

Sensor Selection and Stage & Result Classifications for Automated Miniature Screwdriving

Xianyi Cheng, Zhenzhong Jia, Ankit Bhatia, Reuben M. Aronson, and Matthew T. Mason *Fellow, IEEE*

Abstract—Hundreds of billions of small screws are assembled in consumer electronics industry every year, yet reliably automating the screwdriving process remains one of the most challenging tasks. Two barriers to further adoption of robotic threaded fastening systems are system cost and technical challenges, especially for small screws. An affordable intelligent screwdriving system that can support online stage and result classification is the first step to bridge the gap. To this end, starting from a state transition graph of screwdriving processes and a labeled screwdriving dataset (1862 runs of M1.4 screws) on multiple sensor signals, we develop classification algorithms and perform sensor reduction. Fast and accurate result classifiers are developed using linear discriminant analysis, while a wrapper method for feature subset selection is used to identify the optimal feature subset and corresponding sensor signals to reduce cost. A stage classifier based on decision tree is developed using the optimal sensor subset. The stage classifier achieves high accuracy in realtime prediction of various stages when augmented with the state transition graph.

I. INTRODUCTION

Threaded fastening is one of the most commonly used methods in industrial assembly [1]. Around 1/4 to 1/3 of typical assembly operations can be classified as bolt and nut insertions [2] [3] [4]. Unfortunately, screwdriving remains one of the most difficult tasks to automate, despite substantial research in this field. One reason might be due to our incomplete understanding of the underlying process, particularly the initial mating step [5]. Our survey paper [1] summarizes various open problems and barriers that confront automated screwdriving systems. Four major improvements need to be made: (1) fast and reliable ways to feed screws with smaller length-to-diameter aspect ratios; (2) strategies for fast and reliable initial thread mating and early fault detection; (3) interactions of multiple objects (screw, driver bit, vacuum adapter, and target); (4) online failure prediction and fault recovery algorithms.

Automated screwdriving becomes even more challenging when it comes to the consumer electronics industry (e.g., laptops, tablets, and smartphones), where hundreds of billions of small screws ($\leq \#4$ or $\leq M3$) are assembled every year [6]. In fact, it is one of the most challenging operations that prevent manufacturing enterprises from further adopting automated robotic systems [6]. Many errors can occur during screwdriving. Even a very small fraction of assembly failures can cause serious consequences. For example, a loose screw can damage the battery inside a laptop, causing overheating and posing a fire hazard [7]. Small screws introduce

additional challenges and design considerations [8] [1]: (1) tighter tolerances for screw feeding and acquisition; (2) higher positioning accuracy and improved locating strategies for misalignment correction; (3) accurate and affordable screwdrivers with online fault prediction and recovery

Automated screw fastening involves multiple steps, often including screw feeding and acquisition, alignment, screwdriving, and post-fastening steps [1]. Comprehensive reviews of threaded fastening, including theoretical fundamentals, tools, control strategies, failure detection and industrial applications, can be found in [9], [10], and [1]. Among various steps, most works focus on the screwdriving process, where a properly acquired screw is driven into the target hole. One way to understand the screwdriving process is by plotting applied torque against the total rotation angle to produce the *torque-angle curve* [1]. This curve has been used for ISO rotary tool evaluation standards [11], control strategies, and failure detection [12]. The screwdriving process can be further divided into three major sub-steps: *initial thread mating*, *rundown*, and *tightening* with some variations or extra steps for self-tapping screws [1].

While most of the literature focuses on big screws (see Section II), this paper focuses on the miniature screwdriving process. Besides aforementioned technical challenges and reliability issues, one factor that prevents further adoptions of automated screwdriving systems is cost, as will be discussed in Section II. Hence, one goal of this paper is to select proper sensor signals to produce affordable intelligent screwdrivers that can be deployed to the actual assembly line for consumer electronics products. The screwdriver should perform both online result (see Section IV) and stage (see Section V) classifications, the very foundations to build fault prediction and recovery system for reliable screwdriving [12].

To summarize, our contributions are:

- A systematic way to identify optimal sensor readings to produce low-cost intelligent screwdriving system.
- Fast algorithms for online stage and result classifications for threaded fastening. To our knowledge, this is the first attempt for screwdriving stage classification.
- Compared with the literature, our evaluations are performed on a much larger screwdriving dataset with rich result and stage information.

II. RELATED WORK

One can refer to our survey paper [1] for a complete review of automated threaded fastening. In the following, we briefly describe some work that are directly related to this paper.

The authors are with the Robotics Institute, Carnegie Mellon University, Pittsburgh, PA 15213, USA. {xianyic, zhenzjia, ankitb, rmaronson, mattmason}@cmu.edu

A. Sensor Selection for Screwdriving Systems

In previous screwdriving literature, the sensor selection and control strategies are mainly empirical, i.e., based on years of engineering practice and expert experiences. For example, many COTS (commercial off-the-shelf) screwdrivers [13] are equipped with sensors to measure or estimate the fastening torque and rotational angle, because these two parameters can provide valuable information for screwdriving, especially for fault detection. Many control strategies also involve in monitoring the torque and/or angle information. These strategies can be divided into three broad categories: *torque-only control*, *torque-angle monitoring and control*, and *torque-rate control* [10] [1].

