Vision-based robotic system for picking and inspection of small automotive components

Oleksandr Semeniuta[1], *Student Member, IEEE*, Sebastian Dransfeld[2], and Petter Falkman[3], *Member, IEEE*

***Abstract*— The use of vision systems for industrial robot guidance and quality control becomes much harder when the manufactured products and their components are small and possess reflective surface. To assure an effective automated visual inspection of such components, novel solutions are required, able to perform more advanced image analysis and tackle noise and uncertainty. This paper proposes a concept of multi-camera/multi-pose inspection station for star washers inspection, and presents the first results of a functional prototype implementation of it in a robotic cell. The processes of vision-guided part picking from a flexible feeder and close-range inspection in a dedicated rig are described. Solutions for the vision-based tasks of parts identification, machine learning-based classification, circular objects image analysis and star washer teeth segmentation are presented, and further directions are outlined.**

I. Introduction

In today's competitive environment, quality of manufactured products becomes an important advantage. In many cases, particularly in the automotive industry, one deals with safety critical components, quality of which is a stringent requirement that directly reduces potential risks.

Many of the automotive parts are small in size, and yet they constitute an important role and place high requirement on their production systems. Moreover, the smaller a component, the more difficult it is to be manipulated and inspected for quality. In addition, reflectivity, which is an inherent property of metallic components, is a well-known challenge to implementation of machine vision-based inspection systems. Though the objects' surface has a large impact on the vision system, there is little correlation between surface quality and product quality.

Cycle time and volume requirements are important factors in the automotive industry. The cycle time sets high demand on how fast the quality control has to be, and how it can be performed. The volume results in the large variation set, and makes it impossible for humans to do the inspection manually, or be able to understand the complete data set without use of computers and data analysis.

An example small yet important automotive components are air brake couplings, used to connect tubes in the breaking system of a vehicle. KArtridge™ is product family of couplings manufactured by Kongsberg Automotive AS. They are multi-material products, comprised from composite, rubber, and metallic parts. Of a big importance in the KArtridge™ couplings is a star washer, a general-purpose metallic component that secures the grip function between a coupling and its housing. Depending on the product size, the outer diameters of star washers range from 15 to 26.5 mm. In addition to the small dimension of the star washers, their teeth, which perform the most important function, are much smaller, ranging in the width from 1.4 to 2.11 mm.

An example of a good and a defective star washer is presented on Figure 1.

In order to assure zero-defect manufacturing of the air brake coupling, 100% quality inspection of star washers and other components has to be done. Challenges to machine vision-based inspection exist due to the inherent reflectivity of the star washers' metallic surface and batch-to-batch variability in color. In addition, the small size of the components makes inspection sensitive to cameras' and lighting setup.

To tackle difficulties of vision-based measurement of small parts, the sophistication of the applied vision system can be increased. Particularly, instead of analysis of one image in a fixed pose, several poses can be registered. For instance, a star washer can be imaged from the top, resulting in a frontal-parallel view, and from the side, so that a more detailed geometry of the teeth could be captured. For each pose of interest, a different lighting condition can be created to highlight the required visual features.

Clearly, such systems would require a complex mechanical setup. It may resemble a dedicated assembly machine, used for high-volume production, but, instead of performing assembly tasks, the resulting system would be designed to manipulate parts in order to position them in the required measurement poses.

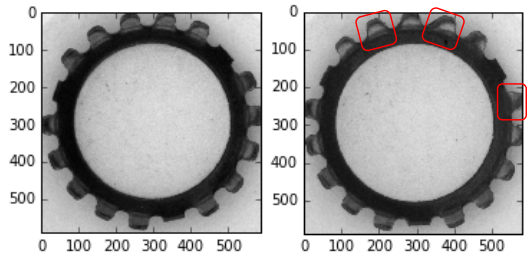

Fig. 1. An example a good star washer (left) and the one with various teeth geometry defects (right)

*This work was supported by the Norwegian Research Council

[1]Oleksandr Semeniuta is with Faculty of Technology, Economy and Management, NTNU Norwegian University of Science and Technology, Gjøvik, Norway oleksandr.semeniuta@ntnu.no

[2]Sebastian Dransfeld is with Department of Production Technology, SINTEF Raufoss Manufacturing AS, Raufoss, Norway sebastian.dransfeld@sintef.no

[3]Petter Falkman is with Department of Signals and Systems, Chalmers University of Technology, Göteborg, Sweden petter.falkman@chalmers.se

Because of inherent mechanical complexity of a dedicated solution, it is rather challenging to recreate it in the laboratory environment for the concept experimentation. On the other hand, the task of multi-camera/multi-pose imaging can be implemented with an industrial robot. In such a way, one can obtain a compact reconfigurable solution for part inspection, which can be continually developed and serve as a playground for experimentation with various machine vision solutions.

