Automated Detection in Complex Objects using a Tracking Algorithm in Multiple X-ray Views

Domingo Mery
Department of Computer Science – Pontificia Universidad Católica de Chile
Av. Vicuña Mackenna 4860(143) – Santiago de Chile
dmery@ing.puc.cl http://dmery.ing.puc.cl

Abstract

We propose a new methodology to detect parts of interest inside of complex objects using multiple X-ray views. Our method consists of two steps: 'structure estimation', to obtain a geometric model of the multiple views from the object itself, and 'parts detection', to detect the object parts of interest. The geometric model is estimated by a bundle adjustment algorithm on stable SIFT keypoints across multiple views that are not necessary sorted. The detection of the object parts of interest is performed by an ad-hoc segmentation algorithm (application dependent) followed by a tracking algorithm based on geometric and appearance constraints. It is not required that the object parts have to be segmented in all views. Additionally, it is allowed to obtain false detections in this step. The tracking is used to eliminate the false detections without discriminating the object parts of interest. In order to illustrate the effectiveness of the proposed method , several applications –like detection of pen tips, razor blades and pins in pencil cases and detection of flaws in aluminum die castings used in the automative industry– are shown yielding a true positive rate of 94.3% and a false positive rate of 5.6% in 18 sequences from 4 to 8 views.

1. Introduction

X-ray imaging has been developed not only for use in *medical imaging* for human beings, but also for *nondestructive testing* (NDT) of materials and objects –called X-ray testing–, where the aim is to analyze internal elements that are undetectable to the naked eye. The most significant application areas in X-ray testing are:

• **Baggage screening:** Since 9/11 X-ray imaging has become a an important issue. Threat items are more difficult to recognize when placed in close packed bags, when superimposed by other objects, and when rotated [31].

• **Foods:** In order to ensure food safety inspection, the following interesting applications have been developed: detection of foreign objects in packaged foods, detection of fishbones in fishes, identification of insect infestation, and fruit and grain quality inspection [8].

• **Cargos:** With the ongoing development of international trade, cargo inspection becomes more important. X-ray testing has been used for the evaluation of the contents of cargo, trucks, containers, and passenger vehicles to detect the possible presence of many types of contraband [5].

• **Castings:** In order to ensure the safety of construction of certain automotive parts, it is necessary to check every part thoroughly using X-ray testing. Within these parts –considered important components for overall roadworthiness–, non-homogeneous regions like bubble-shaped voids or fractures, can be formed in the production process [17].

• **Weldings:** In welding process, a mandatory inspection using X-ray testing is required in order to detect defects (porosity, inclusion, lack of fusion or penetration and cracks). X-ray images of welds is widely used for the detecting those defects in the petroleum, chemical, nuclear, naval, aeronautics and civil construction industries [12].

We observe, that there are some application areas, like castings inspection, where automated systems are very effective; and other application areas, like baggage screening, where human inspection is still used. Additionally, there are certain application areas, like weldings and cargos, where the inspection is semi-automatic. Finally, there is some research in food science where food quality is beginning to be characterized using X-ray imaging. X-ray testing remains an open question and it still suffers from: i) *loss of generality* because approaches developed for one application may not be used in other one; ii) *deficient detection accuracy* because commonly there is a fundamental trade–off between false alarms and miss detections; iii) *limited robustness* because prerequisites for the use of a method are often fulfilled in simple structures only; and iv) *low adaptiveness* because it may be very difficult to accommodate an automated system to design modifications or different specimens.

In this paper, we propose a methodology based on state-of-the-art techniques of computer vision to perform an automated X-ray testing that can contribute to reduce the four problems mentioned above. We believe that our methodology is an useful alternative for examining complex objects in a more *general*, *accurate*, *robust* and *adaptive* way, because we analyze an X-ray image sequence of a target object in several viewpoints automatically and adaptively.

Why multiple views? It is well known that *an image says more than thousand words*, however, this is not always true if we have an intricate image. In this sense, multiple view analysis can be a powerful option for examining complex objects where uncertainty can lead to misinterpretation. Multiple view analysis offers advantages not only in 3D interpretation. Two or more views of the same object taken from different viewpoints can be used to confirm and improve the diagnostic done by analyzing only one image. In the last years, there are many important contributions in multiple view analysis, to cite a few: in object class detection where multiple view relationships are incorporated in order to perform viewpoint-independent inference [28, 23, 26], in motion segmentation where moving objects must be separated in image sequences [32], in visual motion registration in order to construct maps and estimate positions for mobile robots (SLAM problem) [11], in 3D reconstruction [1, 22, 25], in people tracking in a dense crowd [6], in breast cancer in screening [27] and in quality control using multiple view inspection [3, 19]. In these fields, the use of multiple view information yields a significant improvement in performance. Nevertheless, multiple view analysis have not been exploited in such areas where vision systems have been usually focused on single view analysis, like in baggage screening for example. In this cases, certain items are difficult to be recognized using only one viewpoint, as we illustrate in Fig. 1, where we detected pen tips in a pencil case using our proposed method. It is clear, that this detection performance could not be achieved with only the first view of the sequence.

The key idea of our general methodology to detect parts of interest in complex object using multiple views is: *i)* to segment potential parts (regions) of interest in each view using an application dependent method that analyzes 2D features in each single view ensuring the detection of the object parts of interest (not necessarily in all views) and allowing false detections, *ii)* to match and track the potential regions based on similarity and geometrical multiple views constraints eliminating those that cannot be tracked, and *iii)* to analyze the tracked regions including those views where the segmentation fails (the positions can be predicted by reprojection). Similar ideas in specific applications were developed for flaw detection in calibrated X-ray image sequences of automotive parts [19], and for non calibrated multiple view inspection of bottles using fiduciary markers [3], however, the main differences with our algorithm, besides the generalization, are: *i)* The multiple view geometric model -required by the tracking algorithm- is estimated from the object itself using a bundle adjustment algorithm on stable SIFT keypoints across multiple views. *ii)* It is not required to have sorted views, because a visual vocabulary tree can be constructed for fast image indexing. *iii)* Our tracking algorithm uses a fast implementation based on *kdtree* structures. In addition, the MATLAB code is available on our webpage [18].