However, one problem associated with these COTS systems is that their sensor signals (torque and angle) may not be rich enough for controls and fault detection in robotic screwdriving. For example, robust detection of *initial mating* is critical for *torque-angle control* because it triggers the clamping angle count — an important parameter for quality control. Many high-end COTS screwdrivers (e.g., [13]) compare the driving torque against user-specified threshold to detect the initial engagement. However, as shown in Fig.2, the driving torque (T_z) is not a robust feature because it is almost constant during *initial mating*. In comparison, the insertion force (F_z) might be a much better candidate. Another limitation is that fault detection algorithms based on torque-angle signals alone cannot detect less-common failure modes, as shown in [12] [14].

There are some robotic screwdriving systems that can provide richer sensor readings. In [15], the insertion force (F_z) can be calculated by measuring the spring displacement. In [16] and [12], 6-axis force/torque (F/T) sensors are used to provide much richer information, at the expense of significant increase in system cost. In fact, besides reliability issues, cost (especially the sensor cost) is another factor that prevents further adoption of robotic screwdriving. For example, high-end screwdrivers [13] can cost more than \$10,000, because a typical 6-axis F/T sensor [17] cost around \$7,000 or even more. In this paper, starting from a much richer dataset, we follow a systematic feature reduction approach to identify the optimal subset of sensors for both online stage and result classifications (the building blocks for fault prediction and recovery system) to minimize system cost.

B. Fault Detection and Quality Monitoring

Reliable fault detection and error recovery are required by autonomous screwdriving, because even well-engineering systems can be tripped up by factors like part tolerance issues, bad material, and tool wear. One of the most commonly used method is the *teaching method*, in which faults typically show up as major deviations from the correct torque-angle *fastening signature* curve. To overcome the inflexibility and generality issues of the *teaching method*, fault diagnosis methods based on artificial intelligence, soft computing, model-based fault detection, and fuzzy systems have also been developed [1]. For example, artificial neural networks (ANNs) and support vector machines (SVMs) have been

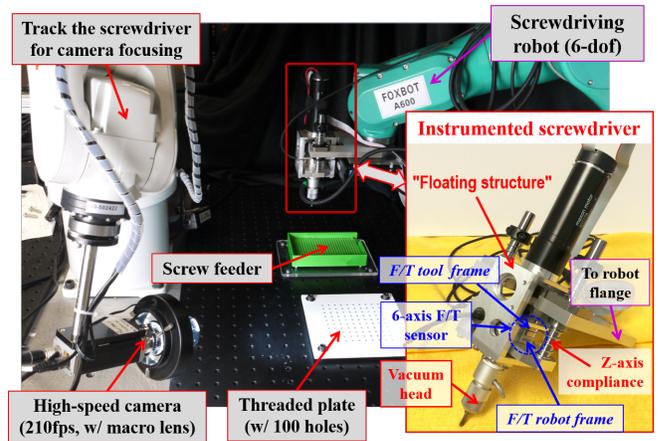


Fig. 1. Screwdriving experiment setup and the instrumented screwdriver.

investigated in [16]. Most methods mentioned above can achieve around 90% or better accuracy in detecting different failure modes. However, many things need to be improved before deploying to the actual assembly line.

Fault detection alone cannot satisfy the stringent requirements for high-volume production [1] in consumer electronics industry. To improve the overall success rate, fault prediction and recovery algorithms that can detect preceding failures earlier and take proper correcting actions upon predicted failure types are desired for future assembly lines [1] [12]; unfortunately, they are still missing in the literature. Our stage classification algorithm developed in Section V is the first step to bridge the gap.

III. DATA COLLECTION AND ANALYSIS

Fig. 1 shows the robotic screwdriving system we build for data collection. An instrumented screwdriver is installed on a 6-dof industrial robot to perform the following operations repetitively: picking up screws (M1.4 × 4 Phillips-head screws for cellphones) from the feeder (shaker tray) using vacuum suction, moving towards the threaded plate, aligning with the precalibrated target, and inserting screws into the target holes. Each run terminates when either the motor current or motor angle reaches a specified threshold. The system then proceeds to the next run. Another robot holds a high-speed camera with manual-focusing macro lens to record the operation.

A 6-axis force/torque (F/T) sensor (model: ATI mini-40, calibration: SI-20-1) is integrated into the screwdriver. The robot frame of the F/T sensor is connected to the robot flange through a linear compliance unit, while other parts of the screwdriver, including the motor, vacuum head, and the driver-bit, are essentially connected to the F/T tool frame. Compared with the wrist-mount F/T design in [16], our “floating structure” design is more suitable for miniature screwdriving because the F/T sensor is much closer to the screw and thus can provide more accurate data. Detailed descriptions and discussions of our data collection system and experiment procedures can be found in [12] [14].

In [12], we collected a total of 1862 screwdriving runs, each of which consists of 6-axis force and torque (see Fig. 2),

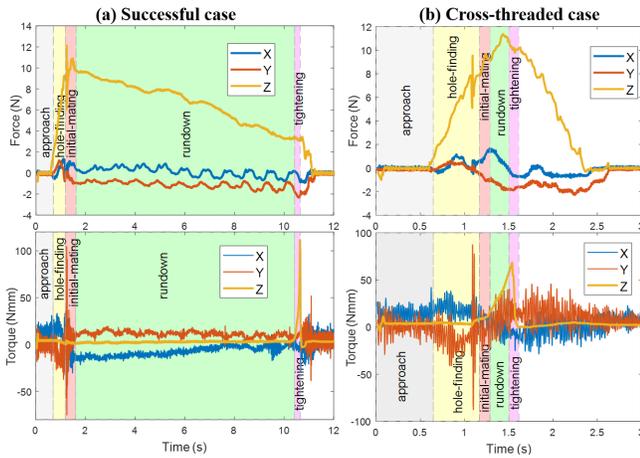


Fig. 2. The 6-axis force and torque signatures of (a) successful case with *hole finding* stage and (b) an unsuccessful (*crossthread*) case.

motor current and speed, and video data (see Fig. 3), all sampled at 100 Hz. After data analysis, we empirically came up with a list of stages (through which the screwdriving operation progresses) and result classes, as shown in Fig. 3. A complete description of the stages and result classes can be found in [12]. Our dataset has also been hand-labeled with corresponding stage (see Fig. 2) and result information for each run, forming the largest screwdriving dataset we know of [12]. All of our subsequent work is performed on this dataset.