This paper presents the first results of the abovementioned laboratory prototype implementation. Firstly, the architecture of the robotic cell and conceptualization of the desired workflow is outlined. Then the the results of two machine vision tasks are presented, namely (1) identification of parts on the feeder, and (2) close-range inspection. The two tasks serve two principally different roles, namely robot guidance and dimensional inspection. Hence, they are performed at different levels of details in terms of image resolution and the measured entities. However, a common approach for circular objects analysis is used in both cases.

Because of the components' size, it is difficult to directly differentiate between two possible orientations when one is lying on the feeder. To tackle this problem, machine learning (ML) is applied, and several classifier are evaluated.

Fig. 2. An example of incorrect (left) and correct (right) orientations of a star washer

The paper is organized as follows. Section II overviews the related work. Section III overviews the details of the cell setup. Section IV then describes the proposed solutions. Section V presents the obtained results. Section VI concludes the paper and sets the direction for the further work.

II. Related work

Vision-based robotic picking highly depends on the kind of the manipulated object, which makes it very application-specific to both the sensing and the manipulation tasks. In this review, the focus is put on previous work that touched upon aspects that are similar to those of star washers' picking and inspection, namely random placement of fed objects, small size, reflectivity, and circular shape.

Fully-flexible assembly system concept is presented in [1], [2]. The concept is based on simplification of mechanical structure of an assembly system through a heavier usage of vision systems for parts identification and measurements. A typical solution of this kind comprises a flexible feeder handling a range of different components, a robotic manipulator, and an assembly station. Though this solution may be unsuitable for high-speed production, its flexibility can allow for such application as flexible small-batch production, 100% inspection of parts in a measurement station, and flexible feeding to high-speed production lines.

A solution for pose estimation of randomly placed highly reflective industrial objects is presented in [3], which is based on a multi-light imaging system combined with data-driven pose estimation. Patch voting is used to estimate poses, and efficient algorithm for database search is implemented using random ferns.

Dimensional vision-based measurement system for small eyeglasses components is presented in [4]. The components are identified on a backlighted plane and then handled by a robot to the measurement subsystem, where detailed vision-based estimation of offset, thickness, holes and thread measures are performed. Working cycle of inspection is improved by applying simultaneous measurement in two areas and part handling.

A system for vision-based measurement of small solder balls on a printed circuit board is presented in [5]. The solution is based on analysis of connected components, obtained after robust thresholding and seed-fill noise removal.

A general-purpose method of circular object inspection, based on building and analyzing radial intensity histograms, is presented in [6].

III. Prototype setup description

The implementation of a functional prototype for star washer picking and inspection is being performed in the robotic laboratory of SINTEF Raufoss Manufacturing AS (Raufoss, Norway).

The architecture of the cell is presented on Figure 3. For part picking and manipulation, an articulated robot Adept Viper s850 is used. Interfacing with the robot controller and robot programming is done from a separate desktop environment, Adept Desktop, of version 4.2.2.8, which is installed on a dedicated PC. The programming is done in the proprietary V+ language.

Part feeding is done using an Anyfeed SX240 flexible feeder. It is designed for random feeding of the parts and usage in a conjunction with a vision system for pose estimation. The feeder is able to communicate with external systems via RS-232 communication interface. The picking surface of the feeder is backlighted, which simplifies further image processing and parts identification.

As the part identification camera, a monochome Allied Vision Prosilica GC1020 camera (1024x768 in resolution) with PENTAX C1614-M lens (8 mm) is used. The inspection rig is comprised of a stereo system with two monochrome Allied Vision Prosilica GC1350 cameras (1360x1024 in resolution) with Fujinon HF35HA-1B 35 mm lens. In the current work, only one camera has been applied.

The stack of software components used to implement the proposed solutions is described in more details further in section V.