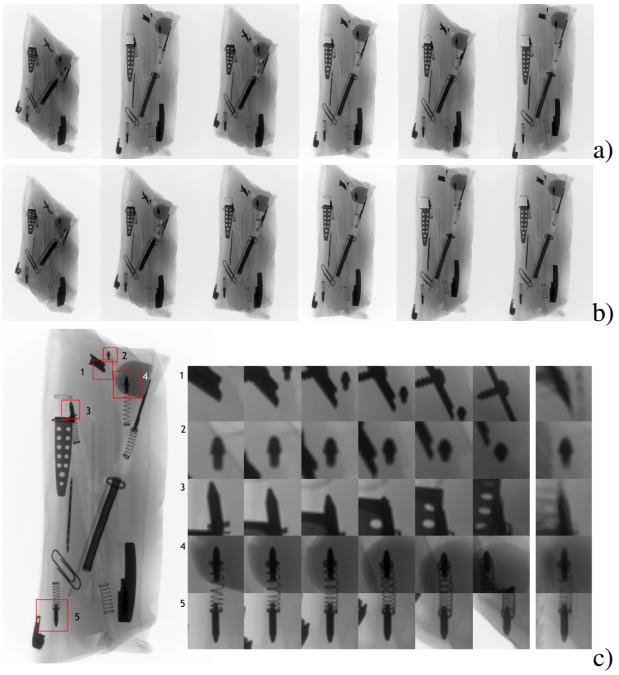

Figure 1. Detection of pen tips in a pencil case: a) Unsorted sequence with 6 X-ray images, 501 $\times$ 315 pixels. b) Sorted sequence. c) Detection (left) and tracked regions (right) in each view (last column is the average). Detection 1 is false. See performance statistics in Tab. 1 (Seq. 5), and details of the detection in Fig. 4. This sequence is used throughout this paper to illustrate the different steps of the proposed method.

Our main contribution is a multiple X-ray view generic methodology that can be used in detection problems that cannot be solved using a single view. Our approach is robust, for example, against poor segmentation or noise because these false detections are not attached to the object and therefore they cannot be tracked. We tested our algorithm in 18 cases (four applications using different segmentation approaches) yielding promising results: a true positive rate of 94.3% with a false positive rate of 5.6%.

In this paper we present: the proposed approach (Section 2), the results obtained in several experiments (Section 3), and some concluding remarks and suggestions for future research (Section 4).

2. Proposed approach

After the X-ray image acquisition, the proposed method follows two main steps: 'structure estimation', to obtain a geometric model of the multiple views from the object itself, and 'parts detection', to detect the object parts of interest (see Fig. 2).

2.1. Structure estimation [step 1]

The approach outlined in this section is based on well known structure from motion (SfM) methodologies. For the sake of completeness, a brief description of this model is presented here. In our work, SfM is estimated from a sequence of m images taken from a rigid object at different viewpoints. The original image sequence is stored in m images $\mathbf{J}_1$, ..., $\mathbf{J}_m$.

Keypoints [step 1a]: For each image, SIFT keypoints are extracted because they are very robust against scale, rotation, viewpoint, noise and illumination changes [13]. Thus, not only a set of 2D image positions $\mathbf{x}$, but also descriptors $\mathbf{y}$, are obtained. Although this method is based on SIFT descriptors, there is no limitation to use other descriptors, *e.g.*, SURF [2].

Image sorting [step 1b]: If the images are not sorted, a visual vocabulary tree is constructed for fast image indexing. Thus, a new image sequence $\mathbf{I}_1$, ..., $\mathbf{I}_m$ is established from $\mathbf{J}_1$, ..., $\mathbf{J}_m$ by maximizing the total similarity defined as $\sum \text{sim}(\mathbf{I}_i, \mathbf{I}_{i+1})$, for $i = 1, ..., m-1$, where the similarity function 'sim' is computed from a normalized scalar product obtained from the visual words of the images [24]. See an example in Fig. 1a and 1b.

Matching points [step 1c]: For two consecutive images, $\mathbf{I}_i$ and $\mathbf{I}_{i+1}$, SIFT keypoints are matched using the algorithm suggested by Lowe [13] that rejects too ambiguous matches. Afterwards, the Fundamental Matrix between views i and $i+1$, $\mathbf{F}_{i,i+1}$, is estimated using RANSAC [10] to remove outliers. If keypoint k of $\mathbf{I}_i$ is matched with keypoint k' of $\mathbf{I}_{i+1}$, the match will be represented as $\mathbf{x}_{i,k} \rightarrow \mathbf{x}_{i+1,k'}$.

Structure tracks [step 1d]: We look for all possible structure tracks –with one keypoint in each image of sequence– that belong to a family of the following matches:

$\mathbf{x}_{1,k_1} \rightarrow \mathbf{x}_{2,k_2} \rightarrow \mathbf{x}_{3,k_3} \rightarrow ... \rightarrow \mathbf{x}_{m,k_m}$.

There are many matches that are eliminated using this approach, however, having a large number of keypoints there are enough tracks to perform the bundle adjustment. We define n as the number of tracks.

