Tooth Guard: A Vision System for Detecting Missing Tooth in Rope Mine Shovel

Ser Nam Lim
GE Global Research
limser@ge.com

Joao Soares
Yahoo, Inc.
jvbsoares@gmail.com

Ning Zhou
GE Global Research
ningzhou@ge.com

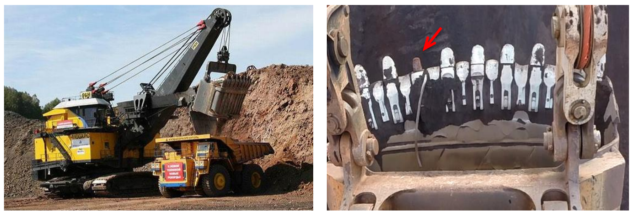
Abstract

Rope shovels are widely used in the mining industry to dig ore. During operation, one or more teeth in the bucket can be lost as a result of the force that impacts the teeth, which causes a serious problem when the broken teeth gets picked up by the haul truck and eventually ended up in the crusher and jamming it. In this paper, we present a vision system for monitoring tooth condition and detecting missing tooth for a mining shovel. Our system leverages the predictable range of motion that the bucket of a rope shovel goes through during operation due to the camera mounting. For this reason, our strategy is to use exemplar based image retrieval and a sliding window procedure to first locate a pre-defined static region of the bucket in a live frame, followed by detecting the tooth line region based on its relative position to the selected exemplar. Once the tooth line region is detected, we proceed by determining its transformation from a recently detected tooth line region, and rectify the tooth line region before conducting image differencing. The difference image is then converted into a response map by sliding a tooth template associated with the retrieved exemplar and computing a correlation score per sliding window. Finally, the tooth line associated with the retrieved exemplar is rigidly scaled and rotated within a specified range in the response map to find a final tooth line position with the largest overall fitting score. An individual tooth is then flagged as missing if its missing measurement exceeds a threshold. The outstanding performance and high reliability of the proposed system have been demonstrated by experiments on video sequences collected from an iron ore mining site and a two-month trial of an installed unit on a production line.

1. Introduction

Rope shovels (Fig. 1(a)) are commonly used in the mining industry for removing large amounts of overburden and ore, at a fraction of the cost per ton. A critical component of the rope shovel is the bucket (Fig. 1(b)) that is responsible for digging up the payload, and is the component that experiences the most impact during operation. As a result, one or more teeth commonly break off, and get picked up by the haul truck unnoticed. A missing tooth event can result in great productivity loss when the tooth eventually ends up in the crusher. Due to the size and hardness of a tooth, it often jams up the crusher, and operation has to be halted in order to deal with the situation, which could also be hazardous. Unfortunately, it is impossible for an operator to find out that a tooth is missing when sitting in the operation room due to the limited sphere of vision. It is extremely beneficial of having a vision system which is able to alert the operator right after a tooth breaks off, so that a tooth change-out can be scheduled to avoid productivity loss and hazardous maintenance.

In this paper, we present a vision system called Tooth Guard to monitor the tooth condition of a mining shovel and detect a missing tooth event using a camera mounted on the shovel's boom (Fig. 3). From an algorithmic perspective of view, the key component of our Tooth Guard system is a missing tooth detection algorithm which utilizes two adjacent video frames. As seen in Fig. 4, we first locate the tooth line regions in each of the two frames and then rectifies the previous tooth line region with the current one based on homography transformation. The difference image between the warped image of previous tooth line region and the image of current tooth line region is computed based on image gradient. To accurately locate the tooth line region, we combine an exemplar-based retrieval and a sliding window procedure to first locate a specified region that is not easily covered up by dirt during operation. For example, we chose a specific area below the tooth line (the bounding box in red in Fig. 7 (a)) since this region is unlikely to be covered by anything and its visual appearance is often consistent. We will call this specific region as "bucket" from here on. In this way, we avoid too many false positives when detecting the tooth line region itself in live frames. The tooth line region (the bounding box in blue in Fig. 7 (a)) is then inferred based on the detected bucket. The difference image is further transformed into a response map by template matching with a tooth image. Finally, a tooth line is found in the response map and a tooth is decided as missing if its missing



(a) A rope mining shovel (b) A bucket with a missing tooth

Figure 1. We present a vision system called tooth guard to monitor and detect missing teeth for a mining shovel.