Our result classes contain outcomes common in the literature and less-studied cases, such as those related to stripped screws. The stages shown in Fig. 3, though, represent the first division of screwdriving into stages, to our knowledge. As discussed in [12], “the stages provide a deeper understanding of the underlying operation, which can be applied to identify process failures that would be missed through simple result classification.” For example, the *hole finding* stage, defined as “screw has touched plate but not yet fallen into hole” [12], might damage crucial parts (e.g., PCB boards) in electronics products. But this stage cannot be detected by a result classifier. On the other hand, a complete list of stages that the operation passes is sufficient to predict the result. Moreover, as explained in [12], it is possible to build a failure prediction and recovery system based on the stage information, thereby improving the overall performance of the assembly line.

IV. RESULT TYPE CLASSIFICATIONS

Classification of the result types is important to fully automated screwdriving system [16]. This helps to identify the cause of failures. And based on correct predictions, proper control/recovery strategies can be developed to improve the performance. Meanwhile, we should choose affordable sensors instead of expensive ones (e.g., 6-axis F/T sensor) to develop low-cost systems that can be deployed to the assembly line.

To this end, fast and accurate result classification is performed by linear discriminant analysis (LDA) model [18] in Section IV-B. A wrapper method for feature subset selection

[19] is used to reduce the number of required features and sensor signals in Section IV-C. Finally, reduced feature subsets (see Table I and Table II) from less sensor signals are selected to produce a highly accurate yet affordable robotic screwdriving system.

A. Preprocessing of Time Series Data

In Section III, we collected and labeled multivariate time series data. In this paper, data from force-torque sensor and motor is used for classification. The video data (see Fig. 3) only serves as the ground truth. In our dataset, the time length of data varies among different runs. Classifying them can be tricky and computationally expensive [20], thus we use some global statistic characteristics of the captured data as features to perform result classification.

For each screwdriving run, 85 global features are extracted from the whole length of time series data from full sensor signals (8 channels in total). The angle feature (θ) is provided by the last motor encoder reading, which is commonly used in industry as an important criterion for failure detection, as discussed in Section II-B. The other 84 features are based on signals from the 6-axis F/T sensor and motor current, i.e., 7 channels in total. For each channel, 12 statistic features are extracted, including the range, mean and standard deviation of the following time series data: the original data, its first order differences (Δ), its second order differences (Δ^2), and its successive ratio (Δdd). Finally, each feature is normalized to have a standard normal distribution.

B. Linear Discriminant Analysis for Result Classification

Given 1862 labeled samples and 85 normalized features, a multi-class linear discriminant analysis (LDA) model is applied to predict the result classes shown in Fig. 3. The classification result is also used as a benchmark for the sensor reduction work described in Section IV-C.

Multi-class LDA is a simple but powerful classification algorithm that can be used to separate multiple classes. The LDA has the advantages of giving linear decision boundaries and requiring less computations. In this method, a set of observation x is classified as the class \hat{y} that has the largest posterior probability $P(Y = y|X = x)$ among all K classes. By Bayes’ theorem, this can be written as

$$\hat{y} = \arg \max_{y=1, \dots, K} P(X = x|Y = y)P(Y = y)$$

where $P(X = x|Y = y)$ is probability density function of x and $P(Y = y)$ is prior probability of class y . For density functions of all the classes, LDA assumes that they are normally distributed with the same covariance Σ but different means μ_1, \dots, μ_K , where μ_1, \dots, μ_K , and Σ can be directly estimated from the training data. By maximizing the posterior probability, a linear decision boundary can be found for each pair of the classes.

With 85 global features, this multi-class LDA method achieves an average classification accuracy of 98.93% in 10-fold cross-validation. In comparison, our previous GTC-DF model [12] (Graph of Temporal Constraint Decision Forest)

