Fault Recognition for Mechanical Arm by Using Relative Margin SVM

Dongzhe Yang, LiaoYuan Vocational Technical College, China*

ABSTRACT

Monitoring and detecting faults during the operation of the manipulator is the prerequisite for fault recognition and safe operation. Accurate classification of mechanical arm faults can support to effectively eliminate mechanical arm faults. In this paper, the authors utilize a relative margin support vector machine (RMSVM) to classify and monitor the faults for mechanical arm. First, the status of mechanical arm are represented a high dimensional vector which consists of the mean, variance, correlation coefficient of the residual momentum signal in time domain, and the wavelet packet energy spectrum in frequency domain. The collected feature vectors for mechanical arm status are used to train RMSVM. A virtual prototype of mechanical arm is used to analyze the changes in the features of the residual momentum caused by fault and evaluate the RMSVM model for future mechanical arm status. The simulation results show that RMSVM can effectively detect the faults during the operation of manipulator.

KEYWORDS

Fault Recognition, Feature Representation, Mechanicla Arm, Relative Margin Support Vector Machine, Wavelet

1. INTRODUCTION

Motor fault (Shao 2020, Chan 2020) is one of the common faults in the operation of the manipulator. The detection of motor fault in the operation of the manipulator is the premise to ensure safe operation (Shi 2016, Brkovic 2017). The accurate classification of mechanical arm faults can provide the support for effectively eliminating potential incidents during operating mechanical arms.

The previous mechanical arm fault detection methods include: comparing the actual actuator torque with the torque calculated by the model to detect whether the mechanical arm is faulty (Xu 2019, Ismail 2021); building a virtual prototype platform and studying the impact of collision parameters on collision of mechanical arms (Hua 2013, Ghaffari 2016); utilizing momentum derivative to design a virtual prototype for analyzing the change of the residual momentum during collision process (Shao 2015); modeling a rigid manipulator to express the faults of the mechanical arms (Santos 2010, Grabbe 1994) etc.

In order to further improve the fault recognition rate, the machine learning method is introduced to predict the mechanical arm fault in recent years, such as support vector machine (SVM) (Scholkopf 1997), Bayesian. In this paper, we adopt relative margin support vector machine (RMSVM) (Song

*Corresponding Author

2017, Zhu 2017) to predict the mechanical arm fault. First, the feature vector in frequency domain from residue momentum is combined with the feature vector in time domain to analyze the change between fault source and feature vectors; then the feature vectors are collected to construct the training set for learning relative margin support vector machine (RMSVM) model to recognize the status of mechanical arms; the training RMSVM model is tested on simulated platform to verify the effectiveness.

The remaining of the this paper is organized as follows. The feature extraction of the signals from mechanical arms is described in Section 2. The relative margin support vector machine is introduced in Section 3 for fault recognition in the operation of mechanical arms. The experiments and simulation of the fault recognition method is provided in Section 4 and Section 5. The last Section is the conclusion and discussion.

2. FEATURE EXTRACTION AND REPRESENTATION FOR RESIDUE MOMENTUM SYGNAL IN MECHANICAL ARMS

The basis to analyze the mechanical arm is the associated dynamic model which is represented as the following equation:

$$M(q)\overline{q} + C(q,\tilde{q})\tilde{q} + G(q,\tilde{q}) = \tau \tag{1}$$

In equation (1), $q = [q_1, q_2, q_3]^T$ is the joint vector of mechanical arm, $M(q)$ is the moment of inertia, $C(q,\tilde{q})$ is the Coriolis force and centrifugal force matrix, $G(q,\tilde{q})$ is the gravity matrix, and τ is the driving torque vector.

The residual momentum operator r is defined as following equation:

$$r = k[\int (\tau + \alpha - r)dt - p] \tag{2}$$

In equation (2), $\alpha = C^T(q)\tilde{q} - G(q)$, $p = M(q)\tilde{q}$, the magnification factor k is a diagonal matrix whole diagonal elements are greater than zero, and p is the total momentum of the robotic arm system.

When the robotic arm collides with the environment, the following equation holds:

$$M(q)\overline{q} + C(q)\tilde{q} + G(q) = \tau + \tau_j \tag{3}$$

In equation (3), τ_j is the torque which is generated when the robotic arm collides with the environment.