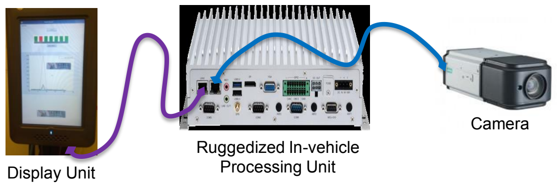
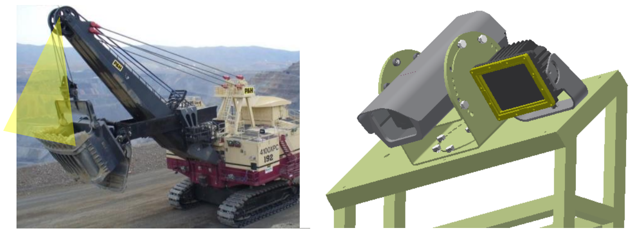


Figure 2. The key hardware components of the tooth guard system include 1) a camera mounted on the boom of a shovel; 2) a computational processing unit ruggedized in the vehicle; and 3) a display unit in front of the operator.



(a) Camera mounting position on shovel (b) Camera mounting bracket

Figure 3. The camera is mounted on the boom of a shovel using a bracket. A high-intensity light source is installed and aligned to provide light for system performing properly during evenings.

measurement is higher than a threshold.

We also introduced in this paper a corresponding annotation tool, Fig. 7, that we used to mark out the bucket exemplars and the associated tooth line regions, tooth templates and tooth locations. The annotation tool helps us to prepare an exemplar set which is used by the missing tooth detection algorithm in an efficient and error free way.

The rest of the paper is organized as follows. We briefly discuss related work in Section 2. In Section 3, we give an overview of the tooth guard hardware system. The missing tooth detection algorithm is described in Section 4, and the annotation tool is presented in Section 5. We present experiments and results in Section 6 and conclude this paper in Section 7.

2. Prior Work

Automatically detection missing tooth of a mining shovel has attracted interests from academia and industry a decade ago. Specifically, Luo *et al.* [7] described a proof-of-concept prototype system in which 3-D laser range data was compared and matched with the CAD model of a tooth line, and the misalignment was then used to determine if any tooth was missing or not. Ridley *et al.* [8] also presented a vision system to detect missing tooth of a shovel where motion segmentation was used to locate the teeth followed by a shape analysis to decide a tooth was missing or not. Our system differs from these prior work in two fundamental aspects. First, our tooth detection algorithm relies on the difference between two adjacent frames through rectification rather than motion segmentation as in [8]. Second, our missing measurement again makes use of the different image to determine a tooth as missing rather than shape analysis as was done in [8, 7].

3. System Overview

The tooth guard system consists of three key hardware units as shown in Fig. 2. They include: 1) an industrial camera which is mounted on the boom of the a shovel to monitor the tooth condition; 2) a computational processing unit which is ruggedized in the shovel operation room to process the video stream for detecting missing teeth; and 3) a small touch-screen display unit which is installed in front of the operator to allow him/her for acknowledging missing tooth detection.

The position of the camera mounting is critical to the success of the tooth guard system. The ideal position should allow the mounted camera to clearly see the bucket and tooth line all the time and also hold the camera stably during operation. We then choose the boom at the wheel end as the camera mounting position as shown in Fig. 3. In addition, an high-intensity auxiliary light source is installed and aligned with the camera using a bracket to provide light for the system running properly in evenings.

4. Missing Tooth Detection Algorithm

Due to the camera mounting, the movement range of the bucket is well constrained. Our missing tooth detection algorithm then leverages the rigid motion of the bucket between two adjacent video frames to detect missing tooth if any (see Fig. 4). First, the tooth line regions is accurately located in each frame respectively using an exemplar-based retrieval method. The previous tooth line region is then rectified with the current one via a homography transformation, and the difference image is computed based on image gradient instead of pixel intensity. We then perform a normalized cross correlation [1, 9] between a tooth template and the difference image in a sliding window manner and thus obtain a matching response map. Finally, the manually annotated tooth line associated with the retrieved bucket exemplar is rigidly scaled and rotated and put into the response map to find the live tooth positions by looking for the the largest overall response scores.