IV. Proposed solution

A. Workflow description

The workflow of the considered process is presented as a sequence of operations diagram in Figure 4. The operations

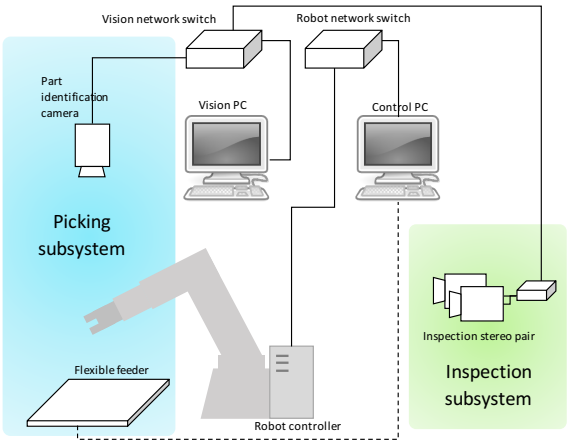

Fig. 3. Communication architecture of the demonstrator cell

tackled in this paper are outlined in bold. A more detailed description of the workflow is described below.

The feeder used in the setup can be controlled via the RS-232 interface to perform such actions as feeding more parts to the feeding surface, flipping the parts on the feeding surface, and moving the parts linearly. This is achieved by vibrations of different forms and intensities.

A mono vision system on top of the feeder takes an image of the feeding surface to find the positions of the components available to pick. Among the all washers on the surface (set W) there is a subset of those in correct orientation ($W_{correct} \subseteq W$). If there is no components in correct orientation ($W_{correct} = \emptyset$), the feeder flips, and the abovementioned vision identification procedure needs to be executed once more. If there is no components at all, the feeder feeds the new ones from its upper part.

This paper excludes the feeding process and leaves its description for the further work.

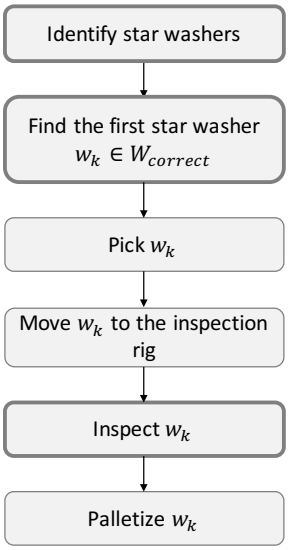

Fig. 4. Workflow diagram

Since there is only one camera used for part identification, the coordinates of the washers are estimated as 2D points, expressed in pixels, on the image plane $ImagePlane \subset \mathbb{R}^2$. Having the part identification camera calibrated using the chessboard-based procedure [7], [8], [9], the known intrinsic parameters of the camera and one known pose of a chessboard lying of the feeder can be used to perform mapping from the image plane to the feeder plane as a homography transformation:

$$H_{feeder} : ImagePlane \rightarrow FeederPlane \quad (1)$$

Because the points on $FeederPlane$ are expressed in millimeters, it is further possible to attach a coordinate frame to its origin and calibrate its pose expressed in the robot's coordinate system. This part of the work, together with the picking process itself, are not considered in this paper and will be described in the following publications.

After the identification, the robot picks the first part in the correct orientation and moves to the inspection rig. Here, a pert is inspected using a dedicated camera system in the close range (Figure 5), and further palletized.

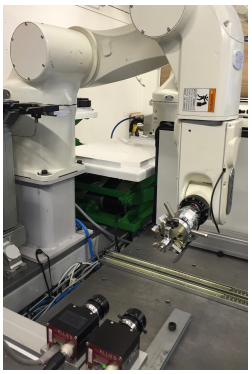

Fig. 5. Robot in the inspection pose

B. Star washers identification

Once an image of the feeder $I_{original}$ is acquired, the components are identified as follows:

1) $I_{original}$ is thresholded using the Otsu's method [10], i.e. the optimal threshold value is found which maximized inter-class variance between the classes of white and black pixels; I_t is obtained.
2) I_t is used to separate the connected components. As a result, a label image I_{label} is obtained, together with the following descriptor of each connected component: $(x_{left}, y_{top}, w, h, a, x_{center}, y_{center})$, corresponding to the left column coordinate, top row coordinate, width, height, area, and center coordinates respectively. All descriptors are stacked-up into a data frame $\mathbf{D} \in \mathbb{R}^{n_{labels} \times 7}$, with each row describing the particular connected component.
3) Data frame $\mathbf{D}$ is filtered to describe only those connected components that correspond to the star washers; $\mathbf{D_w}$ is thus obtained.