Bundle adjustment [step 1e]: The determined tracks define n image point correspondences over m views. They are arranged as $\mathbf{x}_{i,j}$ for $i = 1, ..., m$ and $j = 1, ...n$. Bundle adjustment estimates 3D points $\hat{\mathbf{X}}_j$ and camera matrices $\mathbf{P}_i$ so that $\sum ||\mathbf{x}_{i,j} - \hat{\mathbf{x}}_{i,j}||$ is minimized, where $\hat{\mathbf{x}}_{i,j}$ is the projection of $\hat{\mathbf{X}}_j$ by $\mathbf{P}_i$. If $n \geq 4$, we can use the *factorization algorithm* [10] to perform an affine reconstruction because for our proposes the affine ambiguity of 3D space is irrelevant[1]. This method gives a fast and closed-form solution using SVD decomposition. A RANSAC approach is used to remove outliers.

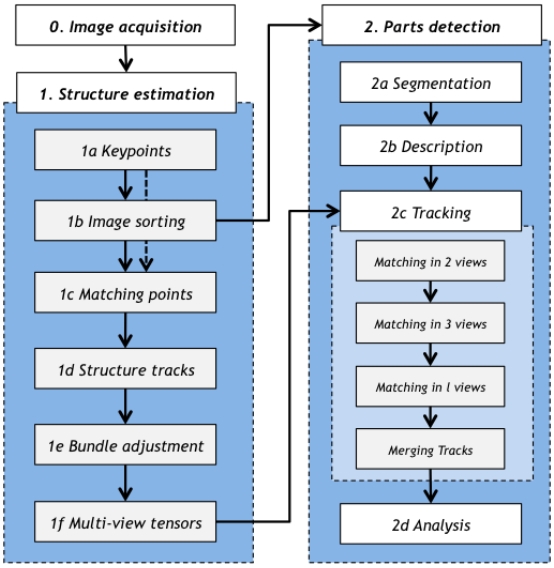

Figure 2. Block diagram of the proposed approach.

Multiple view tensors [step 1f]: Bundle adjustment provides a method for computing bifocal and trifocal tensors from projection matrices $\mathbf{P}_i$ [10], that will be used in the next section.

A similar way to find matching points was proposed in [22, 25] to reconstruct a 3D scenes. The differences to our approach are: *i)* we use certain features to estimate the geometric model, and other features to detect the object parts of interest; and *ii)* we compute the geometric model directly in a closed form using a factorization algorithm.

2.2. Parts detection [step 2]

In this section we give details of the algorithm that detects the object parts of interest. The algorithm consists of four steps: segmentation, description, tracking and analysis as shown in Fig. 2.

Segmentation [step 2a]: Potential regions of interest are segmented in each image $\mathbf{I}_i$ of the sequence. It is an *ad-hoc* procedure that depends on the application. For instance, one can be interested in detecting razor blades or pins in a bag, or flaws in a material, etc. This step ensures the detection of the object parts of interest allowing false detections. The discrimination between these two classes takes place by tracking them across multiple views (see steps 2c

[1] In this problem, the projective factorization can be used as well [10], however, our simplifying assumption is that only small depth variations occur and an affine model may be used.

and 2d). In our experiments we tested three segmentation approaches.

• **Spots detector:** The X-ray image is filtered using a 2D median filter. The difference between original and filtered images is thresholded obtaining a binary image. A potential region r is segmented if size, shape and contrast criteria are fulfilled. This approach was used to detect small parts (like pen tips or pins in a pencil case).

• **Crossing line profile (CLP):** Laplacian of Gaussian edges are computed from the X-ray image. The closed and connected contours of the edge image define region candidates. Grey level profiles along straight lines crossing each region candidate in the middle are extracted. A potential region r is segmented if the profile that contains the most similar grey levels in the extremes fulfills contrast criteria [15]. This approach was used to detect discontinuities in a homogeneous material, *e.g.*, flaws in automotive parts.

• **SIFT matching:** SIFT descriptors are extracted from the X-ray image. They are compared with SIFT descriptors extracted from the image of a reference object of interest. A potential region r is segmented if the descriptors fulfill similarity criteria [13, 7]. This approach was used to detect razor blades in a bag.

Other general segmentation approaches can be used as well. For example, methods based on saliency maps [20], Haar basis features [30], histogram of oriented gradients [4], corner detectors [9], SURF descriptors [2], Maximally Stable regions [14], Local Binary Patterns [21], etc.

Description [step 2b]: Each segmented potential region r is characterized using a SIFT descriptor. The scale of the extracted descriptor, *i.e.*, the width in pixels of the spatial histogram of 4×4 bins, is set to $\sqrt{A_r}$, where A_r is the corresponding area of the region r.

Tracking [step 2c]: In previous steps, n_1 potential regions where segmented and characterized in the whole image sequence. Each segmented region is labeled with an unique number $r \in \mathbf{T}_1 = \{1, ..., n_1\}$. In view i, there are m_i segmented regions that we arrange in a subset $\mathbf{t}_i = \{r_{i,1}, r_{i,2}, ..., r_{i,m_i}\}$. Thus, $\mathbf{T}_1 = \mathbf{t}_1 \cup \mathbf{t}_2 \cup ...\mathbf{t}_m$. Additionally, we store the 2D coordinates of the centroid of the region r in vector $\mathbf{x}_r$ and the SIFT descriptor, that characterizes the region, in vector $\mathbf{y}_r$.

The matching and tracking algorithm combine all regions to generate consistent tracks of the object parts of interest across the image sequence. The algorithm has the following four steps (see an example in Fig. 3):

• **Matching in 2 views:** We look for all regions in view i_1 that have corresponding regions in the next p views, *i.e.*, we look for those regions $r_1 \in \mathbf{t}_{i_1}$ that have corresponding regions $r_2 \in \mathbf{t}_{i_2}$ for $i_1 = 1, ..., m-p$ and $i_2 = i_1+1, ..., i_1+p$ (we use $p = 3$ in order to reduce the computational cost). The matched regions (r_1, r_2) are those that fulfill the following two constraints: *i)* similarity, the Euclidean distance between $\mathbf{y}_{r_1}$ and $\mathbf{y}_{r_2}$ must be smaller than ε_1; and *ii)* location, the Sampson distance between $\mathbf{x}_{r_1}$ and $\mathbf{x}_{r_2}$, *i.e.*, the first-order geometric error of the epipolar constraint, is smaller than ε_2 (in this case, we use $\mathbf{F}_{i_1,i_2}$, the Fundamental Matrix between views i_1 and i_2 computed from projection matrices $\mathbf{P}_{i_1}$ and $\mathbf{P}_{i_2}$ [10]).