The detection algorithm requires a set of exemplars, $\mathbb{E} = \{E_i\}_{i=1}^{N}$, where N is the number of exemplars and $E_i = \{B_i, P_i, R_i, T_i, L_i\}$ is an individual exemplar which includes a bucket exemplar image B_i, a rectangle P_i recording the size and location of B_i, a rectangle R_i indicating the location of the corresponding tooth line region, a tooth template image T_i and a tooth line L_i which contains M tooth spacial locations where M is the number of teeth. We have also developed an annotation tool (Section 5) to efficiently and effectively collect such an exemplar set for system deployment.

4.1. Bucket and Tooth Region Detection

Locating the tooth line region of a mining shovel during operation is challenging. One of the reasons is that the bucket is easily covered up by dirt or ore which makes its appearance changes dramatically from time to time. We therefore choose to locate a specific region whose appearance is much more consistent during operation instead of the bucket or tooth line region themselves. For example, we selected a specific area below the bucket (the rectangle in red in Fig. 7 (a)) as the region of interest. It is noted that, however, other regions can be also considered if their appearance are consistent during operation, such as the rope wheels. We call this specific region as bucket in the sequel when the context is clear.

To locate the specific bucket region in a live frame, a exemplar-based retrieval method is used here due to its efficiency and the fact that the motion range of the bucket is very predictable. Assuming the N bucket images in the exemplar set $\mathbb{E}$ are uniformly sampled from the entire motion range of the bucket, we can locate the bucket in a online frame by comparing each bucket exemplar image B_i with the live frame at a minimal expense. Specifically, given a live frame and an bucket exemplar image B_i, we generate a number of source image tiles $\{I_i^j\}$ of the same size as B_i by sliding a window around its annotated location P_i in the live frame, and compute the a matching score based on their HoG [3] features, given as

$$match\_score(I_i^j) = d\left(f(B_i), f(I_i^j)\right), \tag{1}$$

where $d(\cdot,\cdot)$ and $f(\cdot)$ is the Euclidean distance operator and HoG feature extractor, respectively. We then rank all the source image tiles generated by all the bucket exemplars by their matching scores, and choose the one with smallest matching score as the detected bucket location.

The tooth line region in current frame is inferred by considering the annotated location of the tooth line region and the positional change between the retrieved image tile (the one with the smallest matching score) and the annotated location of the corresponding bucket exemplar. Let $\hat{P} = \{\hat{x}^p, \hat{y}^p, w^p, h^p\}$ (i.e., a bounding box) be the location of the retrieved image tile where $\hat{x}^p$ and $\hat{y}^p$ are the x and y coordinates of the top-left corner and w^p and h^p are the width and height, respectively. Let B_i be the corresponding bucket exemplar image and $P_i = \{x_i^p, y_i^p, w^p, h^p\}$ is the annotated location of the bucket exemplar image. The positional change between the retrieved image tile and the corresponding bucket exemplar is calculated as

$$\Delta x = \hat{x}^p - x_i^p, \qquad \Delta y = \hat{y}^p - y_i^p. \tag{2}$$

The location of the tooth line region $\hat{R} = \{\hat{x}^r, \hat{y}^r, w^r, h^r\}$ is then computed as

$$\begin{aligned} \hat{x}^r &= x_i^r + \Delta x, & \hat{y}^r &= y_i^r + \Delta y, \\ w^r &= w_i^r, & h^r &= h_i^r, \end{aligned} \tag{3}$$

where $R_i = \{x_i^r, y_i^r, w_i^r, h_i^r\}$ is a bounding box representing the position of the annotated tooth line region which is associated with bucket exemplar B_i.

4.2. Tooth Region Rectification and Differencing

After detecting the tooth line regions in two adjacent frames, we rectify the previous tooth line image and compute the difference image between the warped tooth line image and the current tooth line image. In our system, a homography based rectification is adopted since the bucket motion is rigid and the tooth line area roughly lies in a plane. To estimate the homography transformation between tooth line images I_i and I_j, we resort to the optical flow [6] algorithm with outlier rejection. The homography transformation matrix is computed based on the inlier feature correspondences and then used to rectify the previous image I_i. Finally, we compute the the deference image D_i by comparing the image gradient similarity between the warped tooth line image and the current one. The image gradient is used here instead of intensity for seeking better robustness to noise.