Because star washers are circular objects, having $\mathbf{D_w}$, it is easy to extract sub-images from $I_{original}$ that contain each individual component in a form of a bounding box with top left corner at (x_{left}, y_{top}), width w, and height h.

C. Circular object analysis with polar transformation

A different view on a circular object can be obtained by transforming it from the Cartesian coordinate system to the polar coordinate system. The latter represents an arbitrary point as tuple (r,θ), where r is a length of a ray from the origin point, and θ is an angle of the ray. With the known Cartesian coordinates of the origin of the polar coordinate system $\mathbf{x_0} = [x_0, y_0]^T$ and the given pair (r,θ), the corresponding Cartesian coordinates $\mathbf{x_1} = [x_1, y_1]^T$ can be computed as follows:

$$\begin{aligned} x_1 &= x_0 + rcos(\theta) \\ y_1 &= y_0 + rsin(\theta) \end{aligned} \tag{2}$$

As in the case of star washers, the estimated coordinates of the center of each washer is already known as a result of connected component analysis. Given that $r = h/2$, an image of a circular star washer can be unwrapped into a polar image. Specifically, for each angle $\theta = i\Delta\theta$ (where $\Delta\theta = 2\pi/n_{angles}$ is a small angular increment), it is possible to obtain a profile along the segment from (x_0, y_0) to (x_1, y_1). Because of the biggest interest is part of the image with the actual metallic component, and not the inner empty area, a part of a radial profile of length $l_{ignored}$ can be ignored, thus taking into account only the segment from $\mathbf{x_{start}}$ to $\mathbf{x_1}$, where $\mathbf{x_{start}} = [x_{start}, y_{start}]^T$ is computed as follows:

$$\begin{aligned} x_{start} &= x_0 + l_{ignored}cos(\theta) \\ y_{start} &= y_0 + l_{ignored}sin(\theta) \end{aligned} \tag{3}$$

The length of the resulting profile can be specified in a fixed number of pixels n_{pixels}. Each radial profile of this kind correspond to a row of a matrix $\mathbf{P} \in \mathbb{R}^{n_{angles} \times n_{pixels}}$. $\mathbf{P}$ (Figure 6), which constitutes a polar image, can be further used for edge detection and other image analysis operations. In addition, each row and column of $\mathbf{P}$ can be used as-is: $\mathbf{P}(i,:)$ represents a radial line profile for angle θ_i, and $\mathbf{P}(:,j)$ represents a profile around a circle of radius r_j:

$$\begin{aligned} \theta_i &= i\frac{2\pi}{n_{angles}} \\ r_j &= l_{ignored} + j\frac{|\mathbf{x_{start}}, \mathbf{x_1}|}{n_{pixels}} \end{aligned} \tag{4}$$

It is important to stress the following aspect. Traditionally the coordinate system of an image is positioned in the top left corner, with x-axis pointing to the right and y-axis pointing down, thus making $[x,y]^T$ the coordinates of xth column and yth row. Positive rotation in this case would appear in clockwise direction on the image (as opposed to the traditional counterclockwise direction). To eliminate this source of confusion, it is handy to flip the coordinate frame, so that x-axis represents rows and y-axis represents columns. This notations would also be more consistent with traditional matrix indexing, and since images are typically handled as matrices, the flipped coordinate system would result in a simpler indexing when interacting with existing matrix-based procedures.

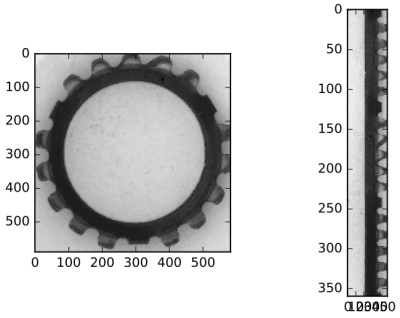

Fig. 6. An example of a star washer subimage (left) and the corresponding $\mathbf{P}$ matrix (right)

D. Classification of star washers' orientation on a feeder

A challenging task from the computer vision point of view is determining orientation of each individual star washer on the feeder in order to guide the robot to pick only those belonging to $W_{correct}$. Let each star washer w, seen on the image of the feeder surface, be characterized with its center coordinate $\mathbf{x_w} \in ImagePlane$ and a binary orientation label $o_w \in \{0,1\}$, such that $w \in W_{correct}$ if $o_w = 1$:

$$w = (\mathbf{x_w}, o_w) \tag{5}$$

Determining o_w is not easy, since there are no evident visual clues on a frontal image of a feeder, which would help to determine whether a washer is correctly oriented or not. In addition, because of the reflective surface of the components, they have different appearance on different regions of the feeder. Since the direct measurement is not possible in the given situation, a classification procedure based on machine learning is proposed.