Finally, we obtain a new matrix $\mathbf{T}_2$ sized $n_2 \times 2$ with all matched duplets (r_1, r_2), one per row. If a region is not matched with any other one, it is eliminated. Multiple matching, *i.e.*, a region that is matched with more than one region, is allowed. Using this method, problems like non-segmented regions or occluded regions in the sequence, can be solved by the tracking if a region is not segmented in consecutive views.

• **Matching in 3 views:** From the matched regions stored in matrix $\mathbf{T}_2$, we look for triplets (r_1, r_2, r_3), with $r_1 \in$

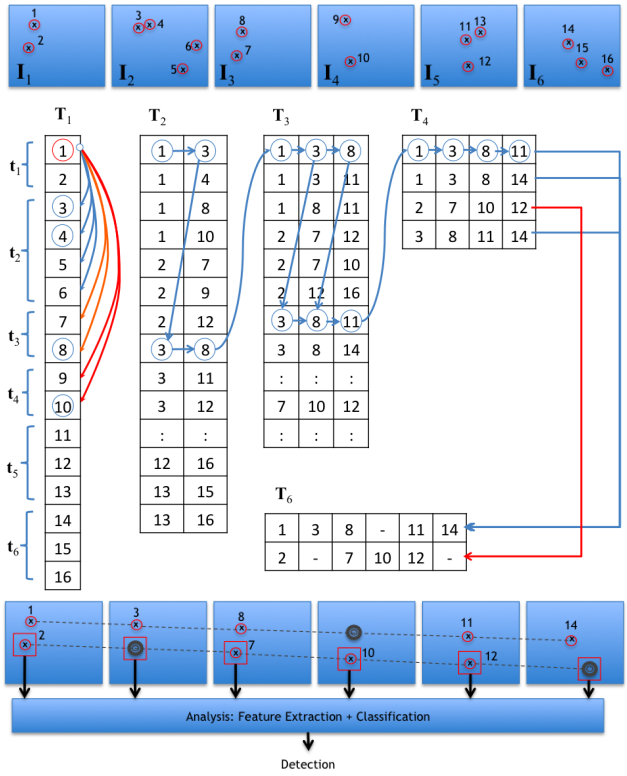

Figure 3. Tracking example with $m = 6$ views. In each view there are 2, 4, 2, 2, 3 and 3 segmented regions, *i.e.*, there are $n_1 = 16$ regions in total. For each region we seek corresponding regions in the next $p = 3$ views (see matching arrows in $\mathbf{T}_1$: region 1 with regions (3,4,5,6) in view 2, regions (7,8) in view 3, and (9,10) in view 4). We observe that after tracking in 2, 3 and 4 views there are only two tracks in $\mathbf{T}_6$ that could be tracked in 5 and 4 views respectively. The regions that were not segmented can be recovered by reprojection (see gray circles in views 2, 4 and 6). Finally, all each set of tracked regions are analyzed in order to take the final decision.

$\mathbf{t}_{i_1}$, $r_2 \in \mathbf{t}_{i_2}$, $r_3 \in \mathbf{t}_{i_3}$ for views i_1, i_2 and i_3. We know that a row k in matrix $\mathbf{T}_2$ has a matched duplet $[T_2(k,1)\ T_2(k,2)] = [r_1\ r_2]$. We look now for those rows j in $\mathbf{T}_2$ where the first element is equal to r_2, *i.e.*, $[T_2(j,1)\ T_2(j,2)] = [r_2\ r_3]$. Thus, we find a matched triplet (r_1, r_2, r_3), if the regions r_1, r_2 and r_3 fulfill the trifocal constraint, *i.e.*, the first-order geometric error of the trilinear constraints (Sampson distance) is smaller than ε_3 [10]. This distance is computed using the trifocal tensors of views i_1, i_2, i_3 from from projection matrices $\mathbf{P}_{i_1}$, $\mathbf{P}_{i_2}$ and $\mathbf{P}_{i_3}$. We build a new matrix $\mathbf{T}_3$ sized $n_3 \times 3$ with all matched triplets (r_1, r_2, r_3), one per row. Regions that do not find correspondence in three views are eliminated.

• **Matching in l views:** For $l > 3$, we built matrix $\mathbf{T}_l$, sized $n_l \times l$, with all possible l-tuplets $(r_1, r_2, ..., r_l)$ that fulfill $[T_{l-1}(k,1)\ ...\ T_{l-1}(k,l-1)] = [r_1\ r_2\ ...\ r_{l-1}]$ and $[T_{l-1}(j,1)\ ...\ T_{l-1}(j,l-1)] = [r_2\ ...\ r_{l-1}\ r_l]$, for $j, k = 1, ..., n_{l-1}$. No more geometric constraint is required because it is redundant. This procedure is performed for $l = 4, ..., q$, with $q \leq m$, and the final result is stored in matrix $\mathbf{T}_q$. For example, for $q = 4$ we store in matrix $\mathbf{T}_4$ the matched quadruplets (r_1, r_2, r_3, r_4) with $r_1 \in \mathbf{t}_{i_1}$, $r_2 \in \mathbf{t}_{i_2}$, $r_3 \in \mathbf{t}_{i_3}$, $r_4 \in \mathbf{t}_{i_4}$ for views i_1, i_2, i_3 and i_4. In our experiments we set $q = 4$ because *i)* a tracking in more views could lead to the elimination of those real regions that were segmented in only four views, and *ii)* a tracking in less views increases the probability of false alarm because the number of tuples in the matrices is very large.

