several international and national research projects including three Eureka projects. She is a member of the International Speech Communication Association and founding member of the Slovenian Pattern Recognition Society and of the Slovenian Language Technologies Society.

Simon Dobrišek is an associate professor of Electrical Engineering at the Faculty of Electrical Engineering, University of Ljubljana, where he teaches courses on Artificial Intelligent Systems, Pattern Recognition, Information Theory and Coding, Intelligent Systems in Automation and Seminar on Biometric Systems. His research interests center on artificial intelligence, pattern recognition, biometrics, smart surveillance systems, and spoken language technologies. He has published popular and peer-reviewed scientific journal articles and conference papers on these topics. Simon Dobriek is a member of IEEE, International Association of Pattern Recognition and International Speech Communication Association.

Vitomir Štruc received his PhD degree in electrical engineering from the University of Ljubljana in 2010. After his PhD studies he took a senior software developer position at a privately held company, where he participated in several projects related to face recognition, biometrics, and computer vision in general. In 2011 Vitomir began working as a Research Fellow on a postdoctoral project related to face recognition in ambient intelligence environments at the Laboratory of Artificial Perception, Systems and Cybernetics (LUKS) of the Faculty of Electrical Engineering, University of Ljubljana. He is a research fellow and teaching assistant since 2013. Vitomir has been involved as a key member in several national and EU

funded R&D projects and authored or co-authored more than 50 research papers for leading international peer reviewed journal and prominent conferences related to different issues of computer vision, image processing, pattern recognition, face recognition and biometrics.