Fifth Croatian Computer Vision Workshop

October 11, 2016, Osijek, Croatia

PROCEEDINGS OF CCVW 2016

University of Zagreb
Center of Excellence
for Computer Vision

Proceedings of the Croatian Computer Vision Workshop CCVW 2016 ISSN 1849-1227
Publisher: University of Zagreb Faculty of Electrical Engineering and Computing, Unska 3, HR-10000 Zagreb, Croatia
http://www.fer.unizg.hr/crv/proceedings

CCVW 2016

Proceedings of the Croatian Computer Vision Workshop

Osijek, Croatia, October 11, 2016

S. Lončarić, R. Cupec (Eds.)

Organizing Institution

Center of Excellence for Computer Vision, University of Zagreb, Croatia
Faculty of Electrical Engineering, Computer Science and Information Technology Osijek, Croatia

Auspiceis

Croatian Academy of Engineering

Technical Co-Sponsors

IEEE Croatia Section
IEEE Croatia Section Computational Intelligence Chapter
IEEE Croatia Section Computer Society Chapter
IEEE Croatia Section Signal Processing Society Chapter IEEE Croatia Section Systems, Man and Cybernetics Society Chapter

Sponsors

The Foundation of the Croatian Academy of Sciences and Arts
Visage Technologies AB

Proceedings of the Croatian Computer Vision Workshop
CCVW 2016

Editor-in-chief

Sven Lončarić (sven.loncaric@fer.hr)
University of Zagreb Faculty of Electrical Engineering and Computing
Unska 3, HR-10000, Croatia

Editor

Robert Cupec (rcupec@etfos.hr)
Josip Juraj Strossmayer University of Osijek
Faculty of Electrical Engineering, Computer Science and Information Technology Osijek
Kneza Trpimira 2b, HR-31000 Osijek, Croatia

Production, Publishing and Cover Design

Tomislav Petković (tomislav.petkovic.jr@fer.hr)
University of Zagreb Faculty of Electrical Engineering and Computing
Unska 3, HR-10000, Croatia

Publisher

University of Zagreb Faculty of Electrical Engineering and Computing
Unska 3, HR-10000 Zagreb, OIB: 57029260362

ISSN 1849-1227

Preface

On behalf of the Organizing Committee it is our pleasure to invite you to Osijek for the 5^{th} Croatian Computer Vision Workshop. This year, the Workshop will be organized in the beautiful town of Osijek at the Faculty of Electrical Engineering, Computer Science and Information Technology, Josip Juraj Strossmayer University of Osijek, Croatia.

The objective of the Workshop is to bring together professionals from academia and industry in the area of computer vision theory and applications in order to foster research and encourage academia-industry collaboration in this dynamic field. The Workshop program includes oral and poster presentations of original peer reviewed research from Croatia and elsewhere. Furthermore, the program includes invited lectures by distinguished international researchers presenting state-of-the-art in computer vision research. Workshop sponsors will provide a perspective on needs and activities of the industry. Finally, one session shall be devoted to short presentations of activities at Croatian research laboratories.

The Workshop is jointly organized by the Center of Excellence for Computer Vision, University of Zagreb and by the Faculty of Electrical Engineering, Computer Science and Information Technology, Josip Juraj Strossmayer University of Osijek. This year, for the first time, the Workshop has been organized under the auspices of the Croatian Academy of Engineering.

Osijek is a beautiful European city with many cultural and historical attractions, which we are sure all participants will enjoy. We look forward to meet you all in Osijek for the 5^{th} Croatian Computer Vision Workshop.

October 2016

Sven Lončarić, General Chair

Robert Cupec, Technical Program Chair

Acknowledgements

The 2016 5th Croatian Computer Vision Workshop (CCVW) has been organized under the auspices of the Croatian Academy of Engineering.

The Workshop is the result of the committed efforts of many volunteers. All included papers are results of dedicated research. Without such contribution and commitment this Workshop would not have been possible.

Program Committee members and reviewers have diligently reviewed submitted papers and provided extensive reviews which will be an invaluable help in future work of collaborating authors. Managing the electronic submissions of the papers, the preparation of the abstract booklet and of the online proceedings also required substantial effort and dedication that must be acknowledged. The Local Organizing Committee members did an excellent job to guarantee a successful outcome of the Workshop.

We are grateful to the Technical Co-Sponsors, IEEE Croatia Section and its four Chapters (CIS11, C16, SP01, SMC28), who helped us in granting the high scientific quality of the presentations.

We are also grateful to the sponsors The Foundation of the Croatian Academy of Sciences and Arts and Visage Technologies that financially supported this Workshop.

Contents

Organizing Committee

General Chair

Sven Lončarić, University of Zagreb, Croatia

Technical Program Chair

Robert Cupec, Josip Juraj Strossmayer University of Osijek, Croatia

Publications Chair

Tomislav Petković, University of Zagreb, Croatia

Technical Program Committee

Bart Bijnens, Spain
Hrvoje Bogunović, Austria
Mirjana Bonković, Croatia
Karla Brkić, Croatia
Hrvoje Gold, Croatia
Mislav Grgić, Croatia
Sonja Grgić, Croatia
Andras Hajdu, Hungary
Edouard Ivanjko, Croatia
Bojan Jerbić, Croatia
Zoran Kalafatić, Croatia
Stanislav Kovačič, Slovenia
Josip Krapac, Croatia
Lidija Mandić, Croatia
Vladan Papić, Croatia
Renata Pernar, Croatia
Tomislav Petković, Croatia
Ivan Petrović, Croatia
Tomislav Pribanić, Croatia
Slobodan Ribarić, Croatia
Damir Seršić, Croatia
Darko Stipanišev, Croatia
Marko Subašić, Croatia
Federico Sukno, Spain
Siniša Šegvić, Croatia
Vladimir Zlokolica, Serbia

Local Organizing Committee

Irena Galić, Chair
Željko Hocenski
Snježana Rimac-Drlje
Goran Martinović
Drago Žagar
Časlav Livada
Hrvoje Leventić
Krešimir Romić
Ivan Vidović
Tomislav Matić
Ivan Aleksi
Tomislav Keser
Emmanuel Karlo Nyarko
Damir Filko
Ratko Grbić
Mario Vranješ

Reviewers

Damir Filko, Croatia
Irena Galić, Croatia
Tomislav Matić, Croatia
Igor Sunday Pandžic, Croatia
Tomislav Petković, Croatia
Tomislav Pribanić, Croatia
Krešimir Romić, Croatia
Federico Sukno, Spain
Marko Subašić, Croatia
Mario Vranješ, Croatia

Probabilistic Eye Contact Detection for the Robot-assisted ASD Diagnostic Protocol

Frano Petric, Damjan Miklić and Zdenko Kovačić
University of Zagreb, Faculty of Electrical Engineering and Computing
Unska 3, 10000 Zagreb, Croatia

Abstract— **This paper describes a probabilistic method for eye contact detection, aimed at facilitating the diagnostic process of Autism Spectrum Disorder (ASD). Eye contact, which is a special case of a wider social-attention cue called gaze, plays a major role in ASD diagnostics protocols used in clinical practice. Therefore, a reliable method for automatic eye contact detection is a key capability to open the way towards robot-assisted autism diagnostics, which has the potential to reduce diagnostic time and increase reliability. The proposed method uses data from a simple low-resolution monocular camera, which is built into the NAO humanoid robot, and uses head pose and gaze direction as its inputs, which are very prone to outliers. We use a probabilistic framework in order to provide continuous measure of the eye contact and increase robustness to outliers. Furthermore, we take into account the temporal aspect of eye contact, in order to discard short glances which do not indicate actual transfer of attention. Initial experimental results, conducted in a laboratory setting, confirm that the proposed method can be effective in detecting eye contact with the NAO humanoid robot.**

I. INTRODUCTION

Autism Spectrum Disorder (ASD) is a neurodevelopmental disorder characterised by impairments in social interaction, verbal and non-verbal communication, accompanied by repetitive behaviours and interests. With increasing prevalence rates, it is expected to become one of the most commonly diagnosed disorders. To increase the inclusion rate and reduce the cost of lifelong care for people with ASD, experts focus on early diagnostics and intervention, which is hindered by the lack of medical markers that could be used in a diagnostic process. The diagnosis relies on behavioural observations made by experienced clinicians, obtained through several approaches: a) using criteria from the Diagnostic and Statistical Manual of Mental Disorders (DSM) [1]; b) testing children using the Autism Diagnostic Observation Schedule (ADOS) [2]; and, c) interviews with the caregivers using the Autism Diagnostic Interview-Revised (ADI-R) [3].

The diagnostic procedure is highly complex due to simultaneous observation, coding and interpretation of many behaviours as well as administration of various specific tasks, lowering the reliability of the diagnostics and prolonging the time needed to obtain the diagnosis [4]. While ASD can be somewhat reliably identified in children by the age of two, authors in [5] state that there is a need for more continuous measures of aspects of ASD. Modern robotics technologies, similar to those that are already in use in the intervention process [6][7], seem capable of providing adequate tools to address the need for a more objective approach that would help clinicians in gathering multi-modal information and coding the social behaviour, as well as provide the consistent stimuli for interaction.

One of the most important aspects of ASD, heavily relied upon in diagnostics, is the eye contact (or absence of it), as can be inferred from the ADOS protocol [2], in which the eye contact is constantly tracked and quantified. Eye contact is a special case of a wider social attention cue called gaze, where the gaze is directed towards another person's eyes, and can be detected by children as early as 4 months after birth [8].

Detection of eye contact plays a major role in the robot-assisted ASD diagnostic protocol [9], [10] which is based on ADOS and aimed at expediting the diagnostics procedure while retaining (and possibly increasing) the reliability of the standard ADOS protocol. Eye contact is of special significance in two tasks, *response to name call* and *joint attention*, where it is used as a direct measure of a child's attention towards the robot. However, strict technical definition of the eye contact is lacking, especially in terms of duration and reciprocity. Authors in [11] expect that natural eye contact during conversation should last from 3 to 10 seconds. However, clinical practice and studies such as [12] suggest that children with ASD have significantly shorter attention span than typically developed children. In terms of reciprocity of eye contact, some authors [13] consider the eye contact to be established when both parties acknowledge the gaze direction of the other party, mainly through facial expressions. Considering the lack of controllable facial features on the NAO robots used in robot-assisted diagnostic protocol, we focus mainly on the recognition of eye contact in terms of gaze aimed towards NAO's eyes.

As reported in [14], looking at the eyes of others was significantly decreased in children (aged 2) with ASD, while looking at the mouth was increased compared to the control group of typically developed children. While other authors [15] suggest that the aforementioned difference was not statistically significant, this debate indicates that the technical system for eye contact detection needs to be precise and able to distinguish between child looking at the eyes of the robot and child looking at the other parts of the robot. Head mounted eye trackers can satisfy this requirement, but mounting them can be difficult with the children with ASD, especially due to the indispensable calibration procedure.

Additionally, they require the object that the child is looking at to be stationary and are difficult to use on-line, rendering them unusable in the robot-assisted diagnostic protocol. To keep the protocol non-invasive, we are using the robot's camera for on-line detection of eye contact. However, methods for head pose and eye gaze estimation from a monocular camera in general do not possess the required precision to differentiate the child looking at different parts of the robot. Additionally, they usually produce a binary result for eye contact detection, which could be prone to false positives and negatives, which is detrimental for the reliable diagnosis of ASD.

To determine the quality of the observed eye contact and ensure continuous measure of attention throughout the diagnostic task, as a main contribution of the paper, we propose probabilistic eye contact detection, based on the sampling of Gaussian representations of head pose and gaze estimates.

The paper is organized as follows. Section 2 provides the overview of head pose and gaze estimation methods and their use for eye contact detection. In Section 3, we describe the probabilistic framework for eye contact detection, which takes into account both the probability and the duration of eye contact. Section 4 discusses results obtained in a laboratory setting. Conclusion and guidelines for future work are given in Section 5.

II. HEAD POSE AND GAZE ESTIMATION FOR EYE CONTACT DETECTION

The most straightforward way to approach eye contact detection is to use machine learning methods on a pre-labeled set of images, avoiding the problem of 3D head pose and gaze estimation. Considering the eye contact detection within the robot-assisted ASD diagnostic protocol, the main drawback of such approach is not being able to take into account that the region of the eyes on the robot is not necessarily in the origin of the camera frame, reducing the accuracy of the training dataset. Additionally, if the approach is to be expanded towards additional regions of interest it would require the whole process of labeling and learning to be repeated.

According to [16], gaze combined with head pose and pointing gestures make significant contributions to the determination of another's focus of attention and subsequently the eye contact. Head pose and gaze estimation are active and prolific areas of research within the scientific community [17][18], and the results from those areas are often used to detect the focus of attention of individuals [19][20]. Herein, we use the output of those algorithms as inputs for the algorithm of eye contact detection we propose. To provide these inputs, we use commercially available face tracking and analysis software *visage|SDK* [21]. Through a well documented API, *visage|SDK* enables head pose, gaze, and facial-features tracking. It also provides 3D positions of all facial features, including the eyes which are the origins of the gaze. Gaze is estimated in terms of azimuth θ and elevation φ with respect to the camera frame as a single measurement

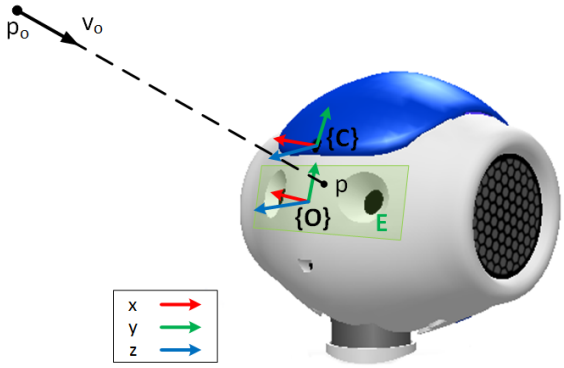


Fig. 1. Detecting eye contact with NAO robot using gaze origin p_o and gaze direction v_o. Point p is projection of p_o along the vector v_o on the plane π_o spanned by x and y vectors of coordinate frame $\{O\}$. If point p is inside eye region $\mathcal{E}$, eye contact is detected.

for both eyes. Therefore, we use the midpoint of the line connecting two eyes as the origin of gaze and denote it $\mathbf{p}_c$. For the upcoming calculations it is neccessary to transform the orientation of gaze into the unit vector of gaze direction $v_c = [r_x, r_y, r_z]$. Taking into account that the camera frame $\{C\}$ for the *visage|SDK* is left-handed, following relations hold:

$$\begin{aligned} r_x &= -\sin(\theta) \\ r_y &= -\cos(\theta)\sin(\varphi) \\ r_z &= -\cos(\theta)\cos(\varphi) \end{aligned} \tag{1}$$

The origin of a subject's gaze $\mathbf{p}_c$ and unit gaze vector $\mathbf{v}_c$ are obtained from the tracking software in coordinate frame of the camera $\{C\}$. Let robot's eye region $\mathcal{E}$ lie in the plane $\pi_o \leftarrow Ax + By + Cz + D = 0$. Plane π_o is defined in the coordinate frame of the eyes $\{O\}$, with known transformation between coordinate frames of the eyes and camera $\mathbf{T}_o^c$. Using $\mathbf{T}_o^c$, one can easily express $\mathbf{p}_c$ and $\mathbf{v}_c$ in the coordinates of $\{O\}$, obtaining origin of gaze $\mathbf{p}_o = [p_x, p_y, p_z]$ and gaze vector $\mathbf{v}_o = [v_x, v_y, v_z]$. Now, $\mathbf{p}_o$ and $\mathbf{v}_o$ define a line in $\{O\}$ (see Fig. 1), and to calculate the point $\mathbf{p}$ where this line intersects the plane π_o, we need to find $t \in \mathbb{R}$ such that the following holds:

$$\mathbf{p} = \mathbf{p}_o + t \cdot \mathbf{v}_o \in \pi_o \tag{2}$$

which can be rewritten as:

$$A(p_x + tv_x) + B(p_y + tv_y) + C(p_z + tv_z) + D = 0 \tag{3}$$

yielding the following solution for t:

$$t = -\frac{Ap_x + Bp_y + Cp_z + D}{Av_x + Bv_y + Cv_z} \tag{4}$$

Equation (4) has exactly one solution if the line defined by the gaze vector is not parallel with the plane π_o, which is the case in the intended application. Point $\mathbf{p}$ on the plane is calculated by plugging t into (2), and eye contact is detected if that point is inside the region which corresponds to the eyes of the robot $\mathcal{E}$ (see Fig 1).

III. PROBABILISTIC EYE-CONTACT DETECTION

To provide a continuous measure, we propose the conditional probability of eye contact, given the random vectors for head pose and gaze:

$$p(X|H,G) \tag{5}$$

X is a random variable describing the eye contact, while H and G are normally distributed random vectors attached to head pose and gaze direction:

$$H \sim \mathcal{N}(\mathbf{p} \in \mathbb{R}^3, \mathbf{\Sigma}_p \in \mathbb{R}^{3x3}) \tag{6}$$
$$G \sim \mathcal{N}(\mathbf{o} \in \mathbb{R}^2, \mathbf{\Sigma}_o \in \mathbb{R}^{2x2}) \tag{7}$$

with mean values $\mathbf{p}$ and $\mathbf{o}$ provided by the face tracking system. Variance matrices $\mathbf{\Sigma}$ of these random variables are a measure of the quality of the face tracking systems, i.e. they represent the confidence in the results of the estimation. Since *visage|SDK* employs extended Kalman filter to track facial features, the underlying variances matrices of the Kalman filter could be used directly, but are not available in the current version of the software. Therefore, we estimate the variance matrices during the calibration proces (see Section IV).