The matching condition for building matrix $\mathbf{T}_i, i = 3, ..., q$, is efficiently evaluated (avoiding an exhaustive search) by using a *kdtree* structure to search the nearest neighbors for zero Euclidean distance between the first and the last $i-2$ columns in $\mathbf{T}_{i-1}$.

• **Merging tracks:** Matrix $\mathbf{T}_q$ defines tracks of regions in q views. We observe that many of these tracks correspond to the same region. For this reason, we can merge those tracks that have $q-1$ common elements. In addition, if a new track has more than one region per view, we can select the region that shows the minimal reprojection error after computing the corresponding 3D location. In this case a 3D reconstruction of $\hat{\mathbf{X}}$ is estimated from tracked points using least squares [10]. Finally, we obtain matrix $\mathbf{T}_m$ with all merged tracks in the m views.

Analysis [step 2d]: The 3D reconstructed points $\hat{\mathbf{X}}$ can be reprojected in those views where the segmentation may have failed to obtain the complete track in all views (see gray circles in Fig. 3). The reprojected points of $\hat{\mathbf{X}}$ should correspond to the centroids of the non-segmented regions. Now, we can calculate the size of the projected region as an average of the sizes of the identified regions in the track. In each view, we define a small window centered in the computed centroids. Afterwards, we can analyze each set of tracked regions. In this step we have the opportunity to use features in images where the segmentation fails (or would be required to be set more lenient). A simple way is to average the tracked windows. See an example in Fig. 1c. Since regions must appear as contrasted zones relating to their environment, we verify if the contrast of each averaged window is greater than ε_C. Thus, the attempt is made to increase the signal-to-noise ratio by the factor $\sqrt{m'}$, where m' is the number of averaged windows. If the region is viewed in all images, $m' = m$. More sophisticated features and classifiers can be implemented in order to ensure a correct detection.

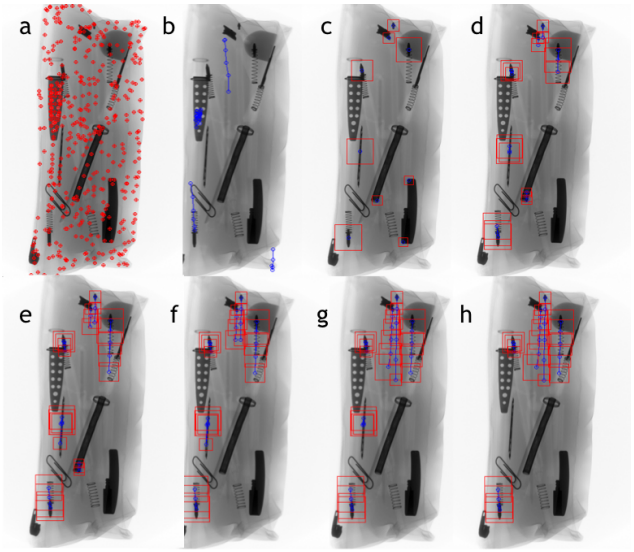

Figure 4. Detection of pen tips in a pencil case step by step (only the last image of the sequence is shown): a) SIFT keypoints, b) structure tracks after bundle adjustment, c) segmentation of parts of interest, d) matched duplets, e) matched triplets, f) matched quadruplets, g) merged tracks, h) verified tracks. The final detection is shown in Fig.1 (see statistics in Tab. 1, Seq. 5).

3. Experimental results

We experimented on X-ray images from 4 different applications: *i)* detection of pen tips (Fig. 1 and Fig. 4), *ii)* detection of razor blades (Fig. 5), *iii)* detection of pins (Fig. 6), and *iv)* detection of discontinuities in aluminum wheels (Fig. 7 and Fig. 8). The first three experiments deal with detection of inner parts in pencil cases. In this sense, the proposed approach could be used in baggage screening. The last experiment corresponds to a non-destructive testing that can be used in automated quality control tasks. The characteristics of the images used in our experiments are various, and the applications are different, however, the problem is common: the segmentation performed in only one image can lead to misclassification. In the segmentation step, we used the spot detector in applications *i* and *iii*;

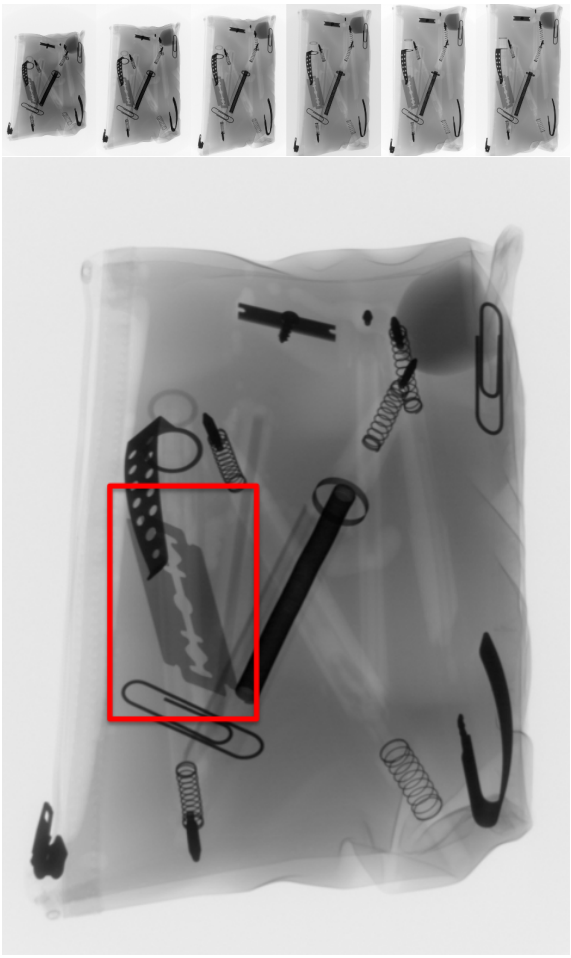

Figure 5. Detection of a razor blade in a pencil case. Top: sequence with 6 X-ray images, 501 $\times$ 305 pixels. Bottom: detection. See performance statistics in Tab. 1, Seq. 10.