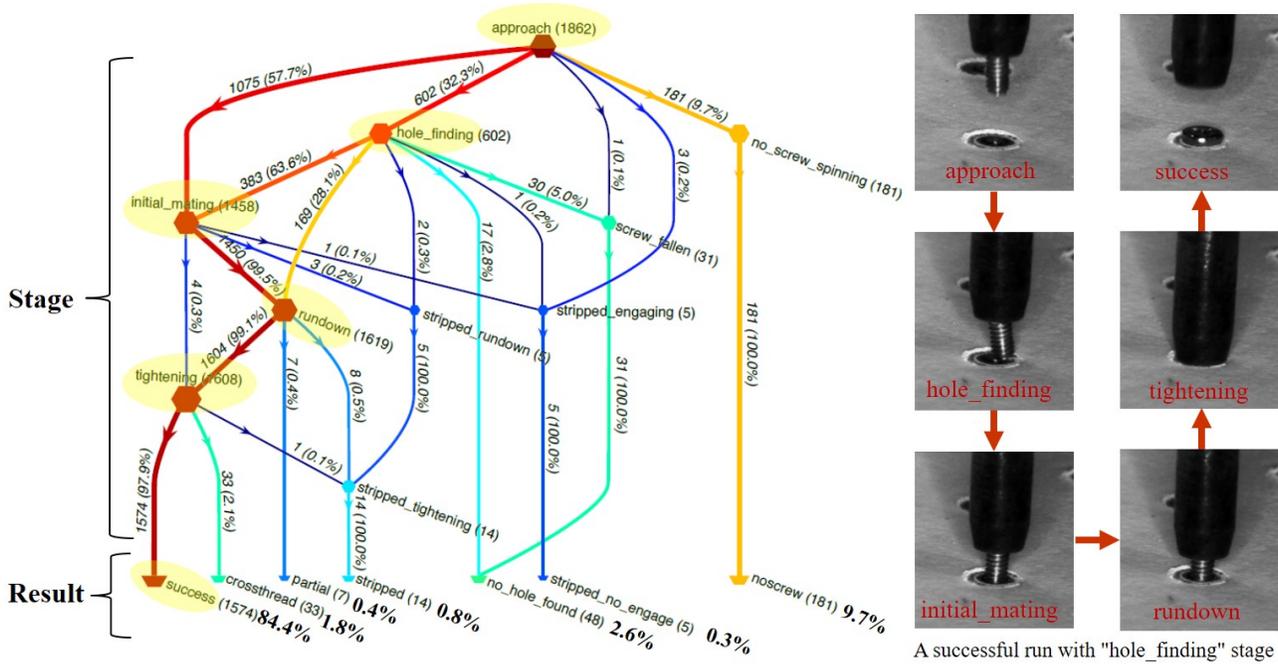


Fig. 3. Left: The state transition graph summarizing all the stage and result classes. The vertices represent stages through which the screwdriving passes, with the terminal stages corresponding to the result classes. Colors and sizes are scaled logarithmically with the number of runs in each transition, stage, or result (modified from [12]). Right: A successful screwdriving run example including a *hole finding* stage.

	1572	0	0	1	0	0	0	99.9%
success	84.4%	0.0%	0.0%	0.1%	0.0%	0.0%	0.0%	0.1%
no screw	1	179	1	0	0	0	0	98.9%
no hole found	0.1%	9.6%	0.1%	0.0%	0.0%	0.0%	0.0%	1.1%
crossthread	0	1	37	0	0	1	2	90.2%
stripped	0.0%	0.1%	2.0%	0.0%	0.0%	0.1%	0.1%	9.8%
stripped no engage	1	1	0	32	0	0	0	94.1%
partial	0.1%	0.1%	0.0%	1.7%	0.0%	0.0%	0.0%	5.9%
success	0	0	4	0	13	2	0	68.4%
no screw	0.0%	0.0%	0.2%	0.0%	0.7%	0.1%	0.0%	31.6%
no hole found	0	0	3	0	1	2	0	33.3%
crossthread	0.0%	0.0%	0.2%	0.0%	0.1%	0.1%	0.0%	66.7%
stripped	0	0	3	0	0	0	5	62.5%
stripped no engage	0.0%	0.0%	0.2%	0.0%	0.0%	0.0%	0.3%	37.5%
partial	99.9%	98.9%	77.1%	97.0%	92.9%	40.0%	71.4%	98.8%
success	0.1%	1.1%	22.9%	3.0%	7.1%	60.0%	28.6%	1.2%

Fig. 4. Confusion matrix for the LDA model trained with 85 features.

trained on the same dataset with 144 features achieves a 99.03% accuracy. But this LDA model is much simpler than the GTC model. The confusion matrix of the LDA model is shown in Fig. 4. We see that the LDA method achieves high accuracy in predicting *success*, *no screw*, *crossthread*, and *stripped* result classes. However, its performance is not ideal for the other result classes. For *stripped no engage* and *partial*, the reason is that we have very few runs from these classes. For the *no hole found* class, there are several types of signals that are quite similar to those of the other classes, as shown in Fig. 5 and Fig. 6. The overall trends are similar, while there exist local differences such as different

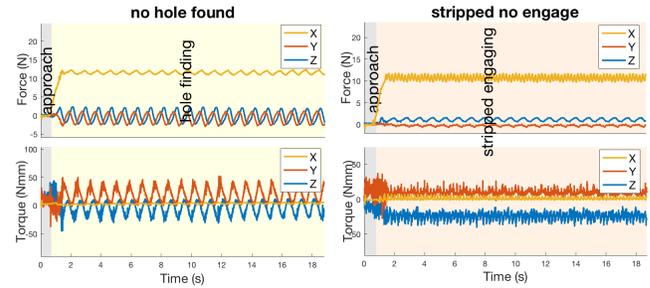


Fig. 5. Similar F/T signals might cause a misclassification between a *no hole found* run and a *stripped no engage* run.

oscillation periods. Since the result classifiers are trained with global features, some meaningful local characteristics might be ignored. To improve the performance, more data for the rare failure types and better understanding of *no hole found* are required.

The results in this part can serve as the baseline for developing more complex algorithms. Note that other classical machine learning methods are also applicable for the result classification in this paper, such as logistic regression, quadratic discriminant analysis and SVMs. All these classifiers yield similar results, except that the accuracy of the LDA is slightly better.