Knowing H and G, our main goal is to obtain the probability density function of gaze direction on the plane π_o, similar to obtaining the point on the plane from Section II. Obtaining an analytical solution to this problem is the subject of our ongoing research. In this paper we provide a numerical solution based on sampling the distributions of H and G.

First, we draw N samples from H and M samples form G, generating $N \cdot M$ pairs of $(point, orientation)$. For each of these pairs we calculate the gaze projection point on the plane π_o, as described in Section II. This procedure produces $N \cdot M$ samples in the plane π_o (Fig. 2), which can be approximated by an arbitrary distribution through statistical analysis.

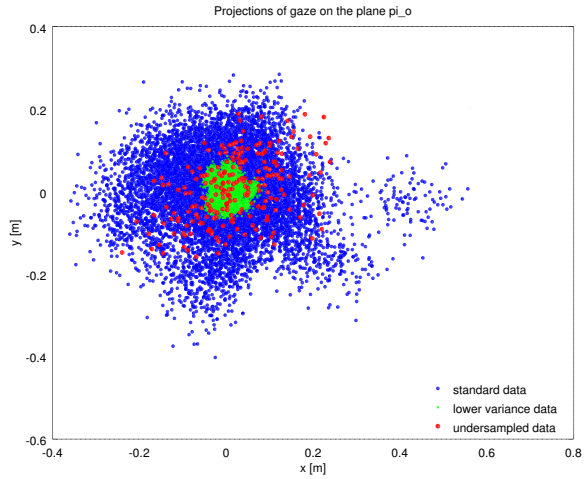


Fig. 2. Projection of the near vertical gaze on the plane π_o ($0°$, distance $1\ m$)

Since all of the samples are lying on a plane, we opt for bivariate normal distribution with the following probability density function:

$$f(\mathbf{x}, \boldsymbol{\mu}, \mathbf{\Sigma}) = \frac{1}{\sqrt{|\mathbf{\Sigma}|(2\pi)^2}} e^{-\frac{(\mathbf{x}-\boldsymbol{\mu})' \mathbf{\Sigma}^{-1} (\mathbf{x}-\boldsymbol{\mu})}{2}} \tag{8}$$

where $\boldsymbol{\mu} \in \mathbb{R}^2$ is mean vector and $\mathbf{\Sigma} \in \mathbb{R}^{2x2}$ is covariance matrix.

As can be seen in Fig. 3, the set of points does not necessarily resemble Gaussian distribution for large angles of gaze, mainly due to the projective transformation which introduces nonlinearity in the mapping of distributions. However, as the field of view of the robot is relatively narrow and the head of the robot can be controlled to always face the subject during the interaction, gaze can never attain angles large enough for this effect to occur, which justifies use of normal distribution from (8).

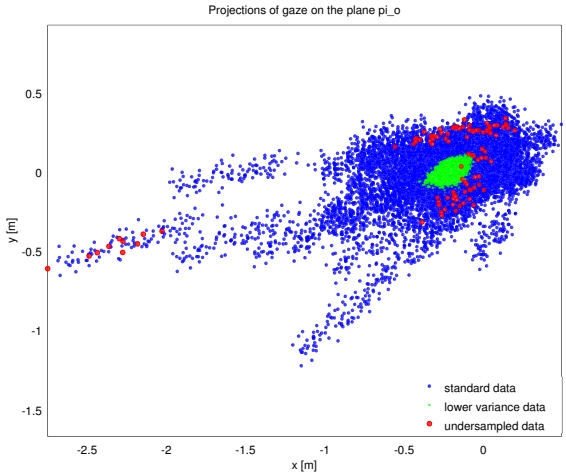


Fig. 3. Projection of the angled gaze on the plane π_o ($45°$, distance $1.5\ m$)

To obtain the probability of eye contact given the probability density function of gaze on the plane π_o, we need to integrate (8) over the region of the robot's eyes $\mathcal{E}$:

$$p(X = eye\ contact) = \iint_{\mathcal{E}} f(\mathbf{x}, \boldsymbol{\mu}, \mathbf{\Sigma})\, d\mathbf{x} \tag{9}$$

Double integral in (9) cannot be analytically solved for an arbitrary area $\mathcal{E}$, therefore it needs to be evaluated numerically.

Along with the probabilty of the eye contact, it is also important to take into account the duration of the eye contact. As stated in the introductory section, studies usually compare durations between children with ASD and typically developed children, but do not state how long the eye contact needs to be before it is classified as eye contact, leaving the definition of eye contact incomplete, at least in technical terms. The main goal of introduction of the minimal duration is to eliminate the short glances which do not represent the complete transfer of attention towards robot's eyes. According to observations in clinical practice which state

that the eye contact duration for children with ASD is short in comparison with typically developed children, we set this duration to $0.5\ s$. To incorporate this temporal dimension of the eye contact into the proposed probabilistic framework, we introduce continuous first order filter of the following form:

$$G(s) = \frac{1}{T \cdot s + 1} \tag{10}$$

where T is the time constant of the filter. We set T such that the filter from (10) reaches 95% of the final value within $0.5\ s$, which equates to $T = 0.15\ s$. Next, we discretize the filter using bilinear transform, taking into account sample time T_s which is determined by the framerate of the camera and may vary depending on the resolution of the video stream:

$$G(z) = \frac{T_s \cdot z + T_s}{(2T + T_s) \cdot z + (T_s - 2T)} \tag{11}$$

From (11) stems the update equation of the first order discrete filter:

$$y_k = \frac{1}{2T + T_s}\left[(2T - T_s)y_{k-1} + T_s\left(x_k + x_{k-1}\right)\right] \tag{12}$$

where subscript k denotes values in the current time step, while $k-1$ denotes the values from the previous time step.

IV. RESULTS

The experiments described in this section are performed by recording the video using the camera of NAO robot with resolution 640×480 pixels. Video is then processed on the computer running *visage|SDK*. While *visage|SDK* has proven to be accurate in terms of tracking, during the experiments we observed an error in the initialization step of the tracker which resulted in an offset in the estimate of the elevation angle φ. This offset is introduced when fitting 3D model to facial features and is observed to be constant for one recording, but may vary between different recordings. Therefore, we introduce the calibration step in each of the experiments. Calibration consists of looking directly at the camera of the robot for the predefined amount of time in the start of the experiment, which enables the tracker to calculate the offset knowing that the subject is looking directly at the camera. During this calibration step we also calculate and record the variances in the head pose and gaze data. Averaging these variances over multiple calibration steps yields the following matrices which are used in the experiments:

$$\begin{aligned} \mathbf{\Sigma}_p &= \begin{bmatrix} 0.010 & 0 & 0 \\ 0 & 0.0005 & 0 \\ 0 & 0 & 0.006 \end{bmatrix} \\ \mathbf{\Sigma}_o &= \begin{bmatrix} 0.017533 & 0 \\ 0 & 0.015020 \end{bmatrix} \end{aligned} \tag{13}$$

In the experiments we use $N = 100$ and $M = 100$, obtaining 10000 samples from which the projected distribution is computed. Finally, the integral from (9) is numerically computed over eye region (green rectangle in Fig. 4) using Gauss-Legendre quadrature. Obtained probability is then filtered through (11), with sample time $T_s = 0.077\ s$, corresponding to 13 FPS which is the maximum for the NAO camera at the resolution of 640×480 pixels. With the aforementioned parameters, the average runtime of the algorithm for the probability calculation is under 2 milliseconds (not taking into account the time needed for *visage|SDK* to estimate head pose and gaze).

A. Transfer of attention

This experiment is designed to evaluate the ability of the proposed probabilistic framework to detect the transfer of attention of the subject towards the eyes of the robot. We instruct the subject to focus on seven different areas (Fig 4) of the robot, starting with the camera region $\mathcal{C}$ to obtain the calibration parameters. Next the subject needs to focus on the eyes region $\mathcal{E}$, denoted with green rectangle in Fig. 4, size of which is $10 \times 4\ cm$.

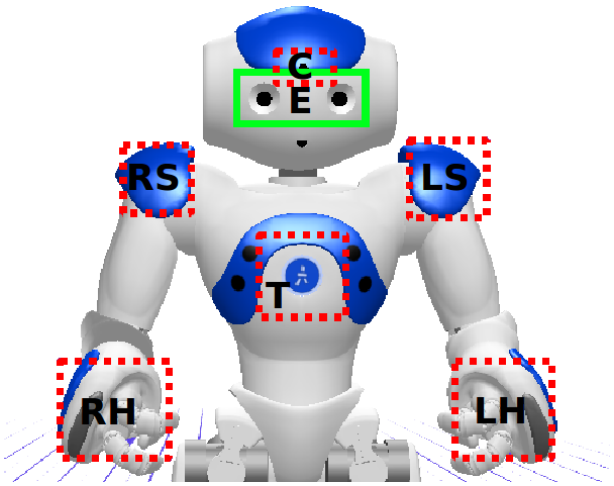


Fig. 4. Regions of interest during the attention transfer experiment.

After focusing on the eyes of the robot, the subject needs to transfer attention to the torso $\mathcal{T}$, left hand $\mathcal{LH}$, right hand $\mathcal{RH}$, right shoulder $\mathcal{RS}$ and left shoulder $\mathcal{LS}$ in a sequence. After the left shoulder, the subject is instructed to look towards robot's eyes once again. The results are shown in Fig. 5.

The plot of the probability of eye contact in Fig. 5 suggests that the proposed framework has the capability to detect eye contact with high probability and is not prone to false positives when looking at the other parts of the robot, with the exception of camera region which is expected due to the partial overlap of the regions.

Next, we lower confidence in the head pose and gaze estimation by doubling the values of variance matrices from (13). The results are shown in Fig. 6.

By comparing the probabilities from Fig. 5 and Fig. 6, one can observe that lower confidence in head pose and gaze estimates results in decrease of the probability of eye contact, meaning that the robot has less confidence that the eye contact really occurred.

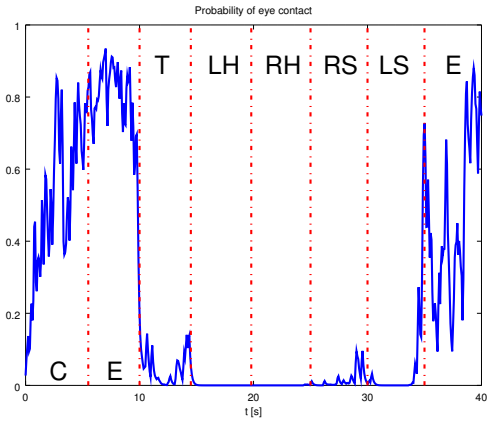


Fig. 5. Probability of eye contact during the experiment. Subject is 56 cm from the robot and looking at camera $\mathcal{C} \rightarrow$ eyes $\mathcal{E} \rightarrow$ torso $\mathcal{T} \rightarrow$ left hand $\mathcal{LH} \rightarrow$ right hand $\mathcal{RH} \rightarrow$ right shoulder $\mathcal{RS} \rightarrow$ left shoulder $\mathcal{LS} \rightarrow$ eyes $\mathcal{E}$.

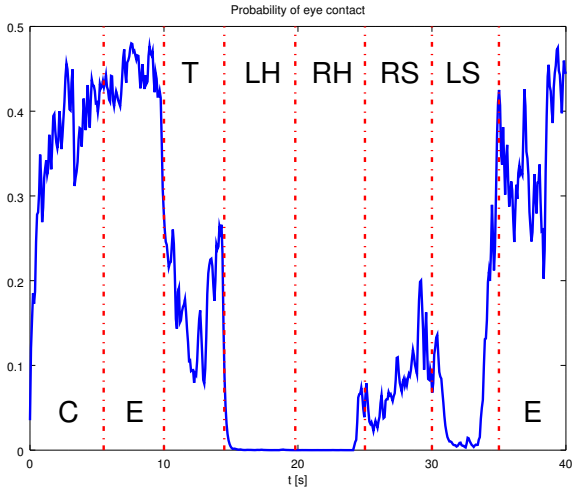


Fig. 6. Probabilty of eye contact with lower confidence in head pose and gaze estimation. Subject is 56 cm from the robot and looking at camera $\mathcal{C} \rightarrow$ eyes $\mathcal{E} \rightarrow$ torso $\mathcal{T} \rightarrow$ left hand $\mathcal{LH} \rightarrow$ right hand $\mathcal{RH} \rightarrow$ right shoulder $\mathcal{RS} \rightarrow$ left shoulder $\mathcal{LS} \rightarrow$ eyes $\mathcal{E}$.

B. *Evaluating glances*

This experiment is designed to assess the proposed frame work with respect to short glances towards the robots eyes, which should not be classified as eye contact. After the calibration step, the subject is instructed to avert gaze from the robot and to produce several short glances towards the robot eyes without focusing on them. The computed probability for this experiment is shown in Fig. 7.

Fig. 7 shows the effect of the discrete filter on the computed probability. Without the filter, first short glance would be classified as eye contact with 90% confidence, while with filter the probability reaches only 35%. The effect is that the robot has less confidence that the eye contact has occurred for short glances, diminishing the possibility of falsely classifying glances as eye contact.

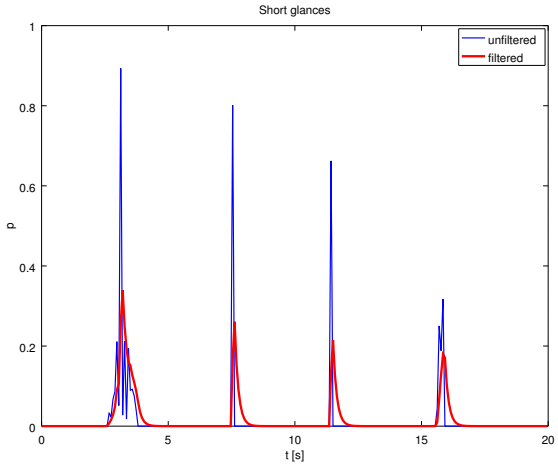


Fig. 7. Probability of eye contact for short glances towards robot's eyes.

C. *Distance from the robot*

This experiment is designed to assess the calculated eye contact probability with respect to the distance of the subject from the robot. The subject was instructed to maintain eye contact while robot was approaching the subject and then moving away, and the calculated probability is shown in Fig. 8.

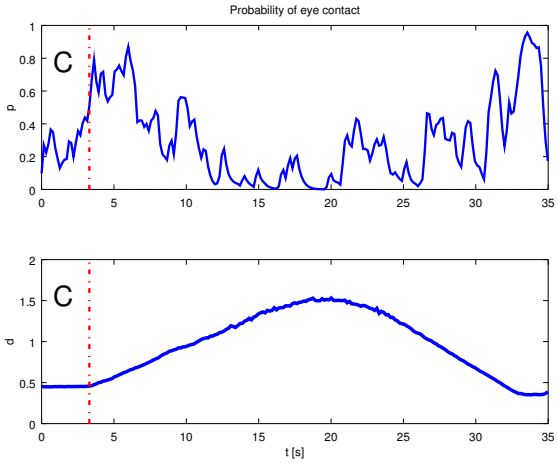


Fig. 8. Probability of eye contact with respect to distance. For first 3.5 s subject is looking at the camera $\mathcal{C}$ to calibrate the tracking system.

After the initial calibration period, the trend of probability of eye contact in Fig. 8 is inversely proportional to the distance from the robot. With larger distance, small variations in head pose and gaze result in larger offsets on the plane π_o, spreading the distribution, thus decreasing the probability over the fixed region. Similar effect is observed when varying the angle of the gaze.

D. *Undersampling the distributions of head pose and gaze*

To illustrate the effect of M and N being too small to correctly represent the projected distribution, we analyze the

recording of experiment from section IV-A with different values for M and N. We run the analysis 3 times, showing the calculated probability in Fig. 9.

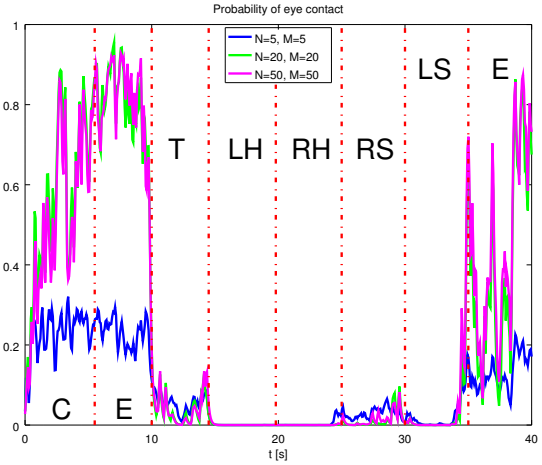


Fig. 9. Probability of eye contact for 3 runs of the algorithm with different number of samples drawn from distributions of head pose and gaze. Subject is 56 cm from the robot and looking at camera $\mathcal{C} \rightarrow$ eyes $\mathcal{E} \rightarrow$ torso $\mathcal{T} \rightarrow$ left hand $\mathcal{LH} \rightarrow$ right hand $\mathcal{RH} \rightarrow$ right shoulder $\mathcal{RS} \rightarrow$ left shoulder $\mathcal{LS} \rightarrow$ eyes $\mathcal{E}$.

Comparing Fig. 5 and Fig. 9, it can be observed that small number of samples cannot represent the projected distribution adequately, resulting in the inconsistent calculation of the probability of eye contact. Although there are no significant differences between plots for number of samples larger than 20 for the given experiment, minimal number of samples increases with increased variance in estimates of head pose and gaze.

V. CONCLUSION

To reliably detect eye contact, a key feature in envisioned robot-assisted ASD diagnostic protocol, we propose a general probabilistic framework based on sampling of normal distributions of head pose and gaze estimates. The proposed framework provides a continuous measure of eye contact in form of probability, with increased robustness to outliers. It also takes into account temporal component of the eye contact, diminishing the possibility of false detections of eye contact by discarding short glances. Through several experiments in a laboratory setting, we show that the proposed framework can successfully cope with variances in the estimates of head pose and gaze.