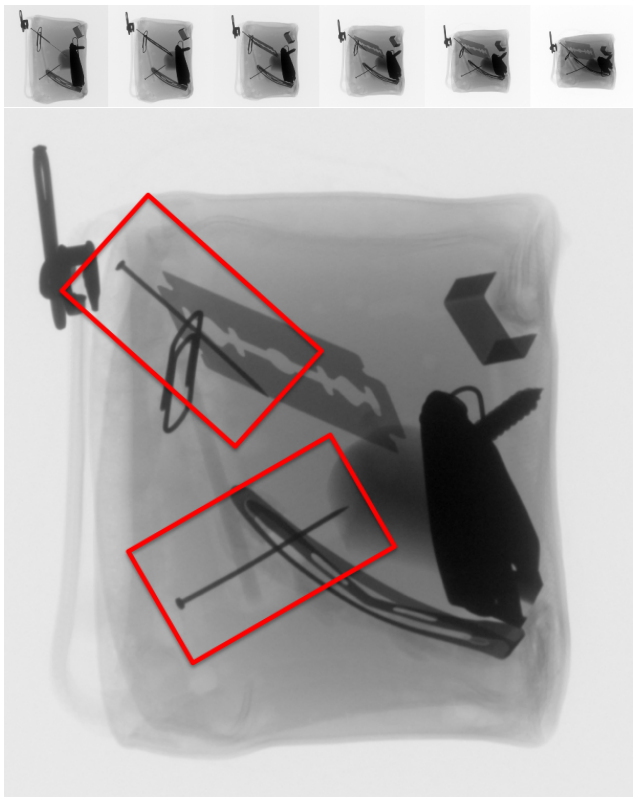

Figure 6. Detection of pins in a pencil case. Top: sequence with 6 X-ray images, 298 $\times$ 298 pixels. Bottom: detection of two pins. See performance statistics in Tab. 1, Seq. 8.

SIFT-matching in application *ii*; and CLP in application *iv*.

In order to illustrate our method step by step we will use the detection of pen tips. In this example, an unsorted image sequence –as shown in first image sequence of Fig. 1– is processed. The sorted image sequence is shown in the second sequence of Fig. 1. The following detection steps are shown in Fig. 4. Many SIFT keypoints were detected (Fig. 4a), however, only a few number of them were stable in the whole sequence in order to perform the bundle adjustment (Fig. 4b). Once the structure was estimated, the segmentation ensured the detection of the pen tips allowing many false detections (Fig. 4c). In this case there were 9 tracked regions, however, only 4 of them corresponded to pen tips. Fig. 4e-h illustrate the tracking process that eliminated false alarms without discriminating the pen tips. The final detection is shown in Fig. 1 (bottom). We can see that all pen tips were detected, even with occlusion (see region 3 in views 1, 2, 3 and 4), however, there was a false positive (region 1) that corresponds to a pencil sharpener with a very similar pattern.

Table 1 shows statistics on 18 sequences of digital X-ray images (101 images). Some of them are illustrated in the mentioned figures. m corresponds to the number of images in the sequence. SIFT/m means the average of the number of SIFT keypoints extracted per image. BA is the number of structure tracks found by bundle adjustment algorithm. n_1 is the number of segmented regions in the whole image sequence, and n_1/m is the average of segmented regions per image. n_l, is the number of l-tuplets tracked in the sequence according to $\mathbf{T}_l$. n_d is the number of detected parts. GT is the number of existing parts (ground truth). FP and TP are the number of false and true positives (n_d=FP+TP). Ideally, FP $= 0$ and TP = GT. In these experiments, the true positive rate, computed as TPR = $\sum$ (TP /GT) is 94.3%, and the false positive rate, computed as FPR = $\sum$ (FP/n_d) is 5.6%. If we compare single versus multiple view detection, the number of regions detected per image is drastically reduced by tracking and analysis steps from n_1/m to n_d (in total from 278 to 71, *i.e.*, 25.5%).

Additionally, it is interesting to observe the results in the detection of aluminum wheel discontinuities (applica-

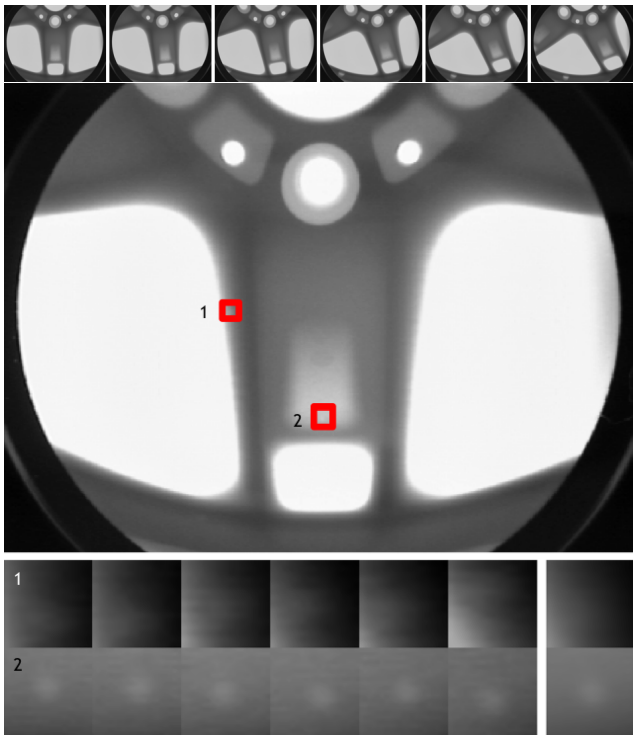

Figure 7. Aluminum wheel with bobbles. Top: sequence with 6 X-ray images, 572 × 768 pixels. Middle: detection. Bottom: tracked regions in each view (last column is the average). The average diameter of these regions is 4.2 and 5.7 pixels respectively. See performance statistics in Tab. 1, Seq. 11.