Various supervised learning algorithms are focused on learning an unknown function $f : X \to Y$ from a set of training samples $\{\mathbf{x_i}, y_i\}$, where $\mathbf{x_i} \in X, y_i \in Y$, and $y_i = f(\mathbf{x_i})$. In case of classification problem, $Y = \{0,1\}$ or $Y = \{0,1,2,\ldots,K-1\}$, where K is the number of classes. The dimensionality of feature vectors $\mathbf{x_i}$ is typically large.

In order to classify star washers, i.e. predict the value of o_w for a washer w, the following solution is proposed. The feeder surface is manually filled in by a large number of star washers (e.g. 120), depending on how many of them fit, all in the correct orientation. An image of the feeder is taken and saved as I_1. The same is done when all washers are positioned incorrectly, giving the image I_0. Having the implementation of star washers identification and Cartesian-to-polar transformation, in both I_1 and I_0 the possible sub-images of the washers are extracted and transformed into the corresponding $\mathbf{P}$-matrices. Regardless the varying sizes of sub-images, the $\mathbf{P}$-matrices will all be of the same size, i.e. $n_{angles} \times n_{pixels}$. Having each $\mathbf{P}$-matrix flattened, it becomes a feature vector of the unified size.

Assuming that one has obtained m_0 and m_1 feature vectors from images I_0 and I_1 respectively, each of the sets (X_0, X_1) have to be partitioned in the training and testing subsets:

$$\begin{aligned} X_0 &= X_0^{train} \cup X_0^{test} \\ X_1 &= X_1^{train} \cup X_1^{test} \end{aligned} \tag{6}$$

Further, feature vectors in set X_0^{train}, labeled with $y = 0$, and in set X_1^{train}, labeled with $y = 1$, can be used for training the chosen ML classifier. The correctness of various classifiers can be benchmarked using the sets X_0^{test} and X_1^{test}.

E. Inspection of star washer teeth

Unlike the picking task, in which the entire feeder surface is imaged, during the inspection, a high-resolution image of each single star washer is acquired and analyzed. Nevertheless, the techniques described above and used for star washers' identification and classification, can be reused for a high-resolution analysis. In particular, the connected component analysis allows for a quick segmentation of the object, even if its pose in relation to camera varies over time. Creation of the polar image (matrix $\mathbf{P}$), especially with a fine-grained angular resolution, simplifies segmentation of each individual tooth. The latter approach will be further described below.

Let $\mathbf{P_B}$ be a binary image derived from $\mathbf{P}$ by thresholding, in which 0-pixels correspond to background and 1-pixels correspond to the washer. Index j describe a column of matrix $\mathbf{P_B}$. Let j_{last} denote the last column with all ones ($\mathbf{P_B}(:, j_{last}) = 1$ and there is no $j > j_{last}$ such that $\mathbf{P}(:, j) = 1$). In such a way, a slice of $\mathbf{P_B}$ to the right from j_{last} would correspond to the teeth area.

After j_{last} is obtained, the task is to find $j_{right} > j_{last}$ which would allow to select $\mathbf{P_R}$, a right-side slice of $\mathbf{P_B}$ that would contain all teeth as isolated connected components:

$$\mathbf{P_R} = \mathbf{P_B}(:, j_{right} : width(\mathbf{P_B})) \tag{7}$$

The algorithm 1 is proposed for determining j_{right}.

Algorithm 1 Computation of j_{right} column index

$w_{P_B} = width(\mathbf{P_B})$
$\mathbf{P_{teeth}} = \mathbf{P_B}(:, j_{last} + 1 : w_{P_B})$
$w = width(\mathbf{P_{teeth}})$
$h = height(\mathbf{P_{teeth}})$
$f = -1$
for $i = 1$ to h **do**
  for $j = 1$ to w **do**
    if $(\mathbf{P_{teeth}}(i, j) = 0) \wedge (j > f) \wedge (j < \frac{w - j_{last}}{4})$ **then**
      $f = j$
      break
    end if
  end for
end for
$j_{right} = j_{last} + f$
return j_{right}