After derivation of equation (2), the dynamic $\tilde{r}$ satisfies the following equations:

$$\tilde{r} = -kr + k\tau_j, r(0) = 0 \tag{4}$$

Equation (4) shows that the change of the stable linear system is determined by the accidental collision force τ_j.

2.1 Time Domain Features

Feature extraction can reduce the difficulty of calculation and better represent the essence of robotic arm status, which can facilitate robotic arm fault recognition and classification. The residual momentum (Wang 2018, David 2011) is a one-dimensional time-varying signal. This paper adopts mean, variance, and correlation coefficient to construct time domain feature vector.

Here, the mean of residual momentum of three-free robotic arm is represented as following equation:

$$D = [D_1, D_2, D_3]$$ (5)

In equation (5), $D_1 = \frac{1}{n}\sum_{i=1}^{n} r_{1,i}$, $D_2 = \frac{1}{n}\sum_{i=1}^{n} r_{2,i}$, $D_3 = \frac{1}{n}\sum_{i=1}^{n} r_{3,i}$. Here, n is the number of collected samples.

The variance of residual momentum of three robotic arm is represented as following equation:

$$V = [V_1, V_2, V_3]$$ (6)

In equation (6), V_1 , V_2 , and V_3 are defined as following equations:

$$V_1 = \frac{1}{n-1}\sum_{i=1}^{n}(r_{1,i} - D_1)^2$$

$$V_2 = \frac{1}{n-1}\sum_{i=1}^{n}(r_{2,i} - D_2)^2$$

$$V_2 = \frac{1}{n-1}\sum_{i=1}^{n}(r_{3,i} - D_3)^2$$

The correlation coefficient represents the similarity between residual momentums. The correlation coefficient of three-free robotic arm is represented as following equation:

$$\rho = [\rho_{1,2}, \rho_{1,3}, \rho_{2,3}]$$ (7)

In equation (7), $\rho_{1,2}$, $\rho_{1,3}$, and $\rho_{2,3}$ are represented as following equations:

$$\rho_{1,2} = (\sum_{i=1}^{n}(r_{1,i} - V_1)(r_{2,i} - V_2))(\sum_{i=1}^{n}(r_{1,i} - V_1)^2(r_{2,i} - V_2)^2)^{-\frac{1}{2}}$$

$$\rho_{1,3} = (\sum_{i=1}^{n}(r_{1,i} - V_1)(r_{3,i} - V_3))(\sum_{i=1}^{n}(r_{1,i} - V_1)^2(r_{3,i} - V_3)^2)^{-\frac{1}{2}}$$

$$\rho_{2,3} = (\sum_{i=1}^{n}(r_{2,i} - V_2)(r_{3,i} - V_3))(\sum_{i=1}^{n}(r_{2,i} - V_2)^2(r_{3,i} - V_3)^2)^{-\frac{1}{2}}$$

2.2 Frequency Domain Features

The wavelet analysis (Wickerhauser 2010, Torrence 1998, Newland 2013) has the characteristic of local amplification. The paper adopts wavelet transform to process the residual momentum after normalization. After the decomposition of wavelet transform, the energy of the reconstructed signal $S_{k,j}$ for the jth band of the kth layer is represented as following equation:

$$E_{k,j} = \int |S_{k,j}(t)|^2 \, dt = \sum_{m=1}^{N} |r_{j,m}|^2 \tag{8}$$

In equation (8), N represent the length of the signal, $j = 1, 2, \ldots, 2^k$ represents the order of decomposed frequency, $r_{j,m}$ represents the amplitude of the discrete points in the reconstructed signal.

The number of decomposition layers is closely related to the cost of calculation. In order to facilitate the calculation, the number of decomposition layers is set as $k = 4$.