Analytical solution to the distribution projection problem is of highest importance in our future work, as is the assessment of the proposed probabilistic framework in clinical sessions with children. We plan to incorporate the proposed method into the robot-assisted diagnostic protocol to deduce whether the eye scanning patterns during the interaction with NAO can be used to more acurately differentiate children with ASD and typically developed children.

ACKNOWLEDGMENT

This work has been fully supported by Croatian Science Foundation under the project Autism Diagnostic Observation with Robot Evaluator (no. 93743-2014).

REFERENCES

[1] *Diagnostic and Statistical Manual of Mental Disorders*. American Psychiatric Association, 4th edition, 2002.

[2] C. Lord, M. Rutter, P.C. Dilavore, and S. Risi. *Autism Diagnostic Observation Schedule*. Western Psychological Services, 2002.

[3] M. Rutter, A. LeCouteur, and C. Lord. *The Autism Diagnostic interview, Revised (ADI-R)*. Western Psychological Services, 2003.

[4] A. Klin, J. Lang, V. Chicchetti, and F.R. Volkmar. Interrater reliability of clinical diagnosis and DSM-IV criteria for autistic disorder: results of the DSM-IV autism field trial. *Journal of Autism and Developmental Disorders*, 30(2):163–167, 2000.

[5] F. R. Volkmar, C. Lord, A. Bailey, R. T. Schultz, and A. Klin. Autism and pervasive developmental disorders. *Journal of Child Psychology and Psychiatry*, 45:135170, 2004.

[6] I. Iacono, H. Lehmann, P. Marti, B. Robins, and K. Dautenhahn. Robots as social mediators for children with Autism - A preliminary analysis comparing two different robotic platforms. In *Development and Learning (ICDL), 2011 IEEE International Conference on*, volume 2, pages 1–6, 2011.

[7] Changchun Liu, K. Conn, N. Sarkar, and W. Stone. Online affect detection and robot behavior adaptation for intervention of children with autism. *Robotics, IEEE Transactions on*, 24(4):883–896, 2008.

[8] D. Maurer. *Social Perception in Infants*, chapter Infants perception of facedness. 1985.

[9] F. Petric, K. Hrvatinic, A. Babic, L. Malovan, D. Miklic, Z. Kovacic, M. Cepanec, J. Stosic, and S. Simlesa. Four tasks of a robot-assisted autism spectrum disorder diagnostic protocol: First clinical tests. In *Global Humanitarian Technology Conference (GHTC), 2014 IEEE*, pages 510–517, Oct 2014.

[10] F. Petric, D. Tolić, D. Miklić, Z. Kovačić, M. Cepanec, and S. Šimleša. *Intelligent Robotics and Applications: 8th International Conference, ICIRA 2015, Portsmouth, UK, August 24-27, 2015, Proceedings, Part II*, chapter Towards A Robot-Assisted Autism Diagnostic Protocol: Modelling and Assessment with POMDP, pages 82–94. Springer International Publishing, Cham, 2015.

[11] Janet Dean Michael Argyle. Eye-contact, distance and affiliation. *Sociometry*, 28(3):289–304, 1965.

[12] G. Dawson, K. Toth, R. Abbott, J. Osterling, J. Munson, A. Estes, and J. Liaw. Early social attention impairments in autism: social orienting, joint attention, and attention to distress. *Dev Psychol*, 40(2):271–283, Mar 2004.

[13] C. L. Kleinke. Gaze and eye contact: a research review. *Psychol Bull*, 100(1):78–100, Jul 1986.

[14] W. Jones, K. Carr, and A. Klin. Absence of preferential looking to the eyes of approaching adults predicts level of social disability in 2-year-old toddlers with autism spectrum disorder. *Arch. Gen. Psychiatry*, 65(8):946–954, Aug 2008.

[15] T. Falck-Ytter and C. von Hofsten. How special is social looking in ASD: a review. *Prog. Brain Res.*, 189:209–222, 2011.

[16] S. R. Langton, R. J. Watt, and I. Bruce. Do the eyes have it? Cues to the direction of social attention. *Trends Cogn. Sci. (Regul. Ed.)*, 4(2):50–59, Feb 2000.

[17] E. Murphy-Chutorian and M. M. Trivedi. Head pose estimation in computer vision: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31(4):607–626, April 2009.

[18] D. W. Hansen and Q. Ji. In the eye of the beholder: A survey of models for eyes and gaze. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(3):478–500, March 2010.

[19] F. Vicente, Z. Huang, X. Xiong, F. De la Torre, W. Zhang, and D. Levi. Driver gaze tracking and eyes off the road detection system. *IEEE Transactions on Intelligent Transportation Systems*, 16(4):2014–2027, Aug 2015.

[20] R. Valenti, N. Sebe, and T. Gevers. Combining head pose and eye location information for gaze estimation. *IEEE Transactions on Image Processing*, 21(2):802–815, Feb 2012.

[21] Visage Technologies. visage|SDK. Available at `http://visagetechnologies.com/products-and-services/visagesdk/`, version used 7.4.25, accessed March, 2016.

Detection and Visibility Estimation of Surface Defects under Various Illumination Angles using Bidirectional Reflectance Distribution Function and Local Binary Pattern

Petra Gospodnetić
Department of Electronic Systems and Information Processing
University of Zagreb, Faculty of Electrical Engineering and Computing, Croatia
petra.gospodnetic@gmail.com

Falco Hirschenberger
Department of Image Processing
Fraunhofer-Institut für Techno- und Wirtschaftsmathematik
Kaiserslautern, Germany
falco.hirschenberger@itwm.fraunhofer.de

***Abstract* – In the development of surface defect inspection systems, the surface illumination often plays a key role for the detectability of the defect. The illumination setup is currently configured manually for each type of defect which needs to be detected. This paper presents the use of a local binary pattern operator together with the bidirectional reflectance distribution function in order to detect various surface defects and estimate their visibility. This method is useful for improving the reliability of the visual surface inspection system and shortening of the time required for conducting a feasibility study. The reflectance of the sample material is acquired through a custom-built image acquisition system, which uses a robot arm in order to automatically retrieve the data under different illumination angles. By combining texture description with reflectance information, it is possible to localize defects without prior knowledge of their characteristics, whereas the defect visibility estimation requires manual ground truth marking in order to produce realistic results.**

Keywords – local binary pattern, image acquisition, surface defect detection, bidirectional reflectance, distribution function, visibility estimation, illumination set-up, texture analysis

I. INTRODUCTION

Automatic visual surface inspection through image processing is a method of non-invasive surface quality inspection for industrial purposes. Even though the implementation of the inspection system appears quite straightforward at first, the detection process soon becomes complex since there is no unambiguous parameterized definition of a defect and its image varies with surface properties such as material, texture and reflectivity. Attempts [1] have been made to define the surface defects descriptively by their shape and source of origin. Since such approaches are inefficient and unreliable, various texture analysis approaches have been proposed [2], but all of them are relying on a stable defect visibility. Therefore, the first step in the development of an inspection system is to make the visibility of each defect stable by determining the appropriate light conditions during the image acquisition phase. The adjustment of the light conditions is done manually, the visibility of a defect is determined heuristically and the duration of the overall process is often measured in days.

In this paper we propose a method for an automatic defect detection, additionally expanded with a defect visibility evaluation. The main advantage of the proposed defect detection method is that it requires no previous knowledge of the defect characteristics. It is carried out using a combination of the bidirectional reflectance distribution function (BRDF) and local binary patterns (LBP). The BRDF-LBP approach combines surface light response (radiance) information with its illumination-dependent texture characteristics in order to distinguish the defective from the correct surface. The visibility of the detected defects is further evaluated using manual ground truth marking in order to obtain the ratio of the found defect size to the ground truth size.

BRDF data is traditionally measured using gonioreflectometers [3], but Ward [4] introduced one of the first methods which used a digital camera instead. In [5] Dana extended the measurement and use of the BRDF for investigation of the visual appearance of real-world surfaces and the dependence of their appearance on imaging conditions. In order to obtain persistent image acquisition conditions, a robot arm was introduced into the setup as a sample holder. In their work, camera angles were changed manually and the light source was fixed, whereas in our paper a somewhat different setup was used – both the camera and the sample plane were fixed, while the robot arm carried the light source.

Defect detection using the LBP operator has already been proven useful for surface defect identification and localization for patterned fabric [6]. It has often been used for texture classification ([2], [7], [8]), while in [9] it was used in combination with a self-organizing map as a classifier for real-time paper surface inspection. For the purpose of this paper, LBP is used as a method for confirmation of the discovered BRDF defect candidates.

As far as known from the available literature [2], surface defect detection has been done using different approaches (statistical, structural, filterbased, modelbased) in order to identify potential defect candidates, but a visibility factor has

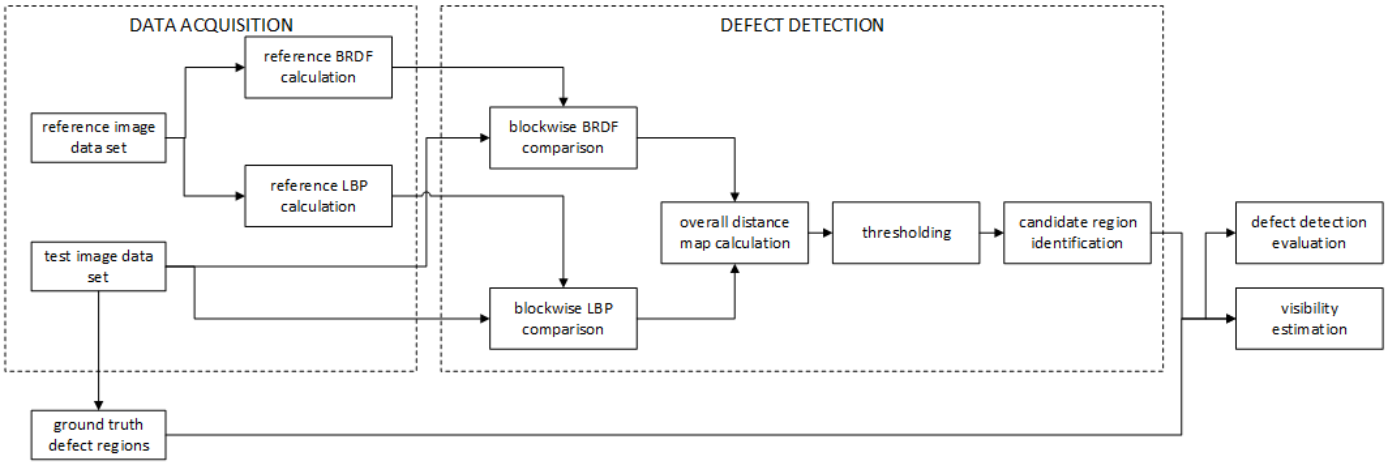


Figure 1 BRDF-LBP method pipeline

been presented only once before by Muehlemann [1], who used no defect detection but instead used manual extraction of a pixel vector across the center for each found defect in order to evaluate the amount of contrast change within the area. On the contrary, this paper proposes an approach which requires only one ground truth image to be used in the visibility estimation.

Further, in chapter II of this paper the methods contributing to our approach will be introduced, followed by the presentation of the obtained results in chapter III and the conclusion in chapter IV.

II. Materials and Methods

The surface texture in an image is a planar representation of the surface geometry, shaped by the highlights and the shadows of the present microstructure. Therefore its texture characteristics will change with the light conditions under which the image is acquired. A defect can be defined as an unwanted change in surface geometry and/or color. For a flat homogeneous material, the texture is uniformly spread and the surface radiance is consistent, which implies that any defect would be represented as a change in texture pattern and radiance. The method used to find and evaluate defects in such a way has three main steps, as presented in Figure 1:

- data acquisition through a parameterized image acquisition and the BRDF and LBP feature extraction;
- defect detection through a comparison of the test sample features to the reference sample features; and
- evaluation of the defect detection process and defect visibility estimation through the comparison of obtained defect candidates to manually marked ground truth defects.

A. Image acquisition

Obtaining measurable and reproducible image acquisition conditions required the construction of a standalone custom acquisition setup represented in Figure 2. It consisted of three main parts: a fixed camera, a movable light source and a flat sample plate. Since the main goal of this work was to assess the defect visibility for various light angles, all other system variables needed to be fixed. Therefore the camera was always mounted perpendicular to the sample surface with a fixed resolution, focal length and aperture. The light source was mounted on a robotic arm, which allowed for the light source to be moved incrementally by an angle of θ over the plate, following a precise arch line. In that way only the light angle α towards the camera was being changed, while maintaining other light source characteristics such as incident irradiance, distance and orientation fixed in relation to the surface plane.

For each new $\theta \epsilon [0, \pi]$, a new image was acquired. Variation of the angle α implied a variation in surface radiance, where the difference between the minimum and the maximum radiance was likely to exceed the dynamic range of the camera, thus providing images with marginal conditions (under- and overexposed images), as visible in Figure 3. While images with marginal conditions contain less information, it is possible that some defects may be easier to distinguish in that way. Therefore, these images should not be excluded from the further analysis. The complete data acquisition required that each light position be captured with an appropriate exposure. In order to fulfill this requirement, as well as to obtain the marginal conditions, multiple acquisition arches were repeated, with each arch having a constant exposure time.

The exact exposure time of each acquisition arch was determined using an automatic exposure evaluation, as presented in [10]. The exposure time used for the first acquisition arch was evaluated for the first image to be acquired. All images acquired in one acquisition arch were further marked as being either correct or incorrect by measuring the mean value of each image. If the mean value was not in the central interval of the pixel value range, the image was considered to be incorrectly exposed.

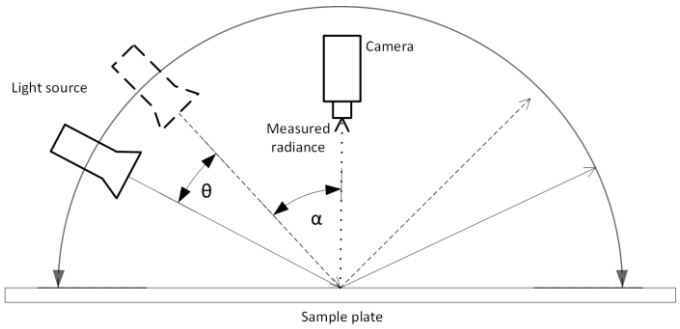


Figure 2 Image acquisition setup

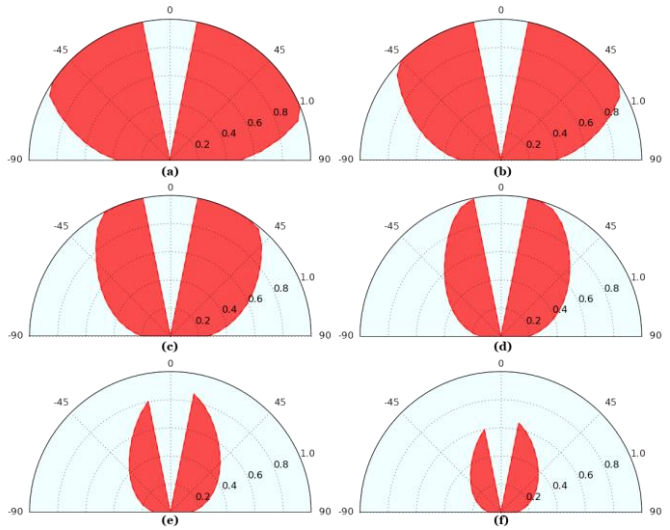


Figure 3 BRDF response for six acquisition arches, where the red field represents the acquired reflectance response of the sample at angles of -80 to 80 degrees, with an increment θ of five degrees. The exposure time used for the acquisition shortens from (a) to (f) respectively.

Further exposure evaluation was done by evaluating the first light position for which the image was marked as incorrect in the preceding acquisition. The process was repeated until each light position was captured with an appropriate exposure in at least one of the acquisition arches.

Exposure times were evaluated only for the reference sample acquisition and were afterwards saved. The saved exposure times were later used in the test sample acquisition in order to ensure that the test sample was acquired under the same conditions as the reference sample.

B. *Feature extraction*

When light hits a surface, it is either reflected, transmitted or absorbed. For opaque materials, the majority of the incident light is transformed into reflected and absorbed light. BRDF f_r [11] provides a complete reflectance information of an opaque surface at a single point x and is evaluated as the ratio of the radiance L exiting the surface in a given direction to the incident irradiance I from an incident solid angle $d\omega_i$ at a given illumination direction, as represented in Figure 4.

$$f_r(\theta_i, \varphi_i, \theta_e, \varphi_e) = \frac{dL(\theta_e, \varphi_e)}{dI(\theta_i, \varphi_i)} \tag{1}$$

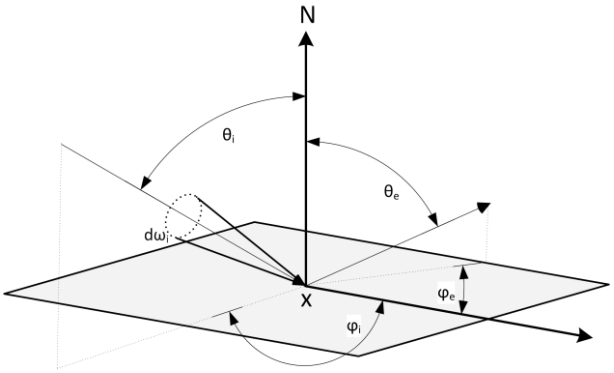


Figure 4 Surface reflection model

For the purpose of this work, the surface of a sample plate was assumed to be an isotropic material as proposed in [12], thus making the reflectance independent of the rotation of the incident and emerging directions about the surface normal, and dependent only on their relative difference $\Delta\varphi = \varphi_e - \varphi_i$. Therefore the BRDF function used for this work is dependent only on three variables $f_r(\theta_i, \theta_e, \Delta\varphi)$, of which θ_e and $\Delta\varphi$ were fixed throughout the acquisition process.