4.3. Tooth Matching and Tooth Line Fitting

In difference image D_i (see Fig. 5) , the tooth line region has much small values than that of the background and tooth tips shape are standing out. We utilize a template matching [2] and rigid tooth line fitting method to locate the final tooth line positions. Assuming the current tooth line region is detected by using the ith exemplar, we slide the associated tooth template image T_i in the difference image D_i and compute the matching response for each location. This gives us a response map H whose values indicate the possibility of having a tooth in the corresponding locations. We then scale and rotate the associated tooth line L_i within certain ranges to generate a collection of candidate tooth lines

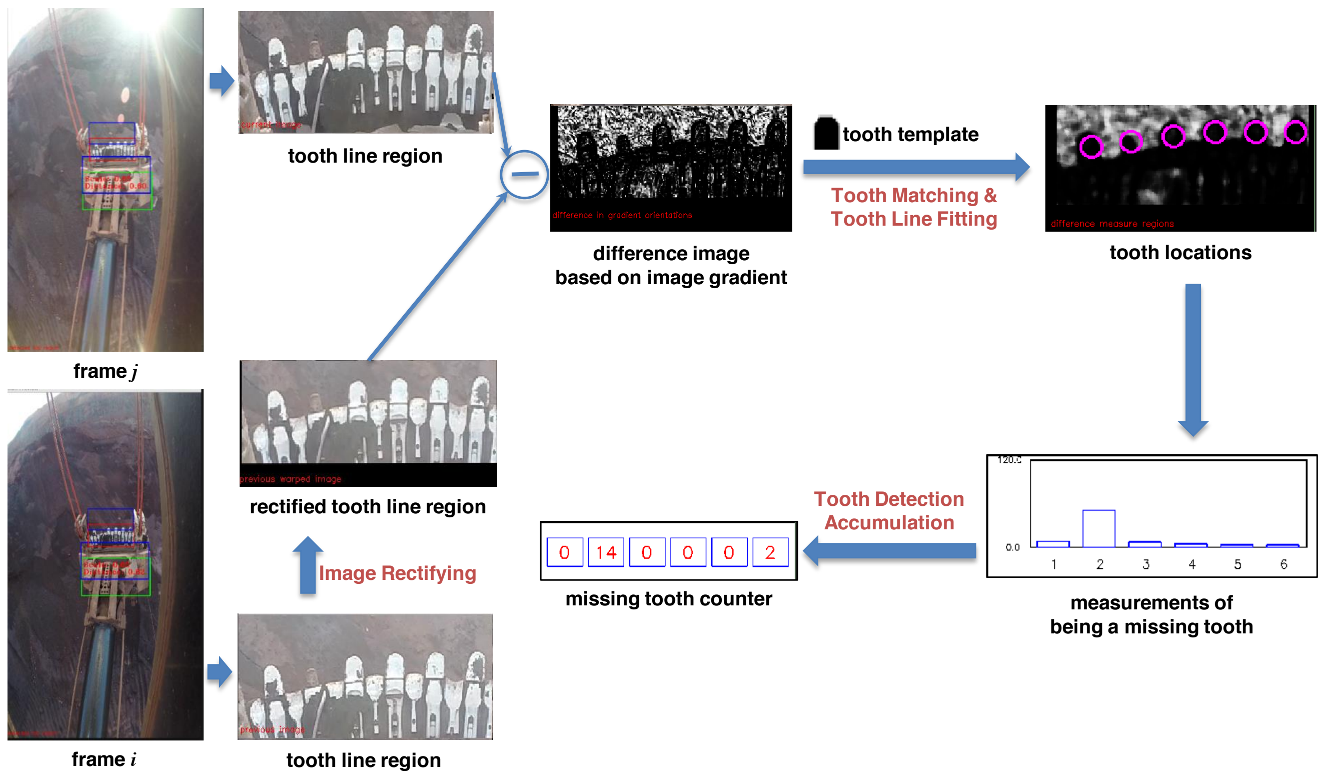


Figure 4. The missing tooth detection algorithm makes use of a previous frame i and a current frame j. The tooth line images are located using a exemplar-based retrieval method. The previous tooth image is then rectified with the current one and a difference image is computed based on their image gradients. The final tooth locations are decided using tooth template matching and rigid line fitting.

$\{l_k\}$, and each line l_k is then scored as

$$fit\_score(l_k) = \max_{p \in \Omega_{l_k}} \left(\sum_{i=1}^{M} H\left(p_i(x), p_i(y)\right) \right), \quad (4)$$

where Ω_{l_k} contains all the valid locations in H for line l_k, p is a vector of length M with each element encode the location of a tooth, $p_i(x)$ and $p_i(y)$ are the x and y coordinates of the ith tooth, respectively. The line of the highest score is taken as the final tooth line indicating the location of each tooth.