The following equation is the wavelet energy spectrum of the residual momentum for the three-free robotic arm:

$$E = [E_{r,1}, E_{r,2}, E_{r,3}] \tag{9}$$

In equation (9), $E_{r,1}$, $E_{r,2}$, and $E_{r,3}$ are defined as following equations:

$$E_{r,1} = (\sum_{j=1}^{n} E_{r,1}^{4,j})^{-\frac{1}{2}} [E_{r,1}^{4,1}, E_{r,1}^{4,2}, \ldots, E_{r,1}^{4,n}]$$

$$E_{r,2} = (\sum_{j=1}^{n} E_{r,2}^{4,j})^{-\frac{1}{2}} [E_{r,2}^{4,1}, E_{r,2}^{4,2}, \ldots, E_{r,2}^{4,n}]$$

$$E_{r,3} = (\sum_{j=1}^{n} E_{r,3}^{4,j})^{-\frac{1}{2}} [E_{r,3}^{4,1}, E_{r,3}^{4,2}, \ldots, E_{r,3}^{4,n}]$$

The number of coefficients for wavelet transform is $3 \times 2^4 = 48$ in total. The features for time domain include mean D, variance V, and correlation coefficients ρ. Therefore, the signal is represented as a vector whose length is 57 for time domain and frequency domain.

3. RELATIVE MARGIN SUPPORT VECTOR MACHINE FOR ROBOTIC ARM FAULT RECOGNITION

The extracted features are used to learn a relative margin support vector machine model. Here, this paper adopts relative margin support vector machine other than original support vector machine. The reason is that relative margin support vector machine (RMSVM) performs better than original support vector machine (SVM). In order to ensure the integrity of this paper, we give a brief review about RMSVM.

Let X represent the set of feature vectors of robotic arm signals, Y represent the associated label set. Here, $x_i \in \mathbb{R}^{57}$, $y_i \in \{+1, -1\}$. The aim of RMSVM is to find a hyperplane $f(x)$ to predict the label of future sample x. If $f(x) > 0$, the associated label of x is 1; otherwise, the associated label of x is -1. The optimal hyperplane of RMSVM must ensure that the minimum relative margin

between two classes is maximized. Let r_i represent the distance between x_i and $f(x)$. The r_+ and r_- are defined as following equations:

$$r_+ = \min_{x_i \in X, y_i = +1} r_i \tag{10}$$

$$r_- = \min_{x_i \in X, y_i = -1} r_i \tag{11}$$

The relative margin is defined as following equation:

$$\gamma = \frac{r_+ + r_-}{\|w\|} \tag{12}$$

Then, the training set is reorganized as $\{x_i, y_i, r_i\}_{i=1}^l$. An illustration of hyperplane and minimum relative margin is shown in Figure 1.

In Figure 1, the triangles and circles are from positive class and negative class, respectively. Finding the optimal hyperplane for RMSVM can be converted as a quadratic programming which is written as following optimal programming:

$$\min_{w,\rho} \quad \frac{1}{2} w^T w + C \sum_{i=1}^l \xi_i$$
$$s.t \quad w^T \varphi(x_i) \geq r_i - \xi_i, \quad \xi_i \geq 0 \quad i = 1,2,\ldots,l \tag{13}$$

In equation (13), $\phi(x_i)$ is the mapping of sample x_i in the reproducing Kernel Hilbert space (RKHS). By introducing Lagrange multipliers for constraints $w^T \varphi(x_i) \geq r_i - \xi_i w^T \varphi(x_i) \geq r_i - \xi_i$ and $\xi_i \geq 0 \xi_i \geq 0$, the optimal programming (13) can be converted as a quadratic programming with respect to Lagrange multipliers $\alpha_i, i = 1,2,\ldots,l$, which is written as following optimal programming:

$$\min_{\alpha} \quad \alpha^T Q \alpha$$
$$s.t. \quad \sum_{i=1}^l y_i \alpha_i = 0, \quad 0 \leq \alpha_i \leq r_i C, \quad i = 1,\ldots,l \tag{14}$$

Figure 1. The illustration of relative margin support vector machine. The triangles are from positive class, while the circles are from negative class. The solid line is the separated hyperplane $f(x)$.
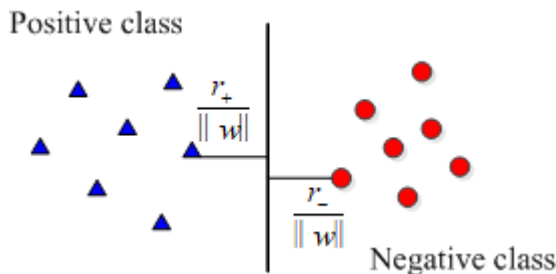
In equation (14), is the vector form of Lagrange multipliers. The sample with non-zero Lagrange multiplier is called as support vector. The separated hyperplane in RMSVM is written as following equation:

$$f(x) = sign\left(\sum_{\alpha_i \neq 0} \alpha_i K(x, x_i) + b\right) \qquad (15)$$