The LBP operator, originally introduced by [13], is a computationally efficient and monotonic gray-scale change invariant operator, commonly used for texture classification. The motivation for using LBP in defect visibility estimation came from the assumption that the defects were represented as a change of pattern within the surface texture and were therefore detectable when compared to a previously acquired reference pattern. The LBP operator labels the image pixels by comparing each of them to its 3x3 neighborhood and summing the results as weighted values by the power of two (2) if the neighborhood pixel is greater than or equal to the center pixel. Since textures can come in different sizes and rotations, the LBP operator was later extended in two ways [13]: to use neighborhoods of various sizes and to use only uniform binary patterns as descriptors. Neighborhoods of different sizes were determined by (P, R), where P represented the number of sampling points and R was the radius of the neighborhood. In that case the sampling points were evenly spaced on a circle around the central pixel g_c with coordinates (x, y) and a sample point pixel g_p whose coordinates were given by $(x_p, y_p) = (x + R\cos(2\pi p/P), y - R\sin(2\pi p/P))$, where $p \in [0, P-1]$. The function s was used to exclude the difference between g_c and g_p from the overall sum if g_p was greater than g_c. When sampling point coordinates did not fall in the center of a pixel, bilinear interpolation was used for an estimation of the value in that point. Uniform patterns were introduced in order to make operator rotation invariant. A pattern is considered uniform only if the pattern contains no more than two bitwise transitions from 0 to 1 or vice versa. In the computation of the LBP histograms, uniform patterns were assigned separate bins, while non-uniform patterns are assigned to a single bin. That way, for a 3x3 neighborhood, there were only 58, instead of 256 different patterns.

$$LBP_{P,R} = \sum_{p=0}^{P-1} s(g_p - g_c)2^p,$$
$$s(x) = \begin{cases} 1, & if\ x \geq 0 \\ 0, & otherwise \end{cases} \tag{2}$$

C. *Defect detection*

For each acquired image in the reference image data set both BRDF and LBP were calculated separately (Figure 1) for later use as baseline features. As both the reference and the test sample were acquired under the same conditions, the defect detection process was always comparing two images (one from the reference set and one from the test set) corresponding in light position and exposure time. By comparing the corresponding images, we searched for deviations from the baseline features.

Baseline BRDF features described a mean expected reflectance response of the material at a given position and were compared to the mean reflectance response in 5x5 sub-blocks of the test sample image. At the same time, the baseline LBP

features described the reference sample image divided into 40x40 sub-blocks, meaning that every block was described with its own LBP histogram.

The BRDF and LBP comparison results were given as two 2D distance maps, both matching pixel positions of the original test sample image. The BRDF map represented a distance from the expected reflectance response, while the LBP map was obtained by dividing the test image in the same manner as the reference image, into 40x40 sub-blocks, and then comparing the LBP histograms of the corresponding sub-block. Histograms were compared using χ^2 metrics (3), where h_1 and h_2 denote the reference sample histogram and test sample histogram respectively.

$$\chi^2(h_1, h_2) = \sum_{i=0}^{n-1} \frac{(h_{1_i} - h_{2_i})^2}{h_{1_i} + h_{2_i}} \quad (3)$$

Fusing the BRDF with the LBP feature calculation allowed us to apply the discriminative power of the LBP operator with the sensitivity of the BRDF to radiance changes. As visible from Figure 5, the distance maps obtained from the comparison of the reference and the test images were fused by multiplication. This multiplication resulted in mutual noise cancellation and an overall increase in solid defect labeling with more precision regarding its shape. Binarization of the newly acquired distance map was performed by low-level thresholding in order to remove the possible low-intensity noise, after which a

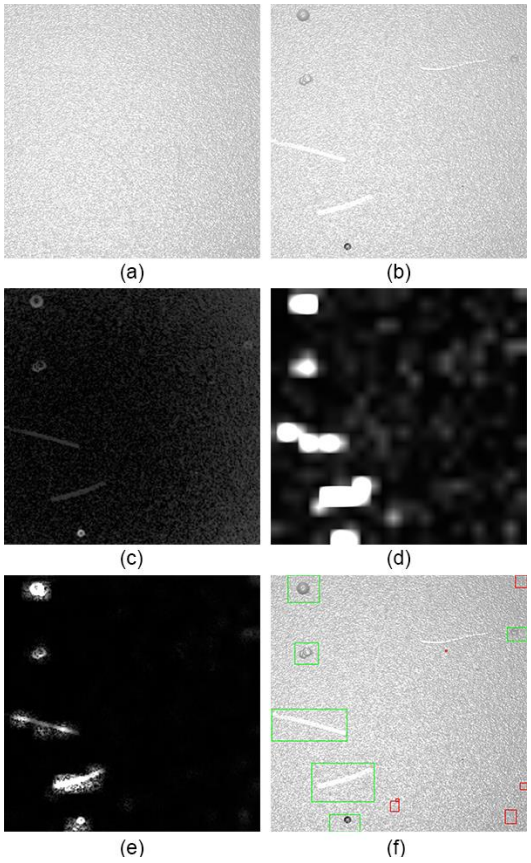


Figure 5 (a) - (f) Candidate region identification procedure for an image of a plastic board (sample 1): (a) reference sample image; (b) test sample image; (c) BRDF distance map; (d) LBP distance map; (e) BRDF-LBP distance map fusion; (f) detected defect candidates.

morphological dilation and closing was applied in order to connect candidates which may belong to the same defect. Values of both the low-level thresholding and the morphological operations were constant throughout the defect detection process. For the purpose of this work, they were determined heuristically.

D. Defect detection process and visibility estimation

At the beginning of the process, a user is given one well exposed image of the test sample and asked to mark known defects in the image, regardless of whether he may think them to be detectable or not in the further process. This image is then used twofold as ground truth: in the evaluation of the defect detection process and for visibility estimation. After the defect detection step, the identified candidate regions were being compared to the ground truth defect regions P. If they overlaped, the candidate was considered a true positive TP, whereas otherwise it was a false positive FP. From the obtained data, the defect detection process was evaluated by two measures: by the precision, also known as positive predictive value (PPV), and by the sensitivity, known as true positive rate (TPR):

$$PPV = \frac{TP}{TP + FP} \quad (4)$$

$$TPR = \frac{TP}{P} \ . \quad (5)$$

For the purpose of this paper we defined the defect visibility $v(D)$ as a relative size of the candidate region $R(C)$ to the ground truth defect region $R(D)$. A reason for this lies in the assumption that a bigger candidate implies a bigger spatial and textural deviation. It is a frequent occurrence that one defect is represented by more than one candidate, in which case the regions of all the candidates corresponding to the evaluated defect are summed up first and then compared to the ground truth region.

$$v(D) = \frac{\sum R(C_D)}{R(D)} \quad (6)$$

E. Experimental details

In order to perform the defect detection and visibility estimation, two samples of the same material were needed: the reference sample representing a defect-free surface and the test sample representing a surface containing various texture defects. For the purpose of this paper, the used reference sample surfaces were flat semi-glossy homogeneous materials: a white plastic board (sample 1) and a black metal-elastomer plate (sample 2).

The image acquisition was carried out using a UR10CB2 robot arm carrying a LED light source EFFI-Flex_5_525_1_3 of the wavelength 525 nm, with a diffused window and the lens set to the highest position. The distance between the light source and the sample plate was set to 0.53 m, at which point the illuminance was close to 1500 Lux, according to the official datasheet [14]. The camera used was a Basler industrial camera acA2500-14gm of resolution 2592 px ×1944 px, mounted with a Kowa 16mm/F 1.4 lens at the distance of 0.28 m. The acquisition process was fully automatic. The acquisition arches were repeated until all the light positions were imaged with an appropriate exposure as explained in section II.A. The duration of one acquisition arch was approximately 90 seconds.

III. RESULTS

The proposed BRDF-LBP approach was evaluated on two different homogeneous materials, as described in II.E, both containing various undefined textural defects. By *undefined*, we assume that the defect detection process had no prior knowledge of the defect locations or even of their existence. The ground truth marked image was used only for the evaluation of the process results and calculation of the defect visibility.

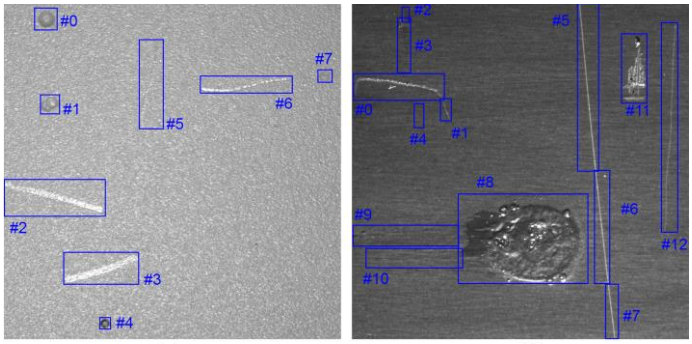


Figure 6 Numbered ground truth defect regions: (a) white plastic board (sample 1) and (b) black metal-elastomer plate (sample 2).

The white board sample contained eight different defects, while the metal-elastomer one contained ten. As can be seen from the ground truth image in Figure 6, some defects were split into sub-defects due to their size and orientation. Here we will present the defect detection performance (Table 1) and visibility plots for several specific defects from both samples. The selection of the presented results is based on the avoidance of redundancies resulting from the similarity of the defects. Defects of similar size and texture pattern produce similar visibility results.

Table 1 shows that the defect detection process had a high overall precision, while the overall sensitivity stayed around 55%. The sensitivity of the defect detection was expected and confirmed the assumption that an illumination angle revealed certain defects, while at the time hiding others. The results presented in Table 1 can be used as a baseline for future work on defect visibility calculation enhancement and automatization.

TABLE 1 DEFECT DETECTION PROCESS EVALUATION FOR WHITE PLASTIC BOARD (SAMPLE 1) AND BLACK METAL-ELASTOMER PLATE (SAMPLE 2)

	Sample 1		Sample 2	
Acquisition arch	**Precision [%]**	**Sensitivity [%]**	**Precision [%]**	**Sensitivity [%]**
1	80.5	55.7	55.9	57.4
2	85.7	56.2	71.5	70.7
3	86.6	59	88.7	62.5
4	89.5	62.	94.9	33.3
5	93.4	59.5		
6	96.7	52.9		
7	97.2	50		

A comparison of the identified candidates and ground truth defect positions by size and position resulted in visibility plots shown in Figure 7 (sample 1) and Figure 8 (sample 2). The visibility was plotted over 180 degrees for all obtained acquisition arches, for each defect separately. A visibility value of 1 marks high visibility whereas 0 marks no visibility. As can be noticed, all plots have a data gap between 85 and 95 degrees. The reason for this gap is of a technical nature – the camera was set directly above the sample plate and hence was casting a shadow, making the data acquisition impossible for those illumination angles.

The defect in Figure 7 (a) was a very sharp bump which was mostly visible since it clearly changed texture for the various illumination angles, whereas defect 2 in Figure 7 (b) had a very high peak for acquisition 6 (shortest exposure time), and was otherwise not clearly distinguishable. In Figure 7 (c) we can notice the shift of the visibility towards the perpendicular illumination angle as the exposure time shortened. Figure 7 (d) demonstrates that defect #5 would be visible from angles of around 70 degrees and 125 degrees.

Figure 8 shows that the defects on the metal-elastomer plate have their visibility peaks spread over the arch. Such results may be explained by metal-derived phenomena. Since the metal-elastomer plate is of a compound material of rubber and metal, it may partially exhibit specular reflections, which is a characteristic of metal. The specular component should not be treated as a problem for the purpose of this paper, but a further study should be made on the defect behavior on non-isotropic surfaces.

The overall process, including the execution of both the defect detection and the visibility estimation lasted approximately 30 minutes for sample 1 and approximately 20 minutes for the sample 2. The process included automatic image acquisition and defect detection, manual ground truth marking and visibility estimation.

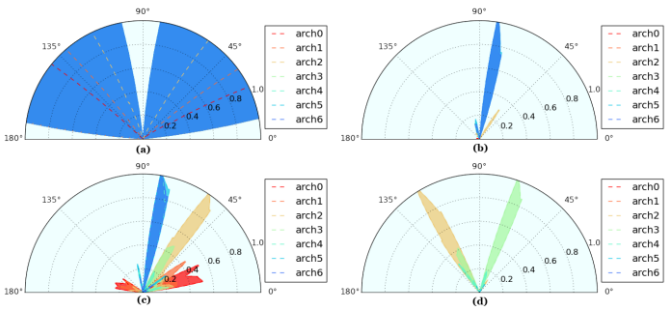


Figure 7 Visibility plots for four specific defects found on white board (sample 1): (a) defect #0; (b) defect #2; (c),defect #3; (d) defect #5

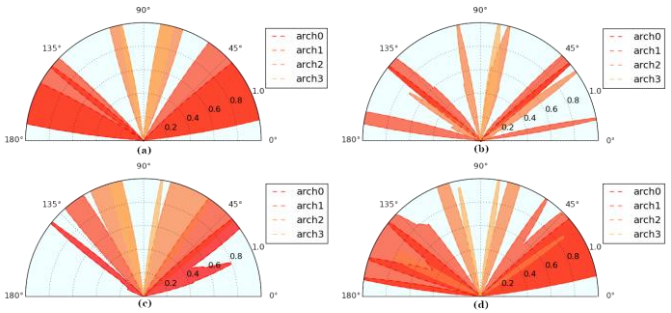


Figure 8 Visibility plots for four specific defects found on black metal-elastomer plate (sample 2): (a) defect #1; (b) defect #5; (c) defect #9; (d) defect #12

IV. Conclusion

The defect detection method using a combination of LBP and BRDF confirmed the possibility of surface defect detection without prior knowledge of the defect characteristics. The results presented in this paper show that the method yields high defect detection precision rates for materials having uniformly textured and nearly isotropic surfaces.

The introduced method shows stable and promising results for two samples. The samples were made of different materials, confirming that the method can be material-independent. This has to be evaluated further in order to achieve a verification of the method.

The expansion of the defect detection process with the defect visibility estimation makes our method very suitable for improving the design and development of visual surface inspection systems. The automatization of the image acquisition process ensures a precision which can hardly be obtained manually, as well as reproducibility. As previously stated, the process of manual visibility estimation is often measured in days. Therefore the time obtained in the presented work shows a significant improvement towards shortening the duration of the feasibility study.

References

[1] M. Muehlemann, "Standardizing Defect Detection for the Surface Inspection of Large Web Steel," 2000. [Online]. Available: http://www.illuminationtech.com/documents/surface_inspection.pdf. [Accessed 18 10 2015].

[2] X. Xie, "A Review of Recent Advances in Surface Defect Detection using Texture Analysis Techniques," *Electronic Letters on Computer Vision and Image Analysis*, vol. 7, no. 3, pp. 1-22, 2008, doi: 10.5565/rev/elcvia.268

[3] S. C. Foo, "A gonioreflectometer for measuring bidirectional reflectance of material for use in illumination computation.," Master's thesis, Cornell University, 1997.

[4] G. J. Ward, "Measuring and Modeling Anisotropic Reflection," *Computer Graphics*, no. 26, pp. 265-273, 1992.

[5] K. J. Dana, B. van Ginneken, S. K. Nayar and J. J. Koenderink, "Reflectance and Texture of Real-World Surface," *ACM Transactions on Graphics*, vol. 18, no. 1, pp. 1-34, 1999.

[6] F. Tajeripour, E. Kabir and A. Sheikhi, "Defect Detection in Patterened Fabrics using Modified Local Binary Patterns," in *International Conferrence on Computational Intelligence and Multimedia Applications*, 2007, doi: 10.1155/2008/783898

[7] H. Y. T. Ngan, G. Kwok Hung Pan and N. Hon Ching Yung, "Automated fabric defect detection - a review," *Image and Vision Computing*, vol. 29, no. 7, pp. 42-458, 2011, doi: 10.1016/j.imavis.2011.02.002

[8] O. Ghita, P. F. Whelan, T. Carew and P. Nammalwar, "Quality Grading of Painted Slates Using Texture Analysis," *Computers in Industry*, vol. 56, pp. 802-815, 2005, doi: 10.1016/j.compind.2005.05.008

[9] T. Mäenpää, M. Turtinen and M. Pietikäinen, "Real-Time Surface Inspection by Texture," *Real-Time Imaging*, vol. 9, no. 5, pp. 289-296, 2003, doi: 10.1016/S1077-2014(03)00041-X

[10] S. Battiatio, G. Messina and A. Castorina, "Exposure Correction for Imaging Devices: an Overview," *ingle-Sensor Imaging, Methods and Applications for Digital Cameras, Image Processing Series*, 2008, doi: 10.1201/9781420054538.ch12

[11] S. R. Marschner, S. H. Westing, E. P. F. Lafortune, K. E. Torrance and D. P. Greenberg, "Image-Based BRDF Measurement Including Human Skin," *Rendering Techniques '99*, pp. 139-152, 1999.

[12] W. Matusik, H. Pfister, M. Brand and L. McMillan, "Efficient Isotropic BRDF Measurement," in *Eurographics Symposium on Rendering*, 2003.

[13] T. Ojala, "Multiresolution Gray-Scale and Rotation Invariant Texture Classificiation with Local Binary Patterns," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp. 971-987, 2002, doi: 10.1109/TPAMI.2002.1017623

[14] effiLux, *Datasheet EFFI-Flex v.2.1*, Les Ulis, 2015.