C. Sensor Reduction through Feature Selection

In this part, sensor reduction for more economical screwdriving system is achieved by feature reduction. We follow a standard feature selection approach to reduce the number of required features and sensor signals. Feature selection requires a search algorithm to select candidate feature subsets and an objective function to evaluate these candidates [21].

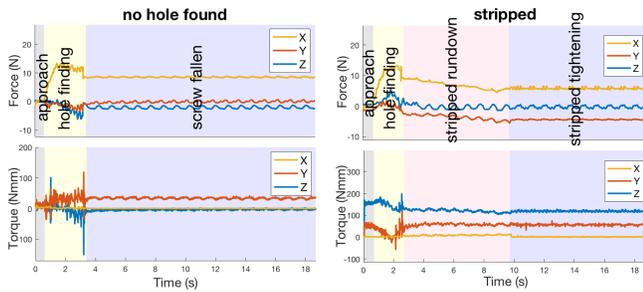


Fig. 6. Similar F/T signals might cause a misclassification between a *no hole found* run and a *stripped* run.

In our study, the sequential backward selection (SBS) [19] is chosen as the search algorithm. Starting from the full feature set, SBS sequentially removes the feature that least reduces the value of the objective function. SBS provides a systematic way to sort the original features according to their importance. Note that alternatives such as bidirectional search might give better results; SBS is chosen here due to its simplicity. For subset evaluation, we use the wrapper approach. In the wrapper approach, a feature subset is evaluated by the performance of the chosen learning algorithm. In this paper, for a given feature subset, a multi-class LDA model is trained as in IV-B. Based on the LDA model, the average 10-fold cross-validation accuracy is used as the criterion for evaluation. The overall feature reduction algorithm is as follows:

- 1) Start with the full feature set $X_0 = \{x_1, x_2, \dots, x_N\}$.
- 2) For every feature $x_i \in X_k$, train a LDA model on $X_k - \{x_i\}$.
- 3) Remove the feature $\hat{x} = \operatorname{argmax}_x J(X_k - \{x\})$, where J is the 10-fold classification accuracy.
- 4) Update the feature set as $X_{k+1} = X_k - \{\hat{x}\}$.
- 5) Goto step 2 and repeat until meeting a certain stop condition (the accuracy and the number of remaining features meet user-specified requirements).

This algorithm helps to determine the optimal feature subset and provides a sensor selection guideline to reduce cost for the final implementation. In addition, fewer features result in reduced complexity and faster algorithm.

To show the whole picture, we set the stopping condition to be $k = N - 1$, i.e., only one feature is left in the final subset. Fig. 7 shows the 10-fold cross-validation accuracy of result classification for each optimal subset. The curve is non-monotonic; the accuracy initially increases when some features are removed and drops significantly when very few features are left. Fig. 8 shows the variation of the optimal feature set (grouped according to the sensor signals) during the reduction process. We see that most features from F_x and F_y are removed at the first few steps, while features from M_c , F_z and T_z can still achieve quite good accuracy when most features from other signals are removed. The angle feature θ remains during the entire reduction process (not shown in the figure); this explains the importance of angle measurement.

This feature selection process indicates that, instead of using all the signals, accurate result type classification can

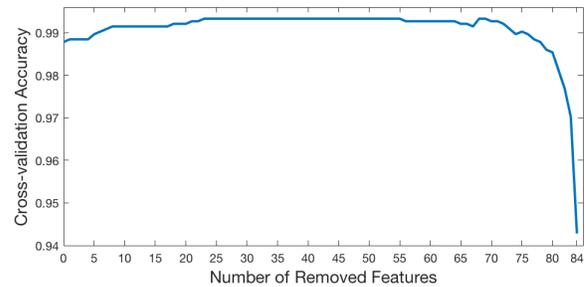


Fig. 7. Prediction accuracy for optimal feature subsets of different sizes.

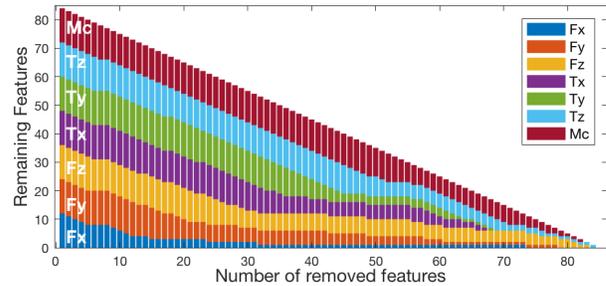


Fig. 8. Stacked bar graph during the feature reduction process.

be achieved with only four signals: M_c , F_z , T_z , and θ (angle). Note that we do not need both M_c and T_z . In fact, we can further reduce the required sensor signals and cost based on the gear ratio. For low gear ratios (5.8:1 in our system), M_c is almost proportional to T_z , thus we can select M_c (motor current) because it costs less to measure. For high gear ratios, it is necessary to measure T_z because the fastening torque cannot be estimated from M_c due to high gear loss.

For our system, we can remove T_z and reduce the signals to $\{M_c, F_z, \theta\}$. To evaluate the performance, we train two LDA classifiers on the selected signals with and without T_z . We obtain an optimal feature subset consisting of 20 features for $\{T_z, M_c, F_z, \theta\}$ signals and 18 features for $\{M_c, F_z, \theta\}$ signals, as shown in Table I and Table II, respectively. As shown in Fig. 9, even with fewer sensor signals and features, both classifiers achieve very high accuracy (99.03% and 98.66%, respectively). Their performance is quite similar to the baseline case shown in Fig. 4, where 85 global features are used. Compared with the baseline, the reduced LDA models perform better in predicting the *partial* and *stripped no engage* classes; they perform slightly worse for the *no hole found* class.