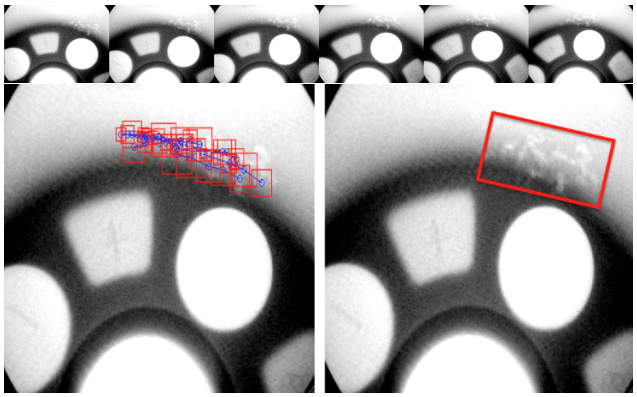

Figure 8. Detection of multiple defects in a wheel. Top: sequence with 6 X-ray images, 286 × 384 pixels. Bottom: detection of 27 small flaws. See performance statistics in Tab. 1, Seq. 7.

tion *iv*), where a similar performance is obtained in [19], however, in our approach we avoid the calibration step [16].

We used the implementation of SIFT, visual vocabulary and *kdtree* form VLFeat [29]. The rest of algorithms were implemented in MATLAB. In the 'Analysis' step we used the contrast on the average window (without rectification)

Table 1. Detection in 18 sequences (see text).

Seq.	size	m	SIFT/m	BA	n_1	n_1/m	n_2	n_3	n_4	n_q	n_d	GT	FP	TP
1	286 × 384	4	95	9	21	5	35	31	7	2	2	3	0	2
2	501 × 315	4	257	99	59	15	57	32	7	7	7	5	2	5
3	572 × 768	4	1045	33	94	24	34	7	1	1	1	1	0	1
4	298 × 298	5	99	33	81	16	55	39	18	5	2	2	0	2
5	501 × 315	5	209	33	48	10	45	32	12	6	5	5	1	4
6	501 × 315	5	508	6	11	2	5	4	1	1	1	1	0	1
7	286 × 384	6	222	14	236	39	673	1325	1975	67	28	28	0	27
8	298 × 298	6	97	22	99	17	70	58	39	5	2	2	0	2
9	501 × 315	6	211	28	60	10	59	46	27	6	5	4	1	4
10	501 × 315	6	521	42	13	2	8	8	3	1	1	1	0	1
11	572 × 768	6	878	23	103	17	94	23	10	2	2	2	0	2
12	572 × 768	6	823	12	91	15	77	43	20	5	2	2	0	2
13	572 × 768	6	1036	5	137	23	62	15	5	1	1	1	0	1
14	572 × 768	6	1032	2	126	21	75	55	29	5	3	3	0	3
15	572 × 768	6	802	22	90	15	77	38	19	7	3	4	0	3
16	572 × 768	6	835	21	97	16	61	30	12	4	2	2	0	2
17	572 × 768	6	824	13	92	15	81	54	25	4	2	2	0	2
18	572 × 768	8	892	5	130	16	123	53	31	4	2	2	0	2
Total	–	101	–	–	–	278	–	–	–	–	71	70	4	66
												100%	5.6%	94.3%

greater than 5%. For multiple view matching, ε_2 = 15 pixeles, ε_3 =25 pixels. The time computing depends on the application, however, in order present a reference, for Fig. 1 the results were obtained after 4.5 seconds on a iMac OS X 10.6.6, processor 3.06GHz Intel Core 2 Duo, 4GB RAM memory. The code of the MATLAB implementation, and the images of Fig. 1 are available on our webpage [18].

4. Conclusions

In this paper we presented a new generic methodology that can be used to detect parts of interest in complex object automatically and adaptively. The proposed approach is an application of state-of-art computer vision techniques. It filters out false positives resulting from segmentation steps performed on single views of an object by corroborating information across multiple views.

Our method consists basically of two steps: 'structure estimation' to obtain a geometric model of the multiple views and 'parts detection' to detect object parts of interest of an object. The geometric model is estimated by a bundle adjustment algorithm on stable SIFT keypoints across multiple views that are not necessary sorted.

The proposed approach takes the opportunity to detect regions, in images where the segmentation fails or would be required to be set more lenient. Using 3D information estimated by using the geometric model, the region location (of a tracked region) could be predicted even in images where it was not segmented.

Our algorithm was efficiently implemented using *kdtree* structures. The algorithm was tested on 18 cases (four applications using different segmentation approaches) yielding promising results: a true positive rate of 94.3% with a false positive rate of 5.6%.

As future work, we will implement our method in sequences with more images using sparse bundle adjustment. We will test the method in more complex scenarios using more general segmentation algorithms based on sliding windows, and more sophisticated features and classifiers in the multiple view analysis step.

Acknowledgments

This work was supported by grant Fondecyt No. 1100830 from CONICYT–Chile. The author thanks to Vladimir Riffo for X-ray images taken from pencil cases.