Sensitivity of Tone Mapped Image Quality Metrics to Perceptually Hardly Noticeable Differences

Nikola Banić and Sven Lončarić
Image Processing Group
Faculty of Electrical Engineering and Computing
University of Zagreb, 10000 Zagreb, Croatia
E-mail: {nikola.banic, sven.loncaric}@fer.hr

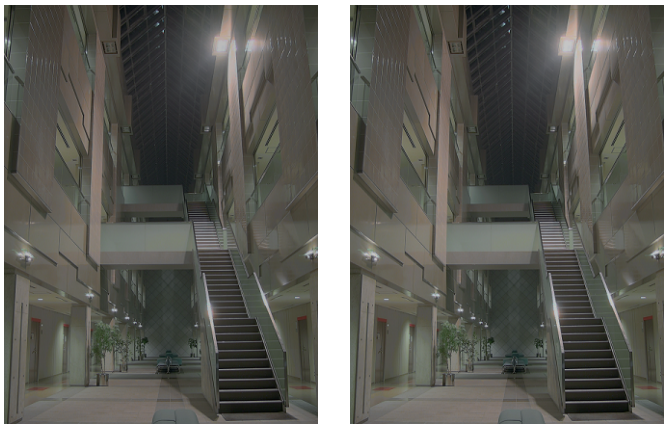
***Abstract*—As high dynamic range images are being used more widely, the need for good tone mapping operators (TMOs) i.e. methods for their conversion to low dynamic range images rises as well. In evaluation of results of TMOs objective image quality metrics are often used for practical reasons. Since these metrics only approximate perceptual evaluation, they are sometimes too sensitive to perceptually unimportant details. In this paper such sensitivities of three recent tone mapped image quality metrics are compared: TMQI, TMQI-II, and FSITM. These metrics have been chosen because they are the most appropriate objective quality metrics for the problem of tone mapping. The comparison is performed by using specifically designed tone mapped images to check the measures' susceptibility to perceptually unnoticeable changes in brightness of the resulting image. It is shown that while values of TMQI and FSITM are only slightly affected by such changes, the recent TMQI-II can obtain significantly different values, which brings into question its ability perform a fair TMO comparison. The results are presented and discussed.**

***Index Terms*—High dynamic range, objective image quality assessment, FSITM, low dynamic range, TMQI, TMQI-II, tone mapping.**

I. Introduction

Images with high dynamic range (HDR) i.e. with a high ratio between the largest and smallest intensity are being more widely used with the advance of imaging technology [1]. Since most display devices still support only low dynamic range (LDR) images, there is a need for tone mapping operators (TMOs) i.e. for dynamic range compression methods that convert HDR images to their LDR versions. Tone mapping is a challenging problem and therefore many TMOs have been proposed so far. TMOs are global [2]–[10] if they handle same intensities in the same way across the whole image. On the other hand, if they handle intensities based on the content of their close neighborhood, then they are local [11]–[18]. The main characteristic of global TMOs is their speed and simplicity, while local TMOs are usually more complex and they produce better LDR images of higher quality [19]–[21].

An important part in development of TMOs is the quality evaluation of their results and an accurate way to do that is to perform subjective quality assessment. However, due to a large number of existing TMOs, subjectively comparing a new TMO even with only state-of-the-art TMOs on a larger testing dataset becomes in most cases too slow and impractical. For this reason the objective image quality metrics have been introduced and currently they are often used to simplify and speed up the quality assessment of LDR images produced by a TMO. Since these metrics only approximate subjective evaluation, it can happen that they assess two very similar images very differently i.e. they are sometimes too sensitive to differences that the human visual system does not even notice. If in such cases only the values of these metrics are taken into account, then these differences can erroneously lead to wrong conclusions about the actual performance of different TMOs.



(a) (b)

Fig. 1: Tone mapped images of the same scene. The values of TMQI, TMQI-II, and FSITM^G_TMQI quality measures are for (a) 0.8455, 0.5723, and 0.8352, respectively, and for (b) 0.8635, 0.8044, and 0.8394, respectively.

For this reason it is important to check to what degree are some of the widely used metrics sensitive to such differences. In this paper three recent tone mapped image quality metrics are tested for such sensitivity: TMQI [22], TMQI-II [23], and FSITM [24]. They are chosen because they are the most appropriate objective quality metrics designed specifically for quality assessment of tone mapped images. The testing is performed by using specifically designed tone mapped images to check the measures' sensitivity to alterations of mean brightness of the resulting LDR images. It is demonstrated that for images with slight, but perceptually unnoticeable mean brightness differences the results of TMQI-II can be significantly different, while the values of TMQI and FSITM

are affected on a much smaller scale, even in the worst case. This brings into question TMQI-II's practical usability and credibility in fair evaluation and comparison of quality of LDR resulting images produced by using different TMOs.

The paper is structured as follows: Section II describes three recent objective tone mapped image quality metrics, Section III gives the motivation for the comparison of their sensitivity to perceptually unnoticeable differences, in Section IV the comparison is performed and its results are presented and discussed, and Section V concludes the paper.

II. Objective tone mapped image quality metrics

Subjectively assessing the quality of LDR images obtained after carrying out tone mapping of the initial HDR images often results in good and relatively accurate comparison of performance of various TMOs. However, a large drawback of such subjective assessment is that it is slow and it usually takes a lot of time. It also makes TMO development based on measuring improvement over some other TMOs impractical due to the lack of automation. For this reason various objective quality metrics for tone mapped images have been introduced.

One of the widely used objective quality metrics that was also one of the first ones designed specifically for the purpose of objective quality assessment of tone mapped images is the Tone Mapped image Quality Index (TMQI) [22]. It evaluates the structural fidelity and statistical naturalness of a tone mapped image by comparing it to the original HDR image. The final result is a real number in range $[0, 1]$ with a higher value meaning higher quality and vice versa. In [23] TMQI has been upgraded to TMQI-II, which is supposed to be its improved version and additionally there is an iterative procedure for improving an initially tone mapped image in terms of its TMQI-II value. Another recent metric is the Feature Similarity Index For Tone-Mapped Images (FSITM), which is based on local phase information of images and like TMQI-II it was also shown to outperform TMQI. If FSITM is combined with TMQI, it gives better results and for this combination the notation FSITM^C_TMQI [24] is used where C is a color channel. In the rest of the paper the green (G) channel is used because the authors have shown that its usage gives good results. The combination FSITM^G_TMQI was shown [25] to outperform both TMQI and TMQI-II as well. It should be mentioned that these three metrics are currently state-of-the-art in the area of objective quality assessment of tone mapped images. The main advantage of objective quality assessment over subjective quality assessment is that it can be used to automate the evaluation of the performance of a TMO.

However, since objective metrics only approximate perceptual subjective evaluation, there are possible cases where the comparison results obtained by objective and subjective quality assessment significantly differ. A particularly interesting case is when an objective metric is too sensitive to perceptually unnoticeable differences that should be disregarded. This is clearly a drawback because such and similar cases can erroneously lead to unfair comparison of even very similar TMOs.

(a) (b)

Fig. 2: Crudely increasing mean image brightness by placing a white rectangle in it with everything else remaining the same. The values of TMQI, TMQI-II, and FSITM^G_TMQI indices are for (a) 0.7993, 0.3861, and 0.8448, respectively, and for (b) 0.7754, 0.8195, and 0.7980, respectively.

(a) (b)

Fig. 3: Crudely decreasing mean image brightness by placing a black rectangle in it with everything else remaining the same. The values of TMQI, TMQI-II, and FSITM^G_TMQI indices are for (a) 0.9043, 0.4701, and 0.8646, respectively, and for (b) 0.9019, 0.8410, and 0.8395, respectively.

III. Motivation for comparison

The direct motivation for a comparison between sensitivities of different objective tone mapped image quality metrics were the observed significant fluctuations of TMQI-II values for the same images before and after slightly changing their mean brightnesses by multiplying them by a constant. An example is shown in Fig. 1 where the difference of image grayscale means is less than 5. By performing some additional similar experiments with manipulation of image grayscale mean, it becomes evident that TMQI-II is so susceptible to image brightness that it sometimes puts it before the content. If e.g. the mean image brightness is adjusted by introducing artificial content as in Fig. 2 and Fig. 3, the values of TMQI and FSITM^G_TMQI decrease as intuitively expected, but the values of TMQI-II increase significantly despite a clear

loss of information. The shown examples are not some rare, specially designed cases and similar results can be obtained for practically any other tone mapped images as well.

Since the shown examples with large content manipulations are highly unlikely to be encountered during development of new TMOs, the mentioned metrics' sensibility should be tested in more realistic conditions. Nevertheless, these examples can point in the direction of more suitable sensitivity tests.

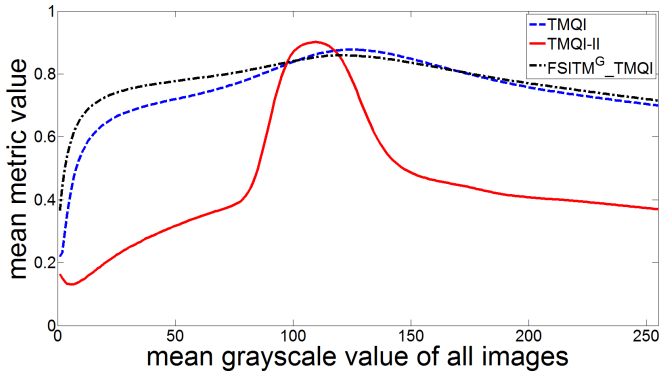


Fig. 4: The impact of forcing all tone mapped images to a given mean grayscale value on the mean metric values.

IV. Experimental results

A. Experimental setup

For further experiments the HDR images available at [26] were used. They were used mainly because they originate from different sources and some of them were even artificially generated, which means that altogether they cover a larger variety of HDR image types. The initial step before carrying out any other experiments was to tone map all of these images by applying Reinhard's TMO [13] implemented in the open-source Luminance HDR software with the same default parameters being used for all images. Reinhard's TMO was chosen mainly because it is a widely known and used TMO and it was shown to give high quality results. Results very similar to the ones described later in the paper could also be obtained by using some other TMO as well. If $\mathbf{I}^{(i)}$ is the LDR resulting image obtained by applying Reinhard's TMO to i-th of the initial HDR images, then for each $\mathbf{I}^{(i)}$ the next step was to create two additional images $\mathbf{I}_A^{(i)}$ and $\mathbf{I}_B^{(i)}$. Their respective j-th pixels were calculated as $\mathbf{I}_A^{(i)}(j) = k_A^{(i)}\mathbf{I}^{(i)}(j)$ and $\mathbf{I}_B^{(i)}(j) = k_B^{(i)}\mathbf{I}^{(i)}(j)$ with $k_A^{(i)} < k_B^{(i)}$. The values of constants $k_A^{(i)}$ and $k_B^{(i)}$ were deliberately chosen so that the difference between a chosen objective quality metric for $\mathbf{I}_A^{(i)}$ and $\mathbf{I}_B^{(i)}$ was maximized under two constraint. The first constraint was that the mean CIELab E_{ab}^* approximated perceptual difference between corresponding pixels of $\mathbf{I}_A^{(i)}$ and $\mathbf{I}_B^{(i)}$ must stay below the just-noticeable difference (JND) threshold of 2.3 [27]. The second constraint was that the values of $k_A^{(i)}$ and $k_B^{(i)}$ must be from set $\{50, 51, ..., 200\}$ to exclude the possibility of unnaturally looking images that could be obtained for too high or too low values of constants $k_A^{(i)}$ or $k_B^{(i)}$.

When this was done for all images $\mathbf{I}^{(i)}$, the result were two new sets A and B with corresponding images $\mathbf{I}_A^{(i)}$ and $\mathbf{I}_B^{(i)}$, which were designed to have slight differences that are perceptually hardly noticeable or not noticeable at all like the ones in Fig. 1. This effectively means that the values of a good objective quality metric for an image from set A and for its corresponding image in set B should differ only slightly. To check whether that holds for TMQI, TMQI-II, and FSITM^G_TMQI, the two mentioned sets were created for each of these metrics and the quality of the obtained images in them was evaluated by calculating these same metrics for them.

B. Numerical results

Table I shows mean values of all objective quality metrics for sets A and B created to maximize the difference for specified metrics. It can be seen that in the individual metrics' worst case scenario of sensitivity to perceptually hardly noticeable differences only TMQI-II is significantly affected. To describe this better, Mann-Whitney U test [28] of the null hypothesis that the distribution of metric values for images in set A is identical to distribution of metric values for images in set B was performed for each metric's worst case. The p-values obtained during the tests for TMQI, TMQI-II, and FSITM^G_TMQI were 0.0811, $3.0260 \cdot 10^{-12}$, and 0.0343, respectively, which clearly shows that TMQI-II is too sensitive.

Another experiment was performed to illustrate the problem more clearly. The initial HDR images were tone mapped by using Reinhard's TMO as was done earlier, but then each image was multiplied by a constant in order to set its mean pixel grayscale value to 1 and then the mean value of all metrics on these images was calculated. This was then repeated by setting the mean pixel grayscale value to every integer in interval $[1, 255]$. The obtained results were as shown in Fig. 4.

C. Discussion

Although in some cases TMQI-II can fail drastically as demonstrated by Table I and Figures 1, 2, and 3, this happens only when the mean grayscale value of a tone mapped image is near the steep parts of the TMQI-II curve shown in Fig. 4. A possible abuse of this situation would be to include this knowledge into a TMO only in order to get a better TMQI-II score and thus seemingly outperform other TMOs. Another less malign case that does not involve including this knowledge into a TMO is when a new TMO accidentally happens to give more images with mean grayscale values favorably valued by TMQI-II than other TMOs do. Since TMQI and FSITM^G_TMQI do not suffer so seriously from this problem, they are probably a significantly better metric choice for evaluating different TMOs in order to determine which of them is supposed to produce results of higher quality. However, it should be mentioned that the results obtained by the iteratively improving an initially given LDR image to gradually improve its TMQI-II metric value [23] still gives high quality results.

V. Conclusions

The sensitivities of several tone mapped image quality metrics to hardly noticeable and unnoticeable differences

TABLE I: Mean values of all objective quality metrics for sets A and B created to maximize the difference for specified metrics.

	Created to maximize TMQI difference			Created to maximize TMQI-II difference			Created to maximize FSITM^G_TMQI difference		
Created dataset	**TMQI**	**TMQI-II**	**FSITM^G_TMQI**	**TMQI**	**TMQI-II**	**FSITM^G_TMQI**	**TMQI**	**TMQI-II**	**FSITM^G_TMQI**
A	0.7920	0.4847	0.7956	0.8040	0.5481	0.8218	0.7805	0.4492	0.7856
B	0.8147	0.5110	0.8130	0.8178	0.7506	0.8290	0.8012	0.4824	0.8049

in images were compared. For two of them, TMQI and FSITM^G_TMQI, it was shown that this sensitivity is not so high. On the other hand, however, in the case of TMQI-II it was shown to be very high, which can represent a significant problem in practical applications of this metric. A conclusion that can be drawn from the presented experimental results is that for comparison of results of different TMOs it is better to use TMQI and FSITM^G_TMQI instead of TMQI-II.

REFERENCES

[1] E. Reinhard, W. Heidrich, P. Debevec, S. Pattanaik, G. Ward, and K. Myszkowski, *High dynamic range imaging: acquisition, display, and image-based lighting*. Morgan Kaufmann, 2010.

[2] J. Tumblin and H. Rushmeier, "Tone reproduction for realistic images," *Computer Graphics and Applications, IEEE*, vol. 13, no. 6, pp. 42–48, 1993.

[3] K. Chiu, M. Herf, P. Shirley, S. Swamy, C. Wang, K. Zimmerman *et al.*, "Spatially nonuniform scaling functions for high contrast images," in *Graphics Interface*. CANADIAN INFORMATION PROCESSING SOCIETY, 1993, pp. 245–245.

[4] G. Ward, "A contrast-based scalefactor for luminance display," *Graphics gems IV*, pp. 415–421, 1994.

[5] C. Schlick, "Quantization techniques for visualization of high dynamic range pictures," in *Photorealistic Rendering Techniques*. Springer, 1995, pp. 7–20.

[6] S. N. Pattanaik, J. Tumblin, H. Yee, and D. P. Greenberg, "Time-dependent visual adaptation for fast realistic image display," in *Proceedings of the 27th annual conference on Computer graphics and interactive techniques*. ACM Press/Addison-Wesley Publishing Co., 2000, pp. 47–54.

[7] F. Drago, K. Myszkowski, T. Annen, and N. Chiba, "Adaptive logarithmic mapping for displaying high contrast scenes," in *Computer Graphics Forum*, vol. 22, no. 3. Wiley Online Library, 2003, pp. 419–426.

[8] E. Reinhard and K. Devlin, "Dynamic range reduction inspired by photoreceptor physiology," *Visualization and Computer Graphics, IEEE Transactions on*, vol. 11, no. 1, pp. 13–24, 2005.

[9] G. W. Larson, H. Rushmeier, and C. Piatko, "A visibility matching tone reproduction operator for high dynamic range scenes," *Visualization and Computer Graphics, IEEE Transactions on*, vol. 3, no. 4, pp. 291–306, 1997.

[10] G. J. Braun and M. D. Fairchild, "Image lightness rescaling using sigmoidal contrast enhancement functions," *Journal of Electronic Imaging*, vol. 8, no. 4, pp. 380–393, 1999.

[11] J. Tumblin and G. Turk, "LCIS: A boundary hierarchy for detail-preserving contrast reduction," in *Proceedings of the 26th annual conference on Computer graphics and interactive techniques*. ACM Press/Addison-Wesley Publishing Co., 1999, pp. 83–90.