In equation (15), $K(x, x_i) = <\phi(x), \phi(x_i)>$ is the inner product of a pair of mappings in RKHS. It is called kernel function. The common used kernel functions include Gaussian kernel, Polynomial kernel etc. The bias b is obtained from the following equation:

$$b = y_i + r_i - \sum_{j=1}^{l} \alpha_i y_i K\left(x_i, x_j\right), \qquad 0 < \alpha_i < r_i C \qquad (16)$$

In optimal programming (13) and (14), the r_i reflects the relative position of sample x_i to hypeplane $f(x)$. It can be determined by one of the following ways:

$$r_i = \exp\left\{\omega \frac{D^k}{d\left(\boldsymbol{x}_i, \boldsymbol{x}_i^k\right)}\right\}, \qquad i = 1, 2, \ldots l \qquad (17)$$

$$\rho_i = 1 - \frac{1}{k}\left|c_i^{sum}\right|, \qquad i = 1, 2, \ldots l \qquad (18)$$

In equation (17), ω is a constant and satisfies $0 < \omega < 1 0 < \omega < 1$. In equation (18), c_i^{sum} is defined as following equation (Zhu 2016):

$$c_i^{sum} = \sum_{j=1}^{k} \cos\theta_j = \sum_{j=1}^{k} \frac{x_i - \overline{x_k}, x_i - x_j}{x_i - \overline{x_k} x_i - x_j} \qquad \left(x_j \in kNN\left(x_i\right)\right) \qquad (19)$$

The RMSVM only considers two classes. For multi-class classification problem, the problem need to be converted as several binary class classification problems via one versus one or one versus rest strategies. In one versus one, a multi-class classification problem containing p classes is converted as $p * (p - 1) / 2$ binary class classification problems, while in one versus rest, a multi-class classification problem containing p classes is converted as p binary class classification problems. Compared with one versus one strategy, the one versus one strategy needs to learn much more two-class classification models, but it can avoid the issue that the samples in two class are imbalance, which occurs in one versus rest strategy.

4. ROBOTIC ARM SIMULATION AND FAULT DETECTION

This paper constructs a complete virtual prototype simulation platform for robotic arm, which is shown in Figure 2. The input of virtual prototype is the torque of three joints, while the output is the angle

Figure 2. The architecture of virtual prototype simulation platform
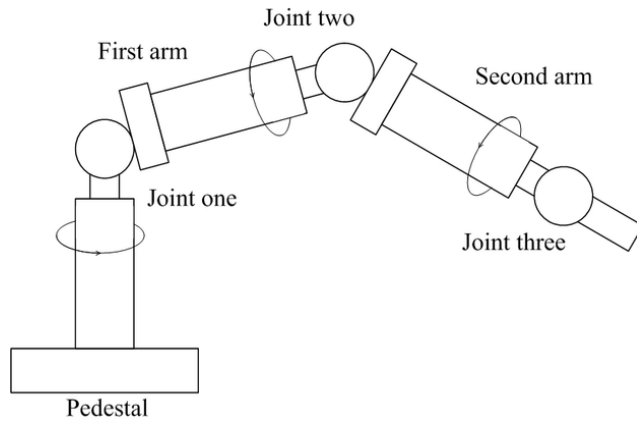
and angular velocity of three joints to facilitate the simulation of robotic arm faults for calculating the residual momentum.

In Figure 2, the virtual prototype consists of a pedestal, three robotic arms and three joints. The associated parameters of the virtual prototype are reported in Table 1.

There are many types of robotic arm faults for virtual prototype of robotic arms. This paper considers nine faults during simulation. The description of faults is reported in Table 2.