4.4. Missing Tooth Measurement and Detection

After locating the positions of the teeth, we need to measure the likelihood of a tooth as missing. Considering a tooth breaks off, more background area would be exposed at the corresponding location in the difference image. We can then make use of this to estimate how likely a tooth is missing. Without loss of generality, considering a tooth is detected as being located at position c with respect to the difference image, we then compute the average value of the region covered by the circle which is centered at c as the measurement of being a missing tooth, given as

$$miss\_score(c) = \frac{1}{|\Theta_c|} \sum_{p \in \Theta_c} D\left(p(x), p(y)\right), \quad (5)$$

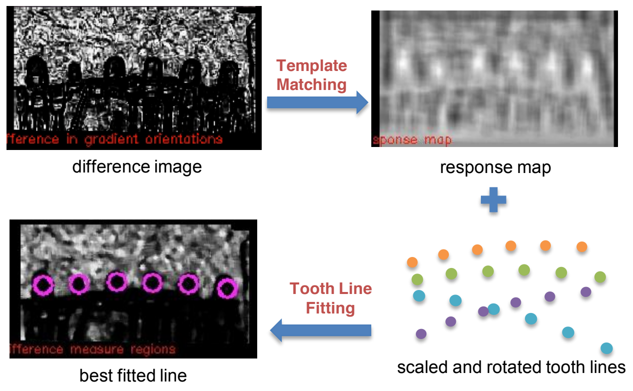


Figure 5. Tooth template matching and tooth line fitting. The response map is generated by using a tooth image as template and the difference image as source. The associated tooth line which is manually annotated (in green) is then scaled and rotated with certain ranges and each of them is put back into the valid locations of the response map for scoring. The line with the highest score is chosen as the final tooth line.

where Θ_c is a circle centered at c with radius r, $|\Theta_c|$ is the total number of points in it, p is a point inside it, D is the difference image which is a matrix, $p(x)$ and $p(y)$ are the x and y coordinates of point p, respectively. The higher the $miss\_score(c)$ value the more likely that the tooth is missing. A tooth is detected as missing if the score is bigger than a threshold to which we refer as missing measurement threshold.

It is too sensitive to only consider the detection result at current frame which would raise many false alarms. It is natural to accumulate the evidence across a number of previous frames. Specifically, if a tooth is decided as missing, we increase its missing count by one, and the system fires an alarm when the count of any tooth is larger than a threshold. We call it missing count threshold.

4.5. Frame Rejection Scheme

The tooth guard system is running 24 hours a day, 7 days a week, in a open-world mining environment. Many factors can affect its performance, such as lighting condition, visibility of the tooth line. To achieve high level performance, a frame rejection scheme is devised to distinguish the good and bad detection results. As seen in Fig. 6, when the detected tooth line (denoted by the purple circles) is indeed on the tooth tips, in the difference image, the area below the detected tooth line is black while the region above it is relatively white. However, when the tooth line is detected incorrectly, none of these two criteria are satisfied. We therefore check the values above and below each detected tooth location and reject current frame if 1) the intensity value in the difference image above any detected tooth is smaller than a threshold; or 2) the intensity value in the difference image below any detected tooth is bigger than a threshold. Our experimental results shows that this rejection scheme is highly effective and yet efficient which brings down the false alarm rates dramatically.

4.6. Parameters

There are a number of parameters in the tooth guard system that can affect its performance. Instead elaborating every parameter here, we describe several parameters that are critical to the system. In fact, for most of the other parameters, we fixed them in various deployment settings and did not see any issue yet.

The missing measurement threshold is an important parameter which is used to determine a tooth is missing or not at each frame. A small threshold would bring about many false positive detection while a big one would probably miss the chance to detect a real missing tooth.

The missing count threshold is used for deciding when to alert the operator. A small value would fire up the alarm more often than a larger one, however, it would increases the false positive rates at the same time. We empirically choose 10 for it and found consistently compelling performance based on different video sequences. In a real-world deployment, we used 10 as well.

It is also important to reset the missing tooth counts to 0's when necessary in order to bring down the false positive rates. We reset the missing count of a tooth if it has been detected as non-missing for a certain number of times. We refer to this number as count reset threshold.

5. Annotation Tool

Annotating the exemplar set $\mathbb{E}$ without any tool is tedious and error prone. A key part of the system is an annotation tool that we have implemented to annotate the set $\mathbb{E}$ which marks the positions and template images of bucket exemplars, the relative positions of corresponding tooth line regions, tooth template images and tooth line locations.