We see that a linear classifier based on reduced sensor signals can successfully predict the results of screwdriving tasks with very high accuracy. This method can be viewed as an extension of current industrial method, which only uses the maximum torque and angle to predict success or failure.

V. STAGE CLASSIFICATIONS

As discussed in Section III, identifying stages in screwdriving tasks is the first step towards building online failure prediction and recovery for advanced robotic screwdriving system. This is particular important for large-volume productions in consumer electronics industry [1]. In this section, we train decision tree models [22] to discriminate different stages during screwdriving process. The reason of using

Predicted Class	approach	10060	70	98	0	0	26	0	0	0	0	0	0	98.1%
		6.3%	0.0%	0.1%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	1.9%
	hole finding	50	6431	497	89	4	35	46	18	6	10	89.5%		
		0.0%	4.0%	0.3%	0.1%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	10.5%		
	initial mating	119	478	5335	41.9	3	27	21	16	5	14	82.9%		
		0.1%	0.3%	3.3%	0.3%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	17.1%		
	rundown	0	157	439	102653	166	231	74	6	40	37	98.9%		
		0.0%	0.1%	0.3%	64.2%	0.1%	0.1%	0.0%	0.0%	0.0%	0.0%	1.1%		
	tightening	0	3	5	134	2332	19	1	0	1	2	93.4%		
		0.0%	0.0%	0.0%	0.1%	1.5%	0.0%	0.0%	0.0%	0.0%	0.0%	6.6%		
	no screw spinning	34	30	22	241	10	24334	11	0	0	5	98.6%		
		0.0%	0.0%	0.0%	0.2%	0.0%	15.2%	0.0%	0.0%	0.0%	0.0%	1.4%		
	screw fallen	3	47	16	45	0	2	3217	0	1	3	96.5%		
		0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	2.0%	0.0%	0.0%	0.0%	3.5%		
stripped engaging	0	15	6	10	0	0	0	568	3	0	94.4%			
	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.4%	0.0%	0.0%	5.6%			
stripped rundown	0	3	3	31	0	0	1	0	307	8	87.0%			
	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.2%	0.0%	13.0%			
stripped tightening	0	11	5	42	4	7	5	0	20	754	88.9%			
	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.5%	11.1%			
	98.0%	88.8%	83.0%	99.0%	92.6%	98.6%	95.3%	93.4%	80.2%	90.5%	97.5%			
	2.0%	11.2%	17.0%	1.0%	7.4%	1.4%	4.7%	6.6%	19.8%	9.5%	2.5%			
Actual Class	approach	hole finding	initial mating	rundown	tightening	no screw spinning	screw fallen	stripped engaging	stripped rundown	stripped tightening				

Fig. 10. Stage classification confusion matrix for the full sensor set.

Predicted Class	approach	10060	21	37	0	1	23	0	0	0	1	99.2%
		6.3%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.8%
	hole finding	21	6819	308	95	3	13	39	29	11	8	92.8%
		0.0%	4.3%	0.2%	0.1%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	7.2%
	initial mating	43	288	5865	145	3	8	30	9	11	4	91.6%
		0.0%	0.2%	3.7%	0.1%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	8.4%
	rundown	1	100	141	102328	58	403	370	0	78	25	98.9%
		0.0%	0.1%	0.1%	64.0%	0.0%	0.3%	0.2%	0.0%	0.0%	0.0%	1.1%
	tightening	2	4	3	63	2497	11	4	0	0	5	96.4%
		0.0%	0.0%	0.0%	0.0%	1.6%	0.0%	0.0%	0.0%	0.0%	0.0%	3.6%
	no screw spinning	13	13	7	417	20	24458	68	1	0	7	97.8%
		0.0%	0.0%	0.0%	0.3%	0.0%	15.3%	0.0%	0.0%	0.0%	0.0%	2.2%
	screw fallen	0	28	37	294	1	37	2868	1	8	9	87.4%
		0.0%	0.0%	0.0%	0.2%	0.0%	0.0%	1.8%	0.0%	0.0%	0.0%	12.6%
stripped engaging	0	18	7	1	0	0	1	538	0	0	95.2%	
	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.3%	0.0%	0.0%	4.8%	
stripped rundown	0	3	8	59	0	0	0	0	239	1	75.2%	
	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.1%	0.0%	24.8%	
stripped tightening	0	4	4	21	1	1	7	0	5	800	94.9%	
	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.5%	5.1%	
	99.2%	93.4%	91.4%	98.9%	96.6%	98.0%	84.5%	93.1%	67.9%	93.0%	97.8%	
	0.8%	6.6%	8.6%	1.1%	3.4%	2.0%	15.5%	6.9%	32.1%	7.0%	2.2%	
Actual Class	approach	hole finding	initial mating	rundown	tightening	no screw spinning	screw fallen	stripped engaging	stripped rundown	stripped tightening		

Fig. 11. Stage classification confusion matrix for $\{M_c, F_z, T_z\}$.

good performance in predicting the overall trends for stages. There exist some stage misclassifications; most of them are either insignificant or correctable.