After j_{right} and $\mathbf{P_R}$ are obtained, the connected component analysis can be done upon $\mathbf{P_R}$, similarly to the process of finding parts on the feeder (subsection IV-B). In the case of $\mathbf{P_R}$, for each of the connected components, y_{top} will correspond to the angle at which the given tooth begins (θ_{start}), and $y_{top} + h$ will correspond to the angle at which the given tooth ends (θ_{stop}):

$$\begin{aligned} \theta_{start} &= y_{top} \frac{2\pi}{n_{angles}} \\ \theta_{stop} &= (y_{top} + h) \frac{2\pi}{n_{angles}} \end{aligned} \tag{8}$$

Having these angles, each star washer tooth can be segmented into an individual rotated image. Figure 7 shows a fragment of a P_B and P matrices with the segmentation results described above. The red line correspond to j_{last}, the green line - j_{teeth}, and the cyan and magenta lines depict the starting and ending angles of each tooth.

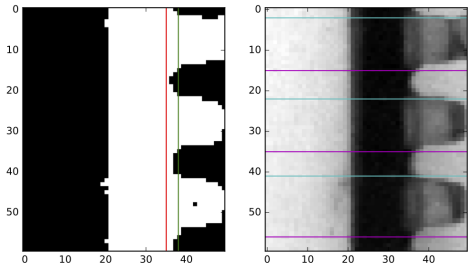

Fig. 7. A fragment of teeth segmentation

V. Results

The solutions and algorithms described in section IV were implemented using Python stack for data analysis and scientific computing, and the associated tools for computer vision, image processing and machine learning: NumPy [11], SciPy [12], Matplotlib [13], Pandas [14], OpenCV [9], Scikit-Image [15], and Scikit-learn [16].

Images used for verification of the solutions, were acquired in the laboratory environment using the available industrial cameras (see section III).

As described in IV-D, different ML classification algorithms has to be tested upon the available data to verify the applicability of ML solutions in general and to determine the most accurate classifiers. Recalling the data sets in (6), the set $X^{test} = X_0^{test} \cup X_1^{test}$ is used for testing the quality of classification.

Let TP (true positives) denote a number of data samples in X_1^{test} (in the correct orientation) that were correctly classified, i.e. yielding $o_w = 1$; FP (false positives) a number of data samples in X_0^{test} (in the incorrect orientation) that were classified as $o_w = 1$; TN (true negatives) a number of data samples in X_0^{test} that were classified as $o_w = 0$; FN (false negatives) a number of data samples in X_1^{test} that were classified as $o_w = 0$.

Having the above numbers, the following performance metrics can be computed: precision P, recall R, F-score F.

$$P = \frac{TP}{TP+FP}$$
$$R = \frac{TP}{TP+FN} \quad (9)$$
$$F = \frac{2PR}{P+R}$$

The used dataset contained 120 data samples extracted from a single image (both for the case with $o_w = 0$ and $o_w = 1$). 90 samples were used for training of classifiers, and 30 for their testing, i.e. $|X_0^{train}| = |X_1^{train}| = 90$ and $|X_0^{test}| = |X_1^{test}| = 30$.

For training and testing of classifiers, their Scikit-learn implementation was used. The following classifiers were tested: support vector machine (`SVC`), multinomial naive Bayes (`MultinomialNB`), Gaussian naive Bayes (`GaussianNB`), decision tree (`DecisionTreeClassifier`), adaptive boosting (`AdaBoostClassifier`), random forest (`RandomForestClassifier`), k-nearest neighbors vote (`KNeighborsClassifier`). Table I summarizes the quality characteristics of the above classifiers.

TABLE I
COMPARISON OF CLASSIFIERS

name	tp	tn	fp	fn	prec	rec	fs
SVC	30	0	30	0	0.500	1.000	0.667
MultinomialNB	27	29	1	3	0.964	0.900	0.931
GaussianNB	29	30	0	1	1.000	0.967	0.983
DecisionTreeClassifier	27	24	6	3	0.818	0.900	0.857
AdaBoostClassifier	29	30	0	1	1.000	0.967	0.983
RandomForestClassifier	27	30	0	3	1.000	0.900	0.947
KNeighborsClassifier	24	29	1	6	0.960	0.800	0.873

One can see that classifiers that produced the best results are adaptive boosting and random forest classifier, which resulted in 1 and 3 false negatives respectively. Neither of them produced false positives, which is particularly useful for preventing failed robotic picking.