As seen in Fig. 7 (a), one can mark the position of a bucket with a red bounding box (P_i) and a tooth line region with a blue bounding box (R_i). The image content in the blue bounding box is cropped and saved as the bucket template B_i. We can then click "tooth line details" button to annotate the tooth line locations including the "baseline" locations (red dots in Fig. 7 (b)) and "tooth tips" locations (yellow dots in Fig. 7 (b)). The tooth line annotation L_i consists of the marked "tooth tips" locations. Finally, a tooth template image can be marked by clicking button "Mask" and a window as shown in Fig. 7 (c) will prompt. One can click as many as possible points to outline the shape of the selected tooth which is saved as the tooth template T_i.

6. Experiments and Results

In this section, we evaluate the performance of the tooth guard system. We collected 4 video clips from an iron mining site at Brazil with the hardware setup described above. Each video clip is about 18-minute long recording the normal operations of a rope shovel and they were recorded in four different time frames to cover different lighting conditions. In each video clip, we manually took off one or two teeth to simulate a missing tooth event. To annotate the exemplar set, another video sequence other than these 4 video clips is used where the the tooth tips are much more visible. We annotated 98 exemplars which completely covers the entire motion range of the bucket.

We first assess the bucket and tooth line region detection performance. We manually marked the positions of the bucket and tooth line region using bounding boxes in 1,000 randomly selected frames. Detections were judged to be true/false positives by the overlap ratio between detection and ground truth bounding boxes which is widely used in the PASCAL VOC challenge [4]. We plot the detection rates of bucket and tooth line region versus the overlap ratio in Fig. 8. It is seen that our bucket and tooth line region

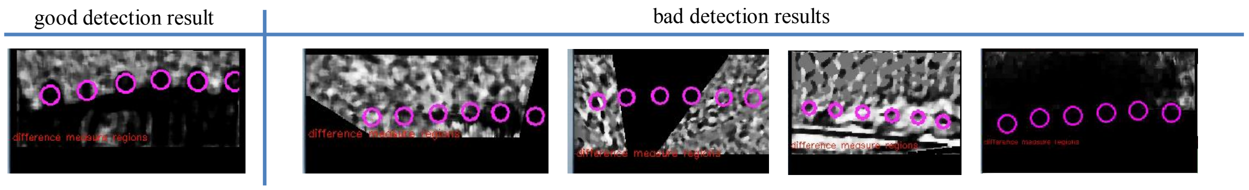
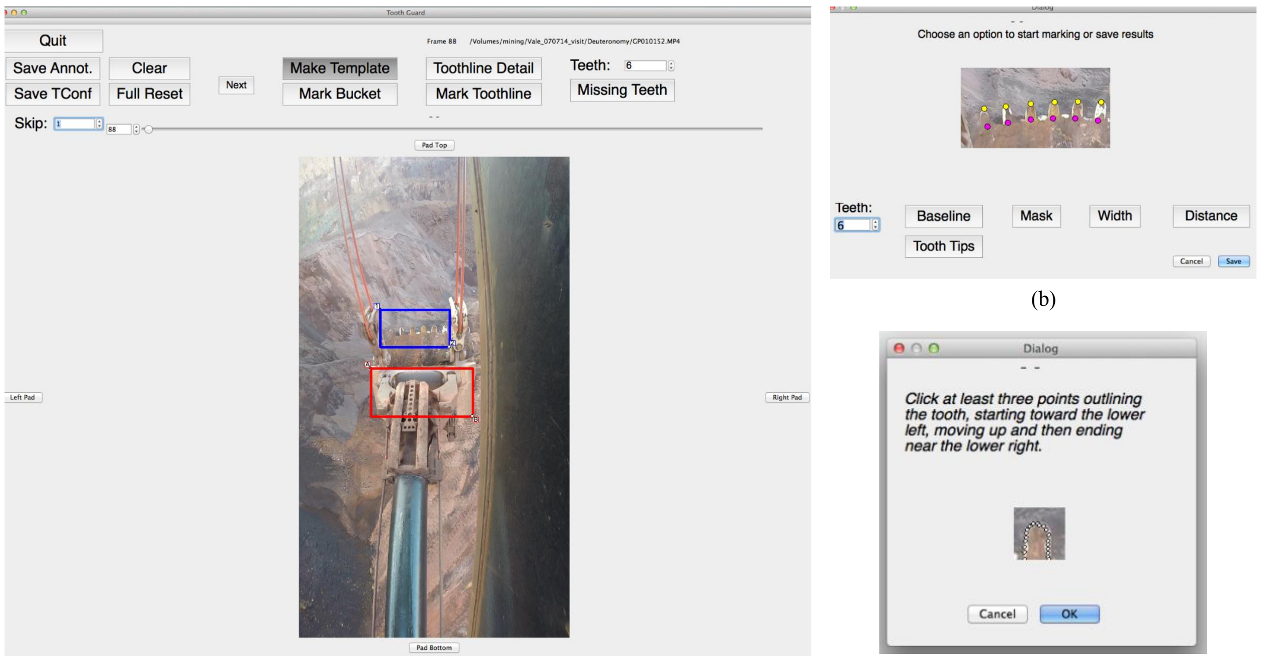