References

[1] S. Agarwal, N. Snavely, I. Simon, S. Seitz, and R. Szeliski. Building Rome in a day. In *IEEE 12th International Conference on Computer Vision (ICCV2009)*, pages 72–79, 2009.

[2] H. Bay, T. Tuytelaars, and L. Van Gool. Surf: Speeded up robust features. In *9th European Conference on Computer Vision (ECCV2006)*, Graz Austria, May 2006.

[3] M. Carrasco, L. Pizarro, and D. Mery. Visual inspection of glass bottlenecks by multiple-view analysis. *International Journal of Computer Integrated Manufacturing*, 23(10):925–941, 2010.

[4] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. *Conference on Computer Vision and Pattern Recognition (CVPR2005)*, 1:886–893, 2005.

[5] X. Duan, J. Cheng, L. Zhang, Y. Xing, Z. Chen, and Z. Zhao. X-ray cargo container inspection system with few-view projection imaging. *Nuclear Instruments and Methods in Physics Research A*, 598:439–444, 2009.

[6] R. Eshel and Y. Moses. Tracking in a dense crowd using multiple cameras. *International Journal of Computer Vision*, 88:129–43, 2010.

[7] A. F. Gobi. Towards generalized benthic species recognition and quantification using computer vision. In *4th Pacific-Rim Symposium on Image and Video Technology (PSIVT2010), Singapore, Nov.14-17, 2010*, pages 94–100, 2010.

[8] R. Haff and N. Toyofuku. X-ray detection of defects and contaminants in the food industry. *Sensing and Instrumentation for Food Quality and Safety*, 2(4):262–273, 2008.

[9] C. Harris and M. Stephens. A combined corner and edge detector. In *Proc. 4th Alvey Vision Conferences*, pages 147–152, 1988.

[10] R. I. Hartley and A. Zisserman. *Multiple view geometry in computer vision*. Cambridge University Press, second edition, 2003.

[11] K. Konolige and M. Agrawal. FrameSLAM: from bundle adjustment to realtime visual mapping. *IEEE Transactions on Robotics*, 24(5):1066–1077, 2008.

[12] T. W. Liao. Improving the accuracy of computer-aided radiographic weld inspection by feature selection. *NDT&E International*, 42:229–239, 2009.

[13] D. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, 2004.

[14] J. Matas, O. Chum, M. Urban, and T. Pajdla. Robust wide-baseline stereo from maximally stable extremal regions. *Image and Vision Computing*, 22(10):761–767, 2004.

[15] D. Mery. Crossing line profile: a new approach to detecting defects in aluminium castings. *Proceedings of the Scandinavian Conference on Image Analysis (SCIA 2003), Lecture Notes in Computer Science*, 2749:725–732, 2003.

[16] D. Mery. Explicit geometric model of a radioscopic imaging system. *NDT & E International*, 36(8):587–599, 2003.

[17] D. Mery. Automated radioscopic testing of aluminum die castings. *Materials Evaluation*, 64(2):135–143, 2006.

[18] D. Mery. BALU: A toolbox Matlab for computer vision, pattern recognition and image processing (http://dmery.ing.puc.cl/index.php/balu), 2011.

[19] D. Mery and D. Filbert. Automated flaw detection in aluminum castings based on the tracking of potential defects in a radioscopic image sequence. *IEEE Trans. Robotics and Automation*, 18(6):890–901, December 2002.

[20] S. Montabone and A. Soto. Human detection using a mobile platform and novel features derived from a visual saliency mechanism. *Image and Vision Computing*, 28(3):391–402, 2010.

[21] T. Ojala, M. Pietikainen, and T. Maenpaa. Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(7):971–987, Jul 2002.

[22] M. Pollefeys, R. Koch, M. Vergauwen, and L. Van Gool. Automated reconstruction of 3D scenes from sequences of images. *ISPRS Journal Of Photogrammetry And Remote Sensing*, 55(4):251–267, 2000.

[23] N. Razavi, J. Gall, and L. van Gool. Backprojection revisited: Scalabale multi-view object detection and similarity metrics for detections. In *Proceedings of the European Conference on Computer Vision (ECCV2010)*, volume LNCS 6311, pages 620–633, 2010.

[24] J. Sivic and A. Zisserman. Efficient visual search of videos cast as text retrieval. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31(4):591–605, 2009.

[25] N. Snavely, S. Seitz, and R. Szeliski. Photo tourism: exploring photo collections in 3D. In *ACM SIGGRAPH 2006 Papers*, pages 835–846. ACM, 2006.

[26] H. Su, M. Sun, L. Fei-Fei, and S. Savarese. Learning a dense multi-view representation for detection, viewpoint classification and synthesis of object categories. In *International Conference on Computer Vision (ICCV2009)*, 2009.

[27] J. Teubl and H. Bischof. Comparison of Multiple View Strategies to Reduce False Positives in Breast Imaging. *Digital Mammography*, pages 537–544, 2010.

[28] A. Thomas, V. Ferrari, B. Leibe, T. Tuytelaars, B. Schiele, and L. Van Gool. Towards multi-view object class detection. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR2006)*, volume 2, pages 1589–1596, 2006.

[29] A. Vedaldi and B. Fulkerson. VLFeat: An open and portable library of computer vision algorithms (http://www.vlfeat.org/), 2008.

[30] P. Viola and M. Jones. Robust real-time object detection. *International Journal of Computer Vision*, 57(2):137–154, 2004.

[31] G. Zentai. X-ray imaging for homeland security. *IEEE International Workshop on Imaging Systems and Techniques (IST 2008)*, pages 1–6, Sept. 2008.

[32] V. Zografos, K. Nordberg, and L. Ellis. Sparse motion segmentation using multiple six-point consistencies. In *Proceedings of the Asian Conference on Computer Vision (ACCV2010)*, 2010.