[12] F. Durand and J. Dorsey, "Fast bilateral filtering for the display of high-dynamic-range images," *ACM transactions on graphics (TOG)*, vol. 21, no. 3, pp. 257–266, 2002.

[13] E. Reinhard, M. Stark, P. Shirley, and J. Ferwerda, "Photographic tone reproduction for digital images," in *ACM Transactions on Graphics (TOG)*, vol. 21, no. 3. ACM, 2002, pp. 267–276.

[14] R. Fattal, D. Lischinski, and M. Werman, "Gradient domain high dynamic range compression," in *ACM Transactions on Graphics (TOG)*, vol. 21, no. 3. ACM, 2002, pp. 249–256.

[15] R. Mantiuk, K. Myszkowski, and H.-P. Seidel, "A perceptual framework for contrast processing of high dynamic range images," *ACM Transactions on Applied Perception (TAP)*, vol. 3, no. 3, pp. 286–308, 2006.

[16] L. Meylan and S. Susstrunk, "High dynamic range image rendering with a retinex-based adaptive filter," *Image Processing, IEEE Transactions on*, vol. 15, no. 9, pp. 2820–2830, 2006.

[17] N. Banić and S. Lončarić, "Color Badger: A Novel Retinex-Based Local Tone Mapping Operator," in *Image and Signal Processing*. Springer, 2014, pp. 400–408.

[18] ——, "Puma: A High-Quality Retinex-Based Tone Mapping Operator," in *Signal Processing Conference (EUSIPCO), 2016 24rd European*. IEEE, 2016, pp. 943–947.

[19] J. Kuang, H. Yamaguchi, G. M. Johnson, and M. D. Fairchild, "Testing HDR image rendering algorithms," in *Color and Imaging Conference*, vol. 2004, no. 1. Society for Imaging Science and Technology, 2004, pp. 315–320.

[20] J. Kuang, H. Yamaguchi, C. Liu, G. M. Johnson, and M. D. Fairchild, "Evaluating HDR rendering algorithms," *ACM Transactions on Applied Perception (TAP)*, vol. 4, no. 2, p. 9, 2007.

[21] C. Urbano, L. Magalhães, J. Moura, M. Bessa, A. Marcos, and A. Chalmers, "Tone mapping operators on small screen devices: an evaluation study," in *Computer Graphics Forum*, vol. 29, no. 8. Wiley Online Library, 2010, pp. 2469–2478.

[22] H. Yeganeh and W. Zhou, "Objective Quality Assessment of Tone Mapped Images," *Image Processing, IEEE Transactions on*, vol. 22, no. 2, pp. 657–667, 2013.

[23] K. Ma, H. Yeganeh, K. Zeng, and Z. Wang, "High dynamic range image compression by optimizing tone mapped image quality index," *Image Processing, IEEE Transactions on*, vol. 24, no. 10, pp. 3086–3097, 2015.

[24] H. Ziaei Nafchi, A. Shahkolaei, R. Farrahi Moghaddam, and M. Cheriet, "FSITM: A Feature Similarity Index For Tone-Mapped Images," *Signal Processing Letters, IEEE*, vol. 22, no. 8, pp. 1026–1029, 2015.

[25] ——. (2015, Nov.) FSITM: A Feature Similarity Index For Tone-Mapped Images (Supplementary material). [Online]. Available: http://www.synchromedia.ca/system/files/FSITM_Sup.pdf

[26] High Dynamic Range Image Examples, month=sep, year=2015, url=http://www.anyhere.com/gward/hdrenc/pages/originals.html.

[27] M. Mahy, L. Van Eycken, and A. Oosterlinck, "Evaluation of uniform color spaces developed after the adoption of CIELAB and CIELUV," *Color research and application*, vol. 19, no. 2, pp. 105–121, 1994.

[28] R. Kirk, *Statistics: an introduction*. Cengage Learning, 2007.

Detection and Localization of Spherical Markers in Photographs

Josip Tomurad
Faculty of Electrical Engineering and Computing
University of Zagreb
Zagreb, Croatia
e-mail: josip.tomurad@fer.hr

Marko Subašić
Faculty of Electrical Engineering and Computing
University of Zagreb
Zagreb, Croatia
e-mail: marko.subasic@fer.hr

***Abstract* - This paper presents two solutions for detection and localization of spherical markers in photographs. The proposed solutions enable precise detection and localization in sub millimeter range. High precision localization is required for brain surgery, and presented research effort is part of a project of developing and deploying a robotic system for neurosurgical applications. Two algorithms for Hough transform using several edge detection algorithms are proposed, and their results compared and analyzed. Results are obtained for both NIR and visible spectrum images, and required high precision is achieved in both domains.**

Keywords – project RONNA, image processing, circle detection, Hough transform, edge detection

I. INTRODUCTION

Project RONNA (Robotic Neuronavigation) is focused in research and development of a robotic system for neurosurgical application. [1] The system consists of two robotic arms assisting the doctor in brain surgery. At the first public demonstration of the project, an operation was demonstrated in which RONNA was tasked with precise drilling of the skull. It also enabled access to the tumor in the patient's brain with accuracy of less than a millimeter. Very precise estimation of the robots position is clearly an important task in the process.

This paper is focused on the specific task of this project, detection of spherical markers that the robot uses to estimate its position in space. Detection is performed using cameras mounted on the robot. The markers are spheres coated in special paint that appears black in visible spectrum, but reflects NIR light very well. The robot is also equipped with NIR light source, which makes the markers very bright in NIR images.

This paper investigates some well-known methods of edge and circle detection in images. Two variants of the Hough transform using several methods of edge detection were selected. Used algorithms, their advantages and disadvantages, as well as other problems that have an effect on rate of detection and localization precision are described.

II. HOUGH TRANSFORM

Hough transform is a technique of feature extraction that is used in image analysis, computer vision, and digital image processing. The purpose of this technique is finding imperfect instances of objects from a certain class of shapes by application of voting. The technique was originally used for line detection in images, and was later expanded to recognition of various kinds of shapes, most commonly ellipses and circles.

Hough transform uses shape edges as its input so first step is finding edge pixels in an image. Then each edge pixel votes in a Hough parameter space in a pattern that describes potential shape of interest. Finally, local maximums in the parameter space provide detection candidates.

Our goal is detection of spherical objects in 2D images so our shape of interest is a circle. In a two-dimensional space, a circle is described by:

$$(x-a)^2+(y-b)^2=r^2, \qquad (1)$$

where (a,b) are coordinates of the circle center, and r the radius. For a given edge point with coordinates (x,y) all possible combinations of parameters a, b and r can be found using the equation (1). In this case, possible parameter combinations lie on the surface of an inverted right-angled cone whose apex is at $(x,y,0)$. In a 3-D Hough parameter space a circle is defined by an intersection of many conic surfaces. The points where those intersections are located are local maximums in the accumulation matrix representing the parameter space. [2]

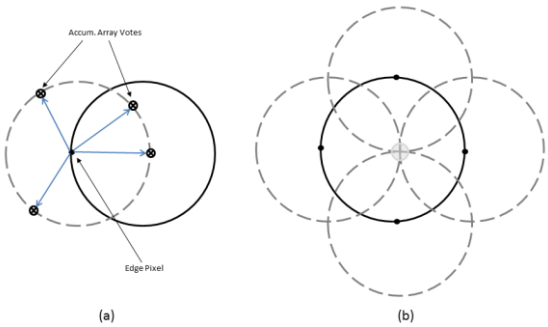


Figure 1. The voting process in the Hough transform [3]

This algorithm is implemented by iterating trough the expected radius range and voting in the accumulation matrix for every edge pixel in the image. Every edge pixel (x, y) increases value (gives vote to) of all matrix cells at coordinates (a, b, r), where r is fixed, and a and b are calculated by using equation (1). Votes for center candidates for selected radius are distributed in a circular pattern designated by the dashed line in Figure 1 (a). Figure 1 (b) shows detected center as a common intersection of these circular patterns originating from several edge points. Each detected center is a local maximum in the accumulation matrix and it needs to be above Hough accumulation threshold in order to be considered.

The accumulation matrix in the classic implementation of the Hough transform is tri-dimensional. Its dimensions are n, m, p, where n and m are image dimensions, and p is the size of radius range within which the circle is searched for. Because this demands a large amount of memory and a long processing time, it is common in modern practice to use a two-dimensional matrix, of dimensions n, m. In this case, only the location of the center is calculated, and this process demands an extra step of radius evaluation. The approach is less computationally demanding, especially if a larger radius range is used. In this paper both approaches are tested.

III. EDGE DETECTION

Hough transform uses object edges as its inputs, and there are many ways to detect them. Sobel operator, alone and in combination with the Gauss filter, as well as Canny edge detector, and Laplacian of Gaussian (LoG) were used in presented research. These methods differ in the way they calculate edge pixels, which entails different complexity, and gives different results. Relevant parameter for all methods is edge detection threshold, the minimum value of gradient a pixel must have to be considered part of an edge.

The Sobel operator works on a principle of gradient estimation in a digital image by summation of vectors of two orthogonal gradient estimations that can be calculated from a 3x3 neighborhood of a point. [4,5,6]

Canny edge detection is an edge detection operator that uses a multi-stage algorithm to detect a wide range of edges in an image. It provides very good and reliable detection. [7]

When searching for edges by using LoG, the image is first convolved by a Gaussian filter, in order to minimize noise. Subsequently, the Laplacian operator is applied to the image. [8] This method is also very precise and reliable, comparable to the Canny algorithm.

IV. METHODS

A. Hough transform with radius estimation

We utilized the Two-Stage method developed by Yuen et al. [9] and discussed by Davies. [10] This method works by utilizing voting of edge pixels to find the center of the circle, and the radius is estimated afterwards. This estimation is based on radial histograms. A 2-D accumulation matrix is used, with the benefit of a significant performance increase relative to a 3-D matrix. In order to optimize performance, information about edge gradient is used to enable voting only along a limited interval in the gradient direction. [11] Consequently this method requires edge detection methods that provide gradient angle. Edge detection can be preceded by Gauss smoothing, in order to reduce the number of detected pixels that are not actual edges.

Variable parameters of this method are the range of radii in which we circles are searched for, edge detection threshold, and Hough accumulation threshold.

B. Hough transform using a 3-D accumulation matrix

The first step of this method is edge detection using one of previously described methods. Then Hough transform is applied to the binary image, by first using midpoint circle algorithm [12] to calculate circle templates of all the radii in the relevant range. Those templates determine which accumulator elements need to be incremented relatively to the current pixel. Then comes the voting, done by iterating the calculated patterns through all the edge pixels and incrementing the appropriate elements of the accumulation matrix.

The next step is searching for as many local maximums in the accumulation matrix as there are circles we are searching for in the image. Neighborhood suppression is applied, meaning that after a particular maximum is found, its immediate neighborhood is no longer taken into consideration. This is due to the large chance that other local maximums will be found adjacent to the first one, but all of them are probably a consequence of the same circle.

The coordinates of local maximums are actual center positions and radii of detected circles. This method can be optimized by not searching for a particular number of circles, but setting a Hough accumulation. This is useful in when we don't know the number of circles in an image is unknown.

Variable parameters in this method are radius range, sigma (standard deviation of the Gauss distribution), and edge detection threshold.

V. RESULTS

MATLAB was used as the development environment in this paper.

The key feature in presented result tables is the deviation of center and radius estimation relative to the manually estimated gold standard ("Center delta" and "Radius delta" in the tables). Manual estimation was done by hand picking 6 points, and calculating the mean of all 20 possible circles those points define.

Infrared images of dimensions 1280x1024 pixels, and visible spectrum images of 2590x1942 pixels were used in experiments. NIR images are processed in the original size, while VS images are reduced to half the original size in order to speed up the process. Our database has 19 NIR and 20 VS images. The markers are 10.9 mm in diameter.

The processing times reported in following subsections are very dependent of the performance capabilities of the computer system used, but it can be used as a framework for comparison of different methods.

A. Hough transform with radius estimation

Radius range for NIR images is [95, 170], edge detection threshold is 0.1, and Hough accumulation threshold is 0.94.

Radius range for VS images is [250, 300], edge detection threshold is 0.1, and Hough accumulation threshold is 0.97.

Two variants were tested, the first with no preprocessing designated as Method (1) in Table 1 and the second with preprocessing using the Gauss filter (2) designated as Method (2) with sigma being set to 2. Both methods use the Sobel operator to find the edge pixels.

These two methods are a bit more imprecise compared to those using the 3-D accumulation matrix, but are significantly faster. Using the 2-D matrix reduces processing time to 1-2 seconds per image. IR images are relatively easily processed because of high contrast between the marker and background. That makes the edges easily detectable (Figure 2).

Table 1. Results of Hough transform with radius estimation

Method (1)				
Image type	Infrared		Visible spectrum	
	Center delta	Radius delta	Center delta	Radius delta
mean [px]	3.14	2.32	5.21	3.71
mean [mm]	0.106	0.078	0.100	0.071
std. dev. [px]	2.17	1.08	5.30	3.21
std. dev. [mm]	0.073	0.036	0.101	0.061
Method (2)				
Image type	Infrared		Visible spectrum	
	Center delta	Radius delta	Center delta	Radius delta
mean [px]	2.09	1.11	7.02	4.29
mean [mm]	0.070	0.037	0.134	0.082
std. dev. [px]	1.14	0.85	6.20	3.69
std. dev. [mm]	0.038	0.029	0.118	0.071

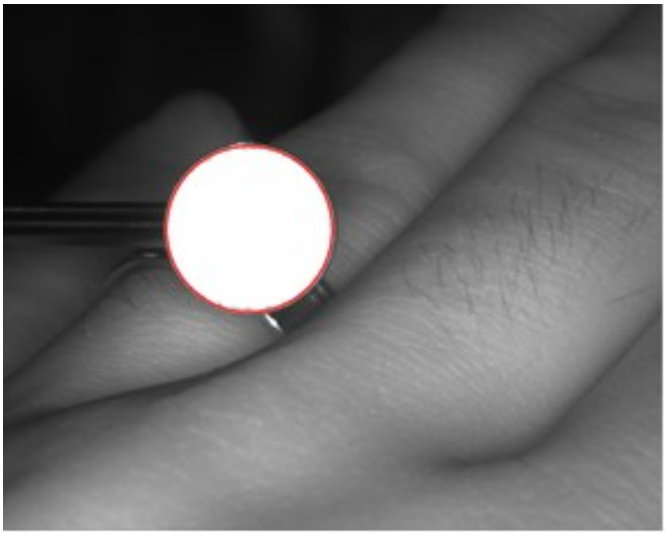


Figure 2. Circle detection by method (1) in an NIR image

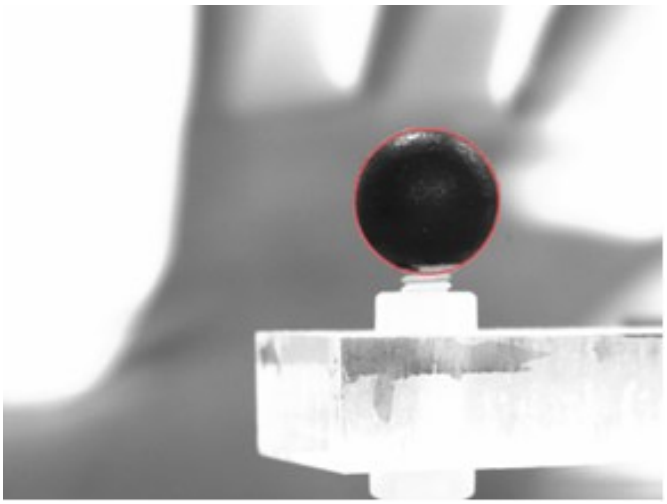


Figure 3. Circle detection by method (2) in an VS image

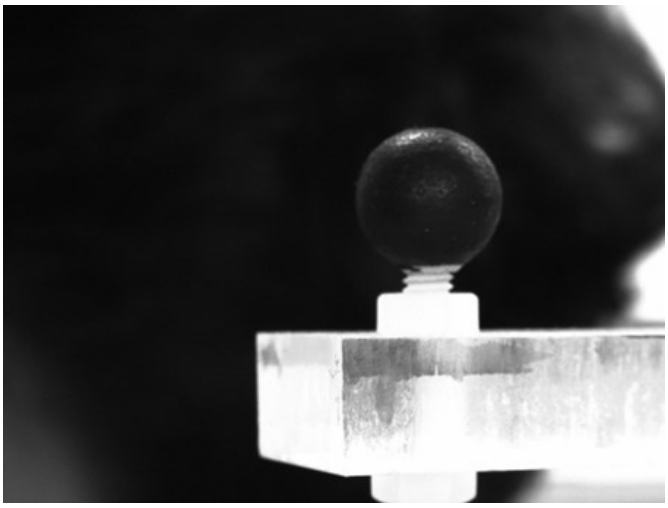


Figure 4. VS, low contrast between the marker and background (method (1))

Some VS images are also suitable for circle detection (Figure 3). However, these images have less distinct marker edges, leading to complete inability to detect circles in some of them (Figure 4) or imprecise detection (Figure 5). Additionally, Hough accumulation threshold has to be set quite low when processing VS images, due to the low number of edge pixels, which often leads to false detection (Figure 5).