VI. Conclusion and further work

This paper presented the first results of a functional prototype implementation of a star washer inspection system. At the given point, the architecture of the robotic cell, overwieved in this paper, serves as a guiding framework for practical implementation. So far, the individual parts of the systems has been developed, and the further goal is to integrate these parts into a single system.

Solutions described in this paper concern the machine vision tasks: identification of star washers on the feeder, classification of their orientation, and segmentation of a star washer's teeth as a part of close-range inspection. Techniques used to solve the latter problems include connected component analysis, Cartesian-to-polar transformation, analysis of polar images of various resolutions, and classification using supervised machine learning. It was found, that even though there are no clear visual clues that differentiate the orientation of star washers, by training a machine learning system, it is possible to achieve satisfactory classification results. The most accurate classifiers for the given case are of the ensemble type: adaptive boosting classifier and random forest classifier.

Further work will put an emphasis on robot/vision integration, multi-pose imaging the inspection rig, testing different illumination modes, more detailed image-based measurement during inspection, and system integration.

Acknowledgments

This research is funded by the Norwegian Research Council.

References

[1] G. Rosati, M. Faccio, A. Carli, and A. Rossi, "Convenience analysis and validation of a fully flexible assembly system," in *Emerging Technologies & Factory Automation (ETFA), 2011 IEEE 16th Conference on*, 2011, Conference Proceedings, pp. 1–8.

[2] C. Finetto, G. Rosati, M. Faccio, and A. Rossi, "Implementation framework for a fully flexible assembly system (f-fas)," *Assembly Automation*, 2015.

[3] J. J. Rodrigues, J. S. Kim, M. Furukawa, J. Xavier, P. Aguiar, and T. Kanade, "6d pose estimation of textureless shiny objects using random ferns for bin-picking," *IEEE International Conference on Intelligent Robots and Systems*, pp. 3334–3341, 2012.

[4] G. Rosati, G. Boschetti, a. Biondi, and a. Rossi, "On-line dimensional measurement of small components on the eyeglasses assembly line," *Optics and Lasers in Engineering*, vol. 47, pp. 320–328, 2009.

[5] C. Qixin, F. Zhuang, X. Nianjiong, and F. Lewis, "A binocular machine vision system for ball grid array package inspection," *Assembly Automation*, vol. 25, no. 3, pp. 217–222, 2005.

[6] E. Davies, "Radial histograms as an aid in the inspection of circular objects," *IEE Proceedings D Control Theory and Applications*, vol. 132, no. 4, p. 158, 1985.

[7] Z. Zhang, "A flexible new technique for camera calibration," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 22, no. 11, pp. 1330–1334, 2000.

[8] P. F. Sturm and S. J. Maybank, "On plane-based camera calibration: A general algorithm, singularities, applications," in *Computer Vision and Pattern Recognition, 1999. IEEE Computer Society Conference on.*, vol. 1, 1999, Conference Proceedings, p. 437 Vol. 1.

[9] G. Bradski and A. Kaehler, *Learning OpenCV: Computer Vision with the OpenCV Library*. O'Reilly Media, 2008.

[10] N. Otsu, "A threshold selection method from gray-level histograms," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 9, no. 1, pp. 62–66, Jan 1979.

[11] S. van der Walt, S. C. Colbert, and G. Varoquaux, "The numpy array: A structure for efficient numerical computation," *Computing in Science Engineering*, vol. 13, no. 2, pp. 22–30, March 2011.

[12] E. Jones, T. Oliphant, P. Peterson *et al.*, "SciPy: Open source scientific tools for Python," 2001–, [Online; accessed 2016-03-19]. [Online]. Available: http://www.scipy.org/

[13] J. D. Hunter, "Matplotlib: A 2d graphics environment," *Computing In Science & Engineering*, vol. 9, no. 3, pp. 90–95, 2007.

[14] W. McKinney, "Data structures for statistical computing in python," in *Proceedings of the 9th Python in Science Conference*, S. van der Walt and J. Millman, Eds., 2010, pp. 51 – 56.

[15] S. van der Walt, J. L. Schönberger, J. Nunez-Iglesias, F. Boulogne, J. D. Warner, N. Yager, E. Gouillart, and T. Yu, "scikit-image: image processing in python." *PeerJ*, vol. 2, p. e453, 2014.

[16] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.