Table 1. The parameters of the virtual prototype simulation platform

	Length (mm)	Average diameter (mm)	Weight (kg)
Pedestal	450	115	119
First arm	260	68	37
Second arm	350	43	29
Third arm	150	29	19

Table 2. The states and faults of robotic arm in virtual prototype simulation platform

	Description	Status
S0	Normal state	Normal
S1	The motor in joint 1 motor occurs abnormal vibration fault.	Motor fault
S2	The motor in joint 1 motor occurs abnormal noise fault.	Motor fault
S3	The motor in joint 2 motor occurs abnormal vibration fault.	Motor fault
S4	The motor in joint 2 motor occurs abnormal noise fault.	Motor fault
S5	The motor in joint 3 motor occurs abnormal vibration fault.	Motor fault
S6	The motor in joint 3 motor occurs abnormal noise fault.	Motor fault
S7	The joint 1 occurs collision fault.	Collision fault
S8	The joint 2 occurs collision fault.	Collision fault
S9	The joint 3 occurs collision fault.	Collision fault

The residual momentum samples are obtained by adding simulating external signals as fault data samples. In the simulation, the sampling time is set as 10 seconds. By using co-simulation, a total of 3500 sets of residual momentum signals are collected through the sliding window on the time axis. In collected signals, 800 sets of robotic arm are under normal status, while 300 sets of robotic arm are under each fault. The collected signals are normalized and processed by equation (5), (6), (7) and (9) to extract the features in time domain and the features in frequency domain. Finally, the collected signals in each set are converted as vectors whose length is 57.

The features in time domain fluctuate drastically and there is no obvious rules. Thus, it is hard to detect the robotic arm faults merely using time domain features. The wavelet packet spectrum under different faults have significantly difference. However, the regularity is not obvious. Therefore, the time domain features and frequency domain features are combined to recognize the robotic arm faults.

The extracted features are used to train RMSVM model. The collected samples are split as training set and test set by using 10-fold cross-validation. The radial basis function (RBF) is used as the kernel function in RMSVM. In RMSVM, the relative margin r_i is determined by equation (18). The penal factor in RMSVM and the width in RBF kernel are tuned to ensure best cross-classification accuracy. The one versus rest strategy is adopted as multi-class classification. The robotic arm fault classification accuracy is reported in Table 3.

In Table 3, we compare the features from time domain, frequency domain, and time domain & frequency domain. For the classifier, we compare SVM and RMSVM. From the result in Table 3, when using support vector machine as classifier, the robotic arm fault recognition accuracy achieves 89%, 91%, 95% for time domain features, frequency domain features, time domain & frequency domain features, respectively; when using relative margin support vector machine, the robotic arm fault recognition accuracy achieves 90%, 93%, 98% for time domain features, frequency domain features, time domain & frequency domain features, respectively. It can be found that the time domain & frequency domain features perform better than merely using time domain features or frequency domain features; the RMSVM can achieves 98% when using time domain & frequency domain features, which is superior to classical SVM.

5. INDUSTRIAL ROBOTIC ARM FAULT RECOGNITION

In this experiment, we adopt industrial robotic arm to verify the proposed robotic arm fault recognition method. The industrial robotic arm consists three joints. In this experiment, we consider the movement of joint 2 and joint 3. The other joints are fixed. During the process of experiment, arm 1 is fixed and arm 2 moves counterclockwise 40 degrees and artificially creates collision faults by placing a workbench in the motion space of the robotic arm. During the operation of robotic arm, the angle and angular velocity data are collected to calculate the residual momentum.

In general, the cycle of the residual momentum changes obviously during normal motion, while the residual momentum changes significantly when a collision occurs. Due to the strict packaging of industrial robotic arms, it is difficult to collect the information of abnormal motor vibration and noise fault. In this experiment, we only evaluate and repeat the above normal and collision fault. We create collision faults at different time. The time domain features and frequency domain features of

Table 3. The robotic arm fault recognition accuracy in virtual prototype simulation platform

	SVM (o-vs-r)	RMSVM (o-vs-r)
Time domain features	89%	90%
Frequency domain features	91%	93%
Time domain & frequency domain features	95%	98%

Table 4. The robotic arm fault recognition accuracy in industrial robotic arm

	SVM (o-vs-r)	RMSVM (o-vs-r)
Time domain features	88%	89%
Frequency domain features	90%	92%
Time domain & frequency domain features	93%	96%

the residual momentum are input into classifiers. We also compare time domain features, frequency domain features, and time & frequency domain features to represent residual momentum of robotic arm signals. For classifier, we compare support vector machine (SVM) and relative margin support vector machine (RMSVM) as well. The associated experimental results are reported in Table 4.