The first type of misclassification appears at the transition of two stages. This will not be considered as a fault prediction, since a transition can be classified as either of its neighboring stages. For example, as shown in Fig. 12, the transition from *hole finding* to *initial mating* is classified as the latter stage, this should also be considered as correctly classified. Note that our algorithm generates a misclassification, a *initial mating* stage at the end, because T_z almost equals to zero in this period. This misclassification actually belongs to the second type (see below).

The second type of misclassifications are those can be corrected by the stage transition graph (see Fig. 3). For example, the *screw fallen* stage in Fig. 13 (a) is not connected with the *initial mating* stage in Fig. 3, so this misclassifi-

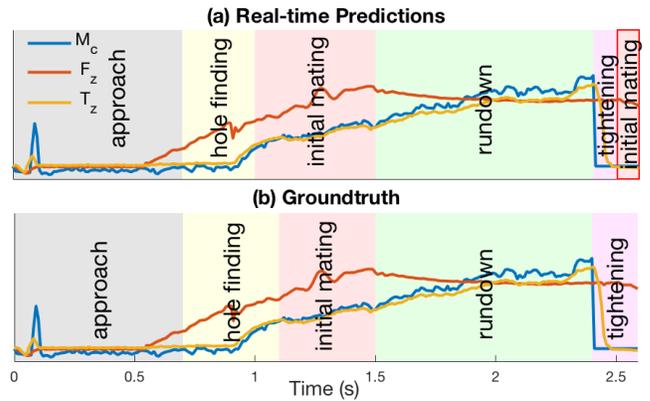


Fig. 12. Realtime stage prediction for a *crossthread* run. The misclassified *initial mating* stage is highlighted by a red block.

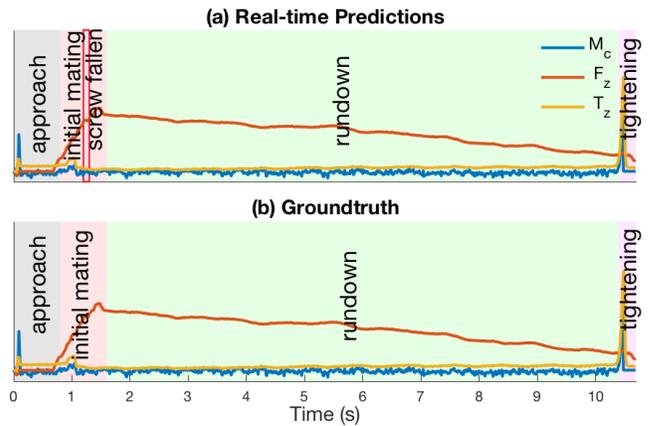


Fig. 13. Realtime stage prediction for a *success* run. The misclassified *initial mating* stage is highlighted a red block.

cation can be corrected by the stage transition graph. The corrected result will match the actual classification shown in Fig. 13 (b). Fig. 14 (b) shows the stage prediction result after augmenting with the stage transition graph. Compared with Fig. 14 (a), there is only one stage misclassification — the *tightening* stage that only lasts 0.1s right after *rundown*. Close examination reveals that our algorithm actually works very well because T_z rapidly increases over this 0.1s window right after the *rundown* phase. A reasonable classifier will predict this 0.1s window as *tightening* stage. However, a human cannot do such precise classification (this may not be necessary) during hand labeling. Instead, a human simply labels a *stripped tightening* right after *rundown* by considering the entire signal profile, including future signals.

VI. DISCUSSIONS AND FUTURE WORK

The stage and result classification algorithms we developed are essentially data-driven, which requires large amounts of training data. This data-driven approach suffers from generality issues. There are many different types of screws and nuts used in consumer electronics industry. The force and torque signatures will change when the part (e.g., buttons and PCBs) to be bolted varies in geometry or material. Therefore, one would need to rerun the experiments for data collection once the customers modify their product

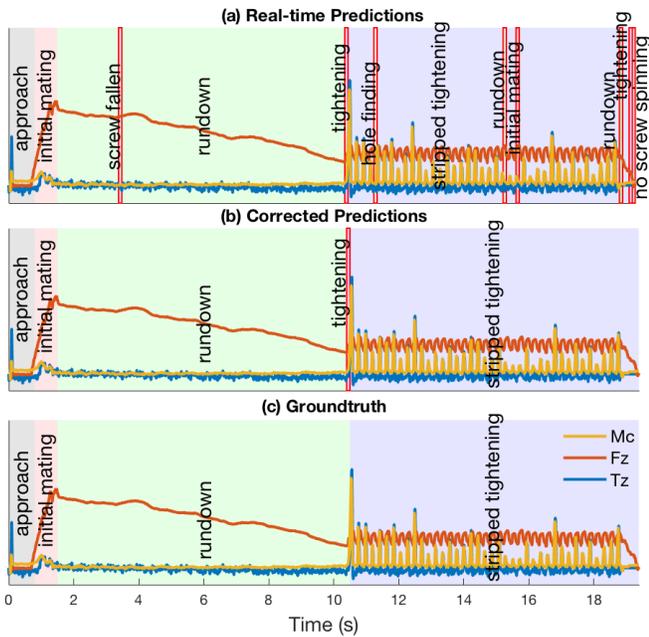


Fig. 14. Realtime stage prediction for a *stripped* run. Misclassified stages are highlighted by red blocks.

designs. Although the retraining process can be quite fast using our algorithms, the data collection process takes time and resources. This process might be worthwhile for very large volume production. However, it is always beneficial to investigate techniques (through modeling or simulation) that can significantly reduce the data size requirement. This can be quite difficult since exact modeling of the interactions and contact forces between different objects and among the screw threads is very challenging.