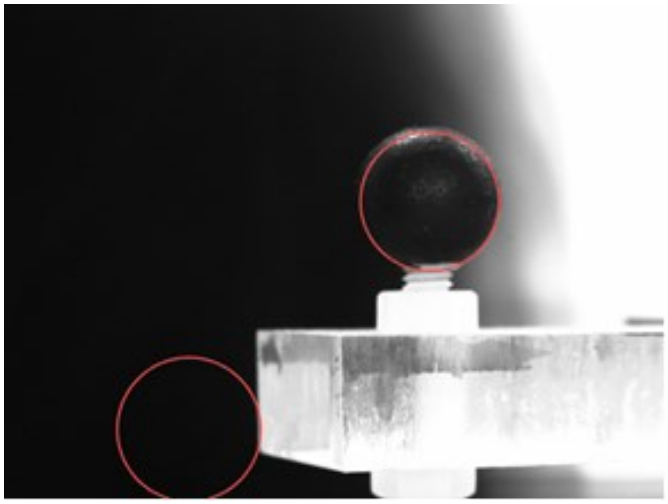


Figure 5. VS, detection by method (2)

B. Hough transform using a 3-D accumulation matrix

Four variants of this method are proposed and tested, depending on the edge detection operator used. Neighborhood suppression parameters are 51 pixels in all 3 directions. Method (3) uses the Sobel operator (the same as method (1)).

Method (4) uses the Sobel operator and Gauss filter preprocessing with the parameter sigma 2 (same as method (2)).

Method (5) utilizes Canny edge detection.

Method (6) uses the Laplacian of Gaussian

The IR images edge detection threshold is 0.2 and Gauss smoothing parameter sigma is 5.

The VS images edge detection threshold is 0.00007 and Gauss smoothing sigma is 8.

As stated previously, these methods take longer to execute, but the results are slightly better. Processing time for an image is on the order of 20-30 seconds. IR images give better results (Figure 6) than VS images (Figure 7) here as well, although the difference is reduced.

Methods (3) and (4) can be directly compared to methods (1) and (2) because of the same way of finding edges. Results show that calculation using a 3-D matrix is more precise than the one with a 2-D matrix. Another advantage of this approach is improved circle detection in low-contrast images. Figure 8 shows such an example where method (3) successfully detects the circle, while method (1) is unsuccessful.

Methods (5) and (6) utilize more advanced edge detection techniques, and are comparable precision-wise to methods (3) and (4). Canny and LoG additionally provide a smaller number of edge pixels, due to their precision and noise reduction, reducing processing time. However, not even these methods can guarantee absolute precision in low-contrast conditions (Figure 9).

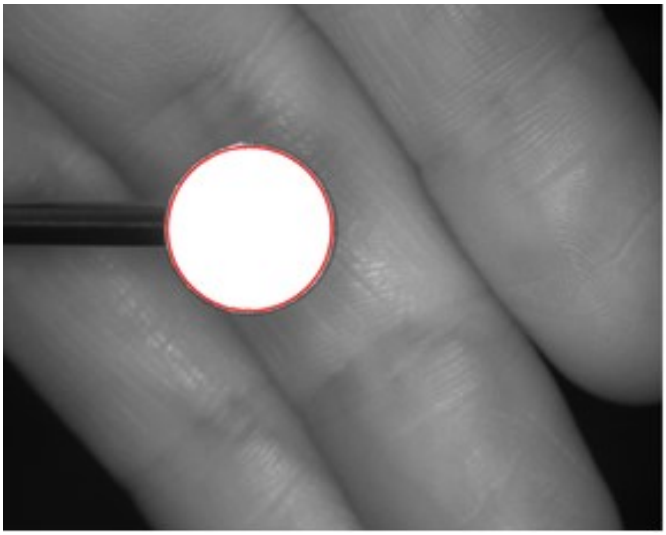


Figure 6. Circle detection using method (6) in an IR image

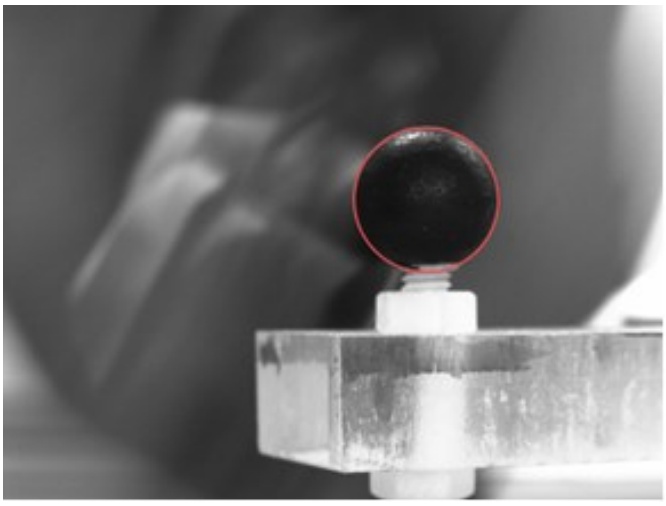


Figure 7. Circle detection using method (5) in a VS image

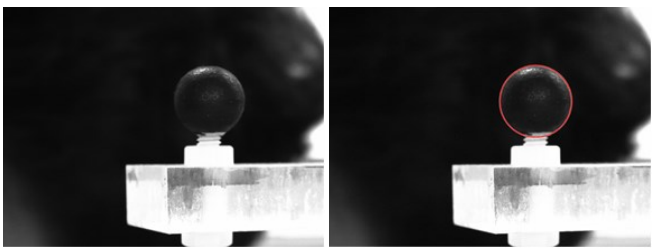


Figure 8. VS, method (1) left, method (3) right

Table 2. Results of Hough transform using a 3-D accumulation matrix

Method (3)				
Image type	Infrared		Visible spectrum	
	Center delta	Radius delta	Center delta	Radius delta
mean [px]	2.67	2.95	3.63	2.68
mean [mm]	0.090	0.099	0.069	0.051
std. dev. [px]	1.51	1.00	3.02	3.54
std. dev. [mm]	0.051	0.034	0.058	0.068
Method (4)				
Image type	Infrared		Visible spectrum	
	Center delta	Radius delta	Center delta	Radius delta
mean [px]	1.74	2.32	3.1	2.50
mean [mm]	0.059	0.078	0.075	0.048
std. dev. [px]	0.93	0.86	3.12	2.58
std. dev. [mm]	0.031	0.029	0.060	0.049
Method (5)				
Image type	Infrared		Visible spectrum	
	Center delta	Radius delta	Center delta	Radius delta
mean [px]	1.21	1.26	3.98	3.40
mean [mm]	0.041	0.042	0.076	0.065
std. dev. [px]	0.41	0.44	3.85	2.71
std. dev. [mm]	0.014	0.015	0.074	0.052
Method (6)				
Image type	Infrared		Visible spectrum	
	Center delta	Radius delta	Center delta	Radius delta
mean [px]	1.23	1.11	4.83	2.70
mean [mm]	0.041	0.037	0.092	0.052
std. dev. [px]	0.52	0.55	3.99	2.51
std. dev. [mm]	0.018	0.019	0.076	0.048

VI. CONCLUSION

Solutions proposed in this paper yielded very precise circle localization and radius estimation. However, the precision depends largely on the quality of the processed image, its background and lighting. This is especially evident in visible spectrum images. They give good results with sufficient contrast between the marker and background, but the precision diminishes along with the reduction in contrast. It is also evident that IR images yield better results than VS images, also due to higher contrast.

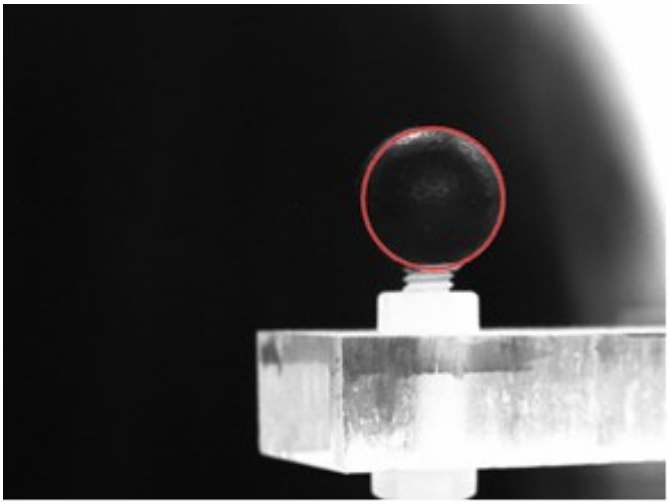


Figure 9. VS, method (5)

The Hough transform is an excellent and highly flexible tool for circle detection. In future work it is possible to research other ways of its implementation, in order to increase flexibility and robustness of circle detection.

The usage of Hough transform with a 3-D accumulation matrix yields the most precise results in both kinds of images, but this comes at the price of higher complexity. Since speed is sometimes equally as important as precision in practical circle detection applications, the fact has to be considered when choosing the optimal solution.

REFERENCES

[1] RONNA, http://www.ronna-eu.fsb.hr, 15. 6. 2016.
[2] Circle Hough Transform, https://en.wikipedia.org/wiki/Circle_ Hough_Transform, 15. 6. 2016
[3] Find circles using circular Hough transform – MATLAB imfindcircles, http://www.mathworks.com/help/images/ref/imfind circles.html, 15. 6. 2016
[4] Sobel, Irwin. History and Definition of the so-called „Sobel Operator“, 2015
[5] Sobel Operator, https://en.wikipedia.org /wiki/Sobel_opera tor, 15. 6. 2016
[6] Feature Detectors - Sobel Edge Detector, http://homepages.inf.ed.ac.uk/rbf/HIPR2/ sobel.htm, 15. 6. 2016
[7] Canny edge detector, https://en.wikipedia. org/wiki/Canny_edge _detector, 15. 6. 2016
[8] Spatial Filters – Laplacian/Laplacian of Gaussian, http://homepages.inf.ed.ac.uk/rbf/ HIPR2/log.htm, 15. 6. 2016
[9] Yuen, H.K., Princen, J., Illingworth, J., and Kittler, J. Comparative study of Hough transform methods for circle finding. Image and Vision Computing. Volume 8, Number 1, 1990, pp. 71–77.
[10] [Davies, E.R. Machine Vision: Theory, Algorithms, Practicalities. Chapter 10. 3rd Edition. Morgan Kauffman Publishers, 2005
[11] Atherton, T.J., Kerbyson, D.J. "Size invariant circle detection." Image and Vision Computing. Volume 17, Number 11, 1999, pp. 795-803.
[12] Midpoint Circle Algorithm, https://en.wiki pedia.org/wiki/Midpoint_circle_algorithm, 15. 6. 2016

Automated Computer Vision-Based Reading of Residential Meters

Karlo Koščević
Faculty of Electrical Engineering and Computing
University of Zagreb
Email: karlo.koscevic@fer.hr

Marko Subašić
Faculty of Electrical Engineering and Computing
University of Zagreb
Email: marko.subasic@fer.hr

***Abstract*—We present a solution for automated reading of various residential meters from photographs, using computer vision algorithm. The solution has several modules that are executed sequentially: meter type recognition, geometric transform of images, ROI extraction and OCR. Algorithm uses SURF and HOG features to detect and describe feature points used for device recognition and OCR. The final goal is to provide complete counter values and complete serial number of the meter. The solution allows for a limited amounts of poor imaging conditions, and usually fails in poor image conditions when even human observers would have difficulties reading meters. The solution has been implemented in MATLAB environment and its computer vision library. An initial image database has been collected for testing purposes. Test results are reported in the paper.**

I. Introduction

The goal of presented research is to enable automated reading of relevant data from images of residential meters, such as one shown in Figure 1. The solution is intended and applied to meters with mechanical counters, but the proposed method could easily be adapted to meters with electronic displays. The proposed solution should effectively turns ordinary residential meters into automated smart meters at no extra cost. The images of meters could be obtained using a smartphone, which could also be use for processing of the image. Ideally, a meter should be photographed at a right angle, with sufficient light, good contrast and no reflections, but the proposed method is capable of compensating for a limited amount of imperfections in images. However, certain light constraints regarding image acquisition must be obeyed.

The problem of automated extraction of visual information may seem trivial to an average person, because humans can easily detect all relevant areas of a meter, and also read out relevant information. A computer vision algorithm that does the same, must have several distinct steps. The steps are: detection and recognition of the specific meter in the image, geometric correction of the image, extraction of relevant regions of interest from the image, and actual reading of numerical data. The algorithm extracts two types of regions of interest, one or more counters and also an ID number, usually a serial number of the meter. Once the numerical data has been successfully read, it can easily be forwarded to the corresponding utility company, and used for personal record of the user.

Fig. 1: Example of photographed device

Device detection, recognition and reading of numerical characters is performed by template matching, so appropriate template database had to be prepared for the set of residential meters used in this research. If reading of an additional type of meter is required, new templates should be added to the database.

II. Methods used

Several image processing and analysis tools have been used in the proposed solution so each is briefly described in this section.

A. SURF

Speed Up Robust Features or shortly SURF [1] is both detector and descriptor of image features. It is partly based on SIFT descriptor [2], but its authors claim that SURF is much faster and robust. It can be used for tasks such as image registration, object recognition, classification or 3D reconstruction. For detection of significant feature points, SURF uses a blob detector based on Hessian matrix. Hessian matrix is calculated at each image pixel I at coordinates p q as shown in Eq. (1). Also, in improve performances, integral images are used. The determinant of Hessian matrix represents a measure of local change around points and points where this determinant form local maximum are chosen. Feature points are looked for at different scales with initial box filter size of 9x9 (corresponding to Gaussian derivatives with σ=1.2).

Invariance to rotation and scaling are also useful qualities of SURF.

As for the local neighborhood descriptor, it consists of fixing a reproducible orientation based around the interest point and extraction of SURF descriptor from a square region aligned to the assigned orientation. The orientation is calculated in order to archive rotational invariance of feature vector. The dominant orientation is estimated by calculating the sum of all Haar wavelet responses weighted by a Gaussian function within the sliding window of size $\pi/3$. Region around the point is described by extraction of square region centered on the point and oriented along the selected orientation. For each sub-region of extracted square (that is split into 4x4 square sub-regions) Haar wavelet response is extracted and weighted with Gaussian. Gaussian filter affects the features in the manner that makes them more robust to the noise, translation and deformations.

$$H(p,q) = \begin{bmatrix} I_{xx}(p,q) & I_{xy}(p,q) \\ I_{xy}(p,q) & I_{yy}(p,q) \end{bmatrix} \quad (1)$$

B. HOG

The Histogram of Oriented Gradients or shortly HOG is a image feature descriptor [3]. It is widely used for purposes of object detection. As its name points, HOG is based on computation of gradient orientations. Image is divided in smaller regions called blocks and each block is also divided in smaller regions called cells. Then in each cell, gradient amplitude and orientation are calculated, and cell is assigned to appropriate orientation bin. Once that is done for all cells in one block, all this data from cells is normalized and concatenated into one vector that represents a descriptor of single block. Afterwards, block descriptors are concatenated and they form a single vector that is a descriptor for the whole image. Because of this principle, HOG is also called a global descriptor and the size of its vector depends on number of blocks and cells, as well as number of orientation bins. It is important note that HOG is not invariant to image rotation.

C. Perspective projection and Homography

Perspective projection is a geometric transform of points form 3D scene to the 2D plane. This happens regularly during 2D image acquisition when projection is effectively done at an image sensor plane. Basic concept of this projection is presented in Figure 2. In our case, we will be comparing two projections (photographs) of the same object projected at different planes. A perspective transform or homography [4] will be required to transform input image to the projection plane of the template image. This will effectively align meter screens of input images to meter screens of template images. Here we presume that meter screens are entirely flat, which is not entirely true. To estimate homography parameters, at least four corresponding point pairs are required, but more can be used to obtain a more reliable estimate.

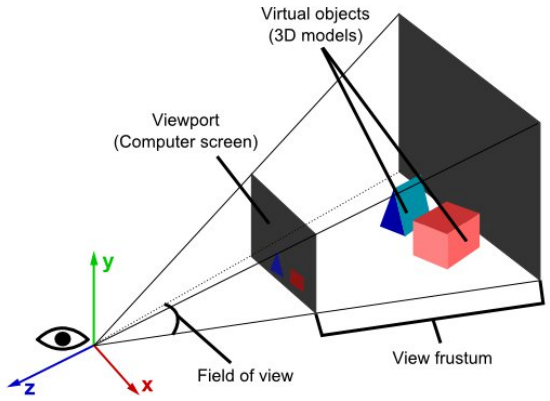


Fig. 2: Example of perspective projection

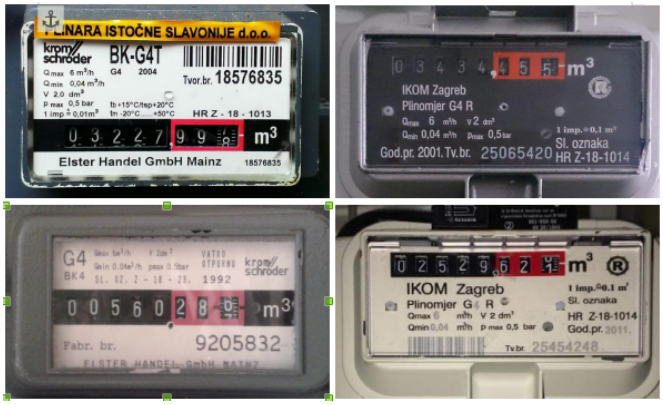


Fig. 3: Samples of residential meters

D. K-d tree

K-d tree is a special case of binary space partitioning trees [5]. K-d trees are arranging points in k-dimensional space and are useful data structure for multidimensional search such as range searches and nearest neighbor searches. The algorithm works by recursively partitioning the set of training instances based on a median value of a chosen attribute. Attributes are alternately selected coordinate axes of training instances. When we get a new data instance, we find the matching leaf of the K-d tree (alternately comparing its data with nodes' data), and compare the instance to all the training point in that leaf. We use k-d tree to increase speed of feature point matching process (searching of two nearest feature points to the one that is observed).