Figure 6. A frame rejection scheme is devised to reject bad tooth line detection results which significantly brings down the false positive rates. It is based on the fact that the difference image of a good detection exhibits unique appearance: the portion below the tooth tips is relatively dark while the portion above is white.



Choose an option to start marking or save results

Click at least three points outlining the tooth, starting toward the lower left, moving up and then ending near the lower right.

(a) (b) (c)

Figure 7. The the annotation tool for efficiently annotate the exemplar set $\mathbb{E}$ for the deployment of tooth guard system. (a) The annotated bucket exemplar (rectangle in red color) and tooth line region (rectangle in blue color). (b) The user interface for annotating the detailed tooth line information, including the baseline of the teeth (purple dots), width of the tooth line, distance between two teeth, and tooth tips (yellow dots). (c) The associated tooth template image generated by a number of user marked dots for outlining the tooth shape.

detection both achieve 0.95+ detection rates with a overlap ratio of 0.8. The accurate bucket and tooth line region detection serves a solid foundation for the tooth guard system to achieve high performance.

To evaluate the missing tooth detection performance, we randomly select 1,750 frames in the 4 video sequences and mark it as positive if one or more teeth are missing and negative, otherwise. We then vary the missing measurement threshold from 20 to 150 at a step of 10 to compute the receiver operating characteristic (ROC) [5] as the performance metric. Fig. 9 shows the ROC curves of three different thresholds for count resetting. It is seen that best performance was achieved when the threshold is 30 and the performance is quite robust to different values. In Fig. 10, we plot the ROC curves of three different missing count thresholds. Again, the missing tooth detection performance is quite robust to different settings.

Our system aims online operations, where the complete system is installed in a shovel which is on an operating production line. The entire algorithm pipeline can process 10 frames per second on a machine with 2.2GHz Intel i7 CPU and 16GB of RAM. The bottleneck is actually the display unit which has to been synchronized with the algorithm process. Given the image acquisition (30fps) step is faster than the algorithm (10fps), we have implemented a buffering method to ensure the two frames being fed to the missing tooth detection algorithm are not far away from each other.

Beyond the performance assessment based on collected videos, we have installed a unit in the iron mining site for testing. The system is running 24 hours a day, 7 days a week, and it have successfully detected 6 missing tooth events and missed 1 true missing tooth event in a 2-month trial. The false alarm rate is as low as 10 times per day on average.

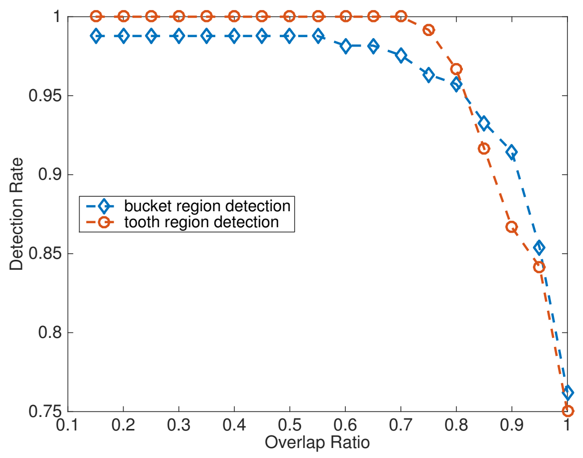


Figure 8. The bucket and tooth region detection performance. The overlapping rate is computed by dividing the area of detected bounding box to that of the ground truth bounding box annotated by human. Detection rate is the ratio of frames whose bucket and tooth line have been correctly detected.