There is some information which could be inferred from the removed sensor signals. For example, the authors of [16] essentially use the auto-correlation function of $\{F_x, F_y\}$ to estimate the insertion length. (Note that they actually use the auto-correlation of T_z ; however, there is a large offset between the F/T z-axis and the center axis of the screwdriver in their setup, so T_z is dominated by F_x and F_y .) However, this insertion length can be easily measured by an encoder — an approach adopted by industry. In our study, we find that the oscillating amplitude of the $\{F_x, F_y\}$ (equivalent to $\{T_x, T_y\}$ after transformation) is strongly correlated with the positioning error. Therefore, one could estimate the misalignment using these signals. However, this will significantly increase the system cost and complexity.

In the future, a low-cost screwdriver will be designed based on our sensor reduction analysis. An online stage and result prediction system will be developed through faster and improved implementation of our algorithms. Recovering strategies corresponding to different failure types will also be designed to improve the overall success rate.

ACKNOWLEDGMENTS

The authors would like to thank Foxconn for providing the financial support. The views and opinions expressed in this paper do not necessarily state or reflect those of Foxconn.

REFERENCES

- [1] Z. Jia, A. Bhatia, R. Aronson, D. Bourne, and M. T. Mason, "A survey of automated threaded fastening," *IEEE Transactions on Automation Science and Engineering*, Conditionally accepted, 2018.
- [2] J. L. Nevins and D. E. Whitney, "Computer-controlled assembly," *Scientific American*, vol. 238, pp. 62–74, 1978.
- [3] J. Nevins and D. Whitney, "Assembly research," *Automatica*, vol. 16, no. 6, pp. 595–613, 1980.
- [4] L. A. Martin-Vega, H. K. Brown, W. H. Shaw, and T. J. Sanders, "Industrial perspective on research needs and opportunities in manufacturing assembly," *Journal of manufacturing systems*, vol. 14, no. 1, pp. 45–58, 1995.
- [5] D. E. Whitney, *Mechanical assemblies: their design, manufacture, and role in product development*. Oxford university press, 2004, vol. 1.
- [6] Z. Li. Robotics research for 3C assembly automation. [Online]. Available: <https://app.box.com/s/zcg8qxt6fw6v4xz22h6>
- [7] U. S. C. P. S. Commission. (2018) Lenovo recalls thinkpad laptops due to fire hazard. [Online]. Available: <https://www.epsc.gov/Recalls/2018/lenovo-recalls-thinkpad-laptops-due-to-fire-hazard>
- [8] A. Weber, "Automation for small screws," *Assembly*, vol. 55, no. 2, pp. 40–43, 2012.
- [9] J. H. Bickford, *Handbook of bolts and bolted joints*. CRC press, 1998.
- [10] —, *Introduction to the design and behavior of bolted joints: non-gasketed joints*. CRC Press, 2007.
- [11] ISO, "ISO 5393: Rotary tools for threaded fasteners – performance test method," ISO, Tech. Rep., 2013.
- [12] R. M. Aronson, A. Bhatia, Z. Jia, M. Guillaume-Bert, D. Bourne, A. Dubrawski, and M. T. Mason, "Data-driven classification of screwdriving operations," in *International Symposium on Experimental Robotics*. Springer, 2016, pp. 244–253.
- [13] *MicroTorque-ToolsTalk MT User Guide*, Atlas Copco.
- [14] R. M. Aronson, A. Bhatia, Z. Jia, and M. T. Mason, "Data collection for screwdriving," in *Robotics Science and Systems, Workshop on (Empirically) Data-Driven Manipulation*, 2017.
- [15] J.-Y. Hwang, D.-H. Jung, Y.-J. Roh, K.-J. Nam, and D.-Y. Hwang, "Low-cost automatic screw machine using a commercial electric screwdriver," in *Control, Automation and Systems (ICCAS), 2012 12th International Conference on*. IEEE, 2012, pp. 1055–1060.
- [16] T. Matsuno, J. Huang, and T. Fukuda, "Fault detection algorithm for external thread fastening by robotic manipulator using linear support vector machine classifier," in *Robotics and Automation (ICRA), 2013 IEEE International Conference on*. IEEE, 2013, pp. 3443–3450.
- [17] A. I. Automation, "Multi-axis force/torque sensors." [Online]. Available: <http://www.ati-ia.com/products/ft/sensors.aspx>
- [18] G. James, D. Witten, T. Hastie, and R. Tibshirani, *An introduction to statistical learning*. Springer, 2013, vol. 112.
- [19] H. Liu and H. Motoda, *Feature selection for knowledge discovery and data mining*. Springer Science & Business Media, 2012, vol. 454.
- [20] A. Nanopoulos, R. Alcock, and Y. Manolopoulos, "Feature-based classification of time-series data," *International Journal of Computer Research*, vol. 10, no. 3, pp. 49–61, 2001.
- [21] V. Kumar and S. Minz, "Feature selection," *SmartCR*, vol. 4, no. 3, pp. 211–229, 2014.
- [22] L. Breiman, J. Friedman, C. J. Stone, and R. A. Olshen, *Classification and regression trees*. CRC press, 1984.
- [23] L. Rokach and O. Maimon, "Top-down induction of decision trees classifiers-a survey," *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 35, no. 4, pp. 476–487, 2005.