III. Input data and template database

We have obtained 79 images of residential meters for electricity and gas (several samples showed in Figure 3). Though we had just two meter categories, differences in positions of counters and serial numbers required separate templates with different ROI positions for the same meter type. Differences in screen color, digit color and some larger details, also required separate templates, as otherwise, chances of false

(a) Template image before homography. Green circle - initial corners; white dashed line - initial x and y coordinates of initial corners; solid red line - new x and y coordinates of corners Each corner is translated to its closest intersection of two red lines

(b) Template image after homography

Fig. 4: Ilustration of homography of meter template

meter recognition are quite high. Hence, effectively we had 9 meter classes, and we had prepared 9 templates.

A template consists of cropped image of the meter screen, with screen rectified (see example in Figure 4). In order to perform cropping and rectification a homography has to be performed. Here we presume that meter screen is completely flat and all counter digits lie in the same plane. This is not entirely true for common residential meters, but distortions induced by this step are negligible in our expected usage scenario, where template is produced using the best obtained image for given meter type (least amount of blurring, noise, no saturation, camera positioned at the right angle). To estimate appropriate homography parameters, four corner of template screens have been manually annotated. Together with desired corner positions of templates, they give all homography parameters. After applying homography, all meter screen edges should be at the right angle as ilustarated in Figure 4.

For the proposed method, regions of interest (ROI) information has to be stored with each template. Coordinates of two opposite corners (of rectangular ROIs) in template coordinate space are stored for each counter digit and for serial number. The coordinates are annotated manually in the rectified template image.

SURF feature points are calculated for each template image. Obtained feature points are filtered to remove feature points outside borders of the meter screen. Feature points are also removed if they are positioned in or very near to the variable

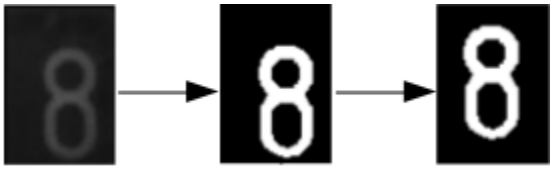


Fig. 5: Preparation of sample image of digit

regions of meter screens. Such variable regions are counters and serial numbers and maybe some minor variable details in meter screens. The idea is to obtain significant feature points that are highly likely to appear with all meters of the given type.

Also, a single binary template image has been prepared for each digit. We haven't noticed significant differences in number fonts of counters and serial numbers between different meter classes. Hence, just ten number templates (one for each numerical character) has been used in this research. Digit templates have also been chosen with the image quality in mind, and each digit template is of the same size 60x84 pixels. Binarization is performed by thresholding, using average gray value of the template as a threshold. Binary representation of the digit is placed centrally within the template image (see example in Figure 5). Binary templates will be compared with binarized digits from the input image.

Our testing database consisted of 3 to 35 images per each meter class. Images were obtained by smartphone cameras in various conditions, and image dimensions were 720 to 3648 x 776 to 3648 pixels. Corner points of all meter screens have been manually annotated for validation purposes.

IV. Overview of the presented method

The presented method starts with recognition of the residential meter in an image. SURF is used for that purpose which makes recognition process insensitive to translation, scaling, image rotation, and a limited amount of in-plane rotation. This enables certain degree of flexibility regarding the position of the camera relative to the screen of the meter. Exaggerations in geometric transformations usually makes meters unreadable even for human readers, so our solution focuses only on images with moderate amounts of geometric transforms. We also assume that whole screen of a meter is visible in an image, and that the screen spans most of at least one image dimension. Such requirements should easily be satisfied if a person is willingly photographing a residential meter. Certain cases when these conditions are not satisfied, could even be detected automatically in real-time to warn the user. Expected image conditions are also reducing the problem of meter detection, as we expect that just one meter screen is present in the image.

The recognition is performed by matching of the SURF points detected in an image with SURF points of all templates. The right meter template should have the largest number of matched points. After successful recognition, the pairs of matched SURF points are used to estimate geometric

transform needed to align the meter in the image with the meter in the template. Once the image is aligned with the template, ROI positions of the template can be used to extract ROIs from the image.

The final step is reading the numerical data for extracted ROIs containing counters and serial number. The appearance of these two numbers usually differ, so slightly different procedures have to be applied. Each counter digit is usually clearly physically separated from others, and all digits can use same size bounding box. This makes counter digit extraction an easy task. Serial numbers are printed, and usually characters don't have fixed widths. Hence, prior to character recognition, serial number has to be segmented. Numerical characters are finally recognized by comparing them to character templates using HOG features.

Significant error in any of the above steps, increases the chance of wrong reading results or inability to properly read data from the meter. Failure to detect an expected character in the ROI can be detected, and the user warned.

V. Recognition of residential meters

First step of meter recognition is SURF feature point matching. Each significant SURF feature point in an input image is matched against all SURF feature points of all templates. Matching is done by searching for the nearest neighbor in k-d tree which is made of feature points extracted from all templates. For each feature point of an input image, two nearest neighbors among template feature points in the feature space are obtained. The Euclidean distances of two nearest neighbor feature points to the input feature point have to be sufficiently different to avoid confusion and to guaranty successful match. If the difference is sufficient, the input feature point can be matched to its closest neighbor. If the difference is below a certain threshold, the input feature point is not matched. Finally, the match is accepted if the distance is below a threshold, otherwise the match is discarded. Once all feature points from an input image have been processed, each will have one or zero matches. An example of matched points for one template can be seen in Figure 6.

We expect that a meter screen in an input image can be adequately aligned with the screen in an template image by homography. All coordinate differences of matched feature points between input image and an template, must correspond to the same transform parameters. Hence, the matched pairs are used to estimate parameters of the transform. If coordinate difference of any matched pair does not conform to the estimated geometric transform, the match is discarded. Since each matched pair presents a vote for a corresponding template, this step reduces the probability of that template match.

An input image is matched with a template that has the largest set of remaining matching feature points.

VI. ROI extraction

Once the meter has been successfully recognized, rectangular ROIs containing relevant numerical data have to be extracted. Each template has a set of ROI coordinates in its coordinate system, so meter screen in the input image has to be aligned with meter screen in the template image. We presume that a meter screen together with its whole content is flat so a homography transform has to be performed to align two meter screens. An input image will not violate the assumption if it has been recorded according to the acquisition requirements of the proposed method explained in section III.

Parameters of the geometric transform have to be estimated using a set of matching points in two images. However, matching pairs of significant points have already been obtained in the previous step. Furthermore, parameters of the geometric transform have also been obtained for the matching refinement. Hence all that is required in this step is to apply the geometric transform to the input image. One requirement is that there is a sufficient number of matched pairs. Four matched pairs is the minimum, but more pairs is preferred. Low number of matched pairs is usually indicator of false match or poor imaging conditions. An example of the geometric transform is shown in Figure 7. One can observe that in the transformed image, meter screen is in the upright position with all four edges at the right angle.

To evaluate the quality of geometric transform, corners of each meter screen have been manually annotated. After the input image transform, new coordinates of screen corners can be compared to the corner positions of the corresponding template. Results section contains evaluation results for geometric transformations.

Once meter screens are aligned, ROI coordinates of the template are used to extract each counter digit and a serial number.

VII. OCR

Digits are recognized using HOG descriptors. We have chosen HOG for this purpose for the reason that the edges of objects that are recognized can be clearly separated and the assumption is that objects can differ based on finite number of cells and orientation bins. Rectangular ROI containing a digit is extracted in the previous step, so the recognition is performed in the ROI only. Each digit is preprocessed in the same way as digit templates to obtain a centered binary representation of the digit. The binary image of the digit must be scaled to match the size of digit templates. HOG vector is calculated for the scaled digit image, and the scaling ensures that HOG vectors of input image and templates have the same size. Images are divided into cells so that each cell has both width and height equal to a quarter of the image width and height. The cell histogram is 1-by-$NumBins$ vector where $NumBins$ is set to a default size of 9 orientation histogram bins. Blocks consists of 4 cells with 1-by-1 cell overlap between blocks. Total number of blocks is 12 and total size of calculated HOG vector is 1-by-216. A digit is recognized as a number whose template has the nearest HOG vector. There are only ten template vectors to compare with moderate number of elements, so no optimized comparison algorithms are required (like k-d tree in section V).

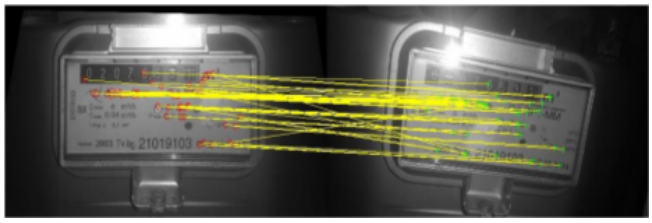


Fig. 6: Visualization of matched features. On the left side is the sample device to which observed one (one on the right) was matched.

(a) Meter before the projection

(b) Meter after the projection

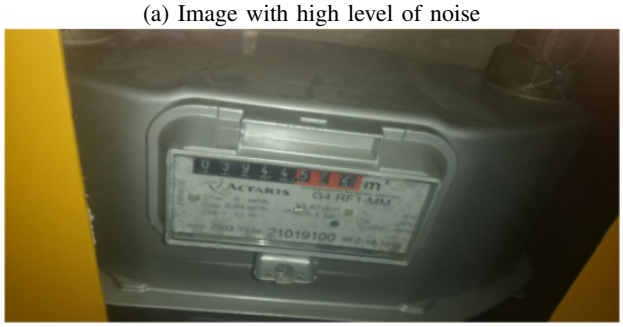
Fig. 7: Visualization of the projected meter

This procedure can be applied directly to counter digits, where each digit has its own bounding box, and bounding boxes are clearly separated. Due to this a separate ROI for each digit is stored with each template. Digits in serial numbers have varying widths (e.g. digit 1 is narrower than other digits) and are not equally spaced. A segmentation step is required to separate digits after the entire serial number has been successfully extracted. The serial number ROI is binarized using its average gray value as a threshold, and sufficiently large connected components are extracted as separate digits. Each digit is than represented as one image and dimensions of the image are equal to width and height of connected component which is assumed to be a digit. Images of digits are once more binarized and HOG vector is calculated for each digit and compared to HOG vectors of digit templates.

TABLE I: Test results of reading residential meters

Device type	No. of images	Correct recognition	Correct geom. transf.	Correct reading of counter (%)	Correct reading of serial No. (%)
1	7	7	7	89.8	94.64
2	35	35	33	96.33	85.71
3	3	3	3	81.95	75
4	5	3	1	100	-
5	20	19	16	86.785	90.15
6	1	1	1	100	88
7	3	3	3	86	83.33
8	5	4	2	100	87.5
TOTAL	79	75	66	92.61	86.19

(a) Image with high level of noise



(b) Device captured from bad angle

Fig. 8: Example of bad images for process of reading residential meters

VIII. Results

As it is explained in section V, to match two features, two thresholds have to be applied. The Euclidean distances of the first nearest neighbor feature point to the input feature point has to be at least 22.5% smaller than the distance of the second nearest neighbor feature point. If the input feature has been matched to its closest neighbor, the match is accepted if their Eucledian distance is below value of 0.25, otherwise the match is discarded.

Results for each algorithm step and each device class are shown in Table 1. Images which were producing an error in one of the steps of algorithm were discarded after that step. Columns annotated as *Correct reading of counter* and *Correct reading of serial number* are representing percentage of meters on which each digit of counter or serial number was correctly read.

Geometric transform of an image was declared unsuccessful when number of matching pairs between observed image and a template image was too small (5 or less matches) or estimated transform was very unlikely. If the number of matching pairs was sufficient, the recognition was always successful. As for reading the serial number, wrong recognitions were mostly due to errors in binarization and the detection of connected components. Usually, the threshold value calculated as the average intensity of the ROI would produce inadequate connected components. This would usually be caused by poor imaging conditions. Examples of bad images of device are shown in Figure 8. Subfigure 8a has excessive amount of shadows and background reflection, as well as the reflection of light which is the major problem in most cases. Although, SURF is invariant to rotation, angle of device in subfigure 8b is too large and after homography, some parts of digits can not be correctly determined. Clearly, our approximate assumption that meter screens are flat, does not hold in this case.

IX. Conclusion

SURF and HOG have proved to be appropriate tools for recognition of residential meters and reading of their numerical data. Although, the results for reading digits are slightly lower than ones for recognition of device type, complete algorithm has overall very good results, and it can be applied on many different types of devices displaying numerical data. All that needs to be done to read devices of a new type is to prepare one or several templates that represent that device type. Algorithm certainly has room for future improvements, especially part of the algorithm that segments the serial number. Most errors occurred in this procedure, while the other parts had success rate around 90% or greater (recognition of device type had a success rate of 95%). After examining the results it can be seen that the most common errors are caused due to the poor quality images, e.g. images with high level of reflected background light, images with shadows or blurred images. Therefore, images should be obtained carefully to assure sufficient image quality. For this reason, our future plans include automatic estimation of image quality.

References

[1] Herbert Bay, Tinne Tuytelaars, and Luc Van Gool. *SURF: Speeded Up Robust Features*, pages 404–417. Springer Berlin Heidelberg, Berlin, Heidelberg, 2006.

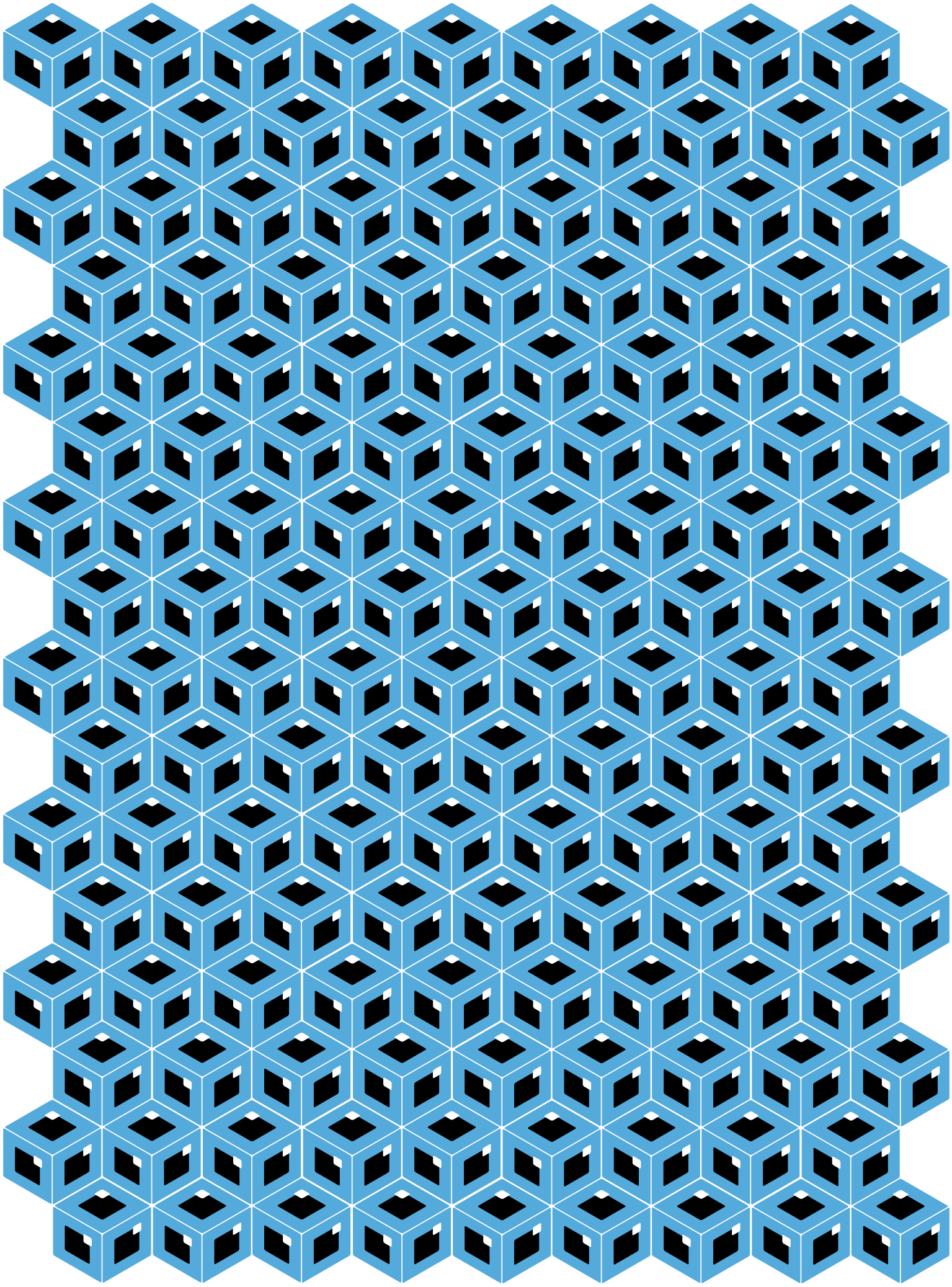
[2] D. G. Lowe. Object recognition from local scale-invariant features. In *Computer Vision, 1999. The Proceedings of the Seventh IEEE International Conference on*, volume 2, pages 1150–1157 vol.2, 1999.

[3] Navneet Dalal and Bill Triggs. Histograms of oriented gradients for human detection. In *Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05) - Volume 1 - Volume 01*, CVPR '05, pages 886–893, Washington, DC, USA, 2005. IEEE Computer Society.

[4] R. Baer. *Linear Algebra and Projective Geometry*. Dover Books on Mathematics. Dover Publications, 2005.

[5] Jon Louis Bentley. Multidimensional binary search trees used for associative searching. *Commun. ACM*, 18(9):509–517, September 1975.

Author Index



Proceedings of the Croatian Computer Vision Workshop CCVW 2016 ISSN 1849-1227
Publisher: University of Zagreb Faculty of Electrical Engineering and Computing, Unska 3, HR-10000 Zagreb, Croatia
http://www.fer.unizg.hr/crv/proceedings