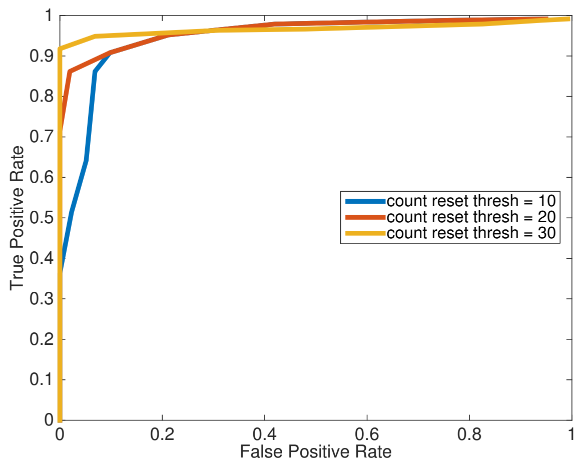


Figure 9. The ROC curves of missing tooth detection with different count reset thresholds. The missing count threshold was 10 all three trials.

7. Conclusion

In this paper, we have presented a solution purely based on compute vision to monitor tooth condition and detect missing tooth for a mining shovel. Our algorithm makes use two adjacent video frames. We first effectively and efficiently detected the tooth line regions by leveraging the predictable range of motion that a bucket of a rope shovel goes through during operation due to the camera mounting. The image of the tooth line region on the first frame was then rectified with the one on the second frame and a difference image is computed in the image gradient space. An normalized cross correlation was performed between a tooth template and the difference image so that a response map was generated. The final tooth positions was located by fitting a tooth line in the response map. A tooth was

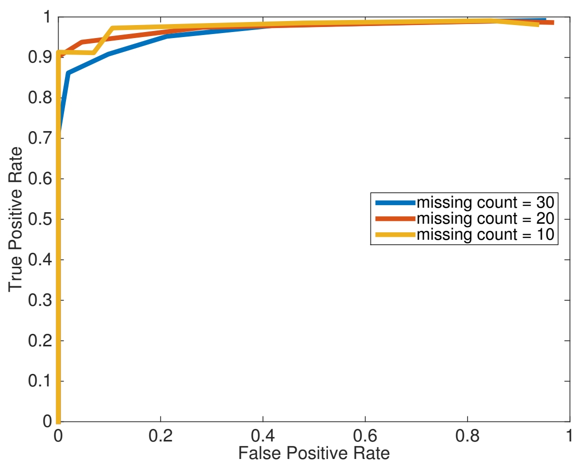


Figure 10. The ROC curves of missing tooth detection with different missing count thresholds. The missing count reset threshold was 30 in all three trials.

detected as missing if its missing measurement was exceeding a threshold at the current frame which was accumulated to fire an alarm if evidence is strong enough. Our experiments based on video sequences collected from an iron ore mine have demonstrated that the outstanding performance and high reliability of our tooth guard system.

References

[1] K. Briechle and U. D. Hanebeck. Template matching using fast normalized cross correlation. *Proc. SPIE*, 4387:95–102, 2001.

[2] R. Brunelli. *Template Matching Techniques in Computer Vision: Theory and Practice*. Wiley Publishing, 2009.

[3] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, volume 1, pages 886–893 vol. 1, June 2005.

[4] M. Everingham, L. van Gool, C. Williams, J. Winn, and A. Zisserman. The pascal visual object classes (voc) challenge. *International Journal of Computer Vision*, 88(2):303–338, 2010.

[5] T. Fawcett. An introduction to {ROC} analysis. *Pattern Recognition Letters*, 27(8):861 – 874, 2006.

[6] B. D. Lucas and T. Kanade. An iterative image registration technique with an application to stereo vision. In *Proc. of the 7th International Joint Conference on Artificial Intelligence*, pages 674–679, San Francisco, CA, USA, 1981.

[7] X. Luo and H. Zhang. Missing tooth detection with laser range sensing. In *World Congress on Intelligent Control and Automation*, volume 4, pages 3607–3610 Vol.4, June 2004.

[8] R. Ridley, T. Kazmierczak, L. Cai, P. Johnston, H. Pinto, J. Sun, M. Provencher, J. Woolley, and G. Cardinal. System and a method for detecting a damaged or missing machine part, Apr. 2 2013. US Patent 8,411,930.

[9] R. Szeliski. *Computer Vision: Algorithms and Applications*. Springer-Verlag New York, Inc., New York, NY, USA, 1st edition, 2010.