In Table 4, when using support vector machine as classifier, the robotic arm fault classification accuracy achieves 88%, 90%, 93% for time domain features, frequency domain features, time & frequency domain features, respectively; when using relative margin support vector machine as classifier, the robotic arm fault classification accuracy achieves 89%, 92%, 96% for time domain features, frequency domain features, time & frequency domain features, respectively. It can be concluded that using time & frequency domain features performs better than merely using time domain features or frequency domain features; using relative margin support vector machine as classifier is better than using support vector machine as classifier for mechanical arm fault recognition.

6. CONCLUSION

In order to ensure safety during the operation of the mechanical arms, it is necessary to monitor and recognize the fault status of the mechanical arms in time. In order to monitor the status of the mechanical arms, we first set the distributed sensors on mechanical arm to collect the status signal of the mechanical arm, then extract the features in frequency domain from residue momentum, lastly learn a relative margin support vector machine by using the training set consisting of the samples from the extracted features. The proposed monitoring and recognizing faults method for mechanical arms is evaluated and verified on a virtual prototype platform and a real mechanical arm dataset. In the future work, the feature extraction should be further improved to enhance the fault recognition for mechanical arms.

ACKNOWLEDGMENT

REFERENCES

Blitz & Huij. (2011). Residual momentum. *Journal of Empirical Finance*.

Brkovic, A., Gajic, D., Gligorijevic, J., Savić-Gajic, I. M., Georgieva, O., & Gennaro, S. (2017). Early fault detection and diagnosis in bearings for more efficient operation of rotating machinery. *Energy*, *136*, 63–71.

Chan, H. P., Kim, H., Lee, J., Ahn, G., Youn, M., & Youn, B. D. (2020). A feature inherited hierarchical convolutional neural network (fi-hcnn) for motor fault severity estimation using stator current signals. *International Journal of Precision Engineering and Manufacturing-Green Technology*, 1-14.

Ghaffari, A., & Hashemabadi, S. H. (2016). Parameter study and cfd analysis of head on collision and dynamic behavior of two colliding ferrofluid droplets. *Smart Materials and Structures*, *26*(3), 035010.

Hua, B., Jisheng, M. A., Zhuo, H., & Dalin, W. U. (2013). *Impact parameter test of virtual prototype based on experiment and sqp optimization algorithm*. New Technology & New Process.

Ismail, M., Windelberg, J., & Liu, G. (2021). Simplified sensorless torque estimation method for harmonic drive based electro-mechanical actuator. *IEEE Robotics and Automation Letters*, *6*(2).

M., T., Grabbe, D., M., & Dawson. (1994). An application of optimal control theory to the trajectory tracking of rigid robot manipulators. *Optimal Control Applications and Methods, 15*(4), 237-249.

Newland, D. E. (2013). An introduction to random vibrations, spectral and wavelet analysis. *Spectral & Wavelet Analysis*, *108*(2), 140–147.

Santos, R., Steffen, V., & Saramago, S. (2010). Optimal task placement of a serial robot manipulator for manipulability and mechanical power optimization. *Intelligent Information Management*, *2*(9), 512–525.

Scholkopf, B. (1997). *Support vector learning* [PhD thesis]. Technischen Universitat Berlin.

Shao, D. L., Wang, B. R., & Jin, Y. L. (2015). Based on the residual momentum of driver fault detection for two-link flexible manipulator. *Acta Meteorologica Sinica*, *36*(3), 279–283.

Shao, S., Yan, R., Lu, Y., Wang, P., & Gao, R. X. (2020). Dcnn-based multi-signal induction motor fault diagnosis. *IEEE Transactions on Instrumentation and Measurement*, *69*(6), 2658–2669. doi:10.1109/TIM.2019.2925247

Shi, J., Liang, M., & Guan, Y. (2016). Bearing fault diagnosis under variable rotational speed via the joint application of windowed fractal dimension transform and generalized demodulation: A method free from prefiltering and resampling. *Mechanical Systems and Signal Processing*, *68*, 15–33.

Song, Y., Zhu, W., Xiao, Y., & Zhong, P. (2017). Robust relative margin support vector machines. *Journal of Algorithms & Computational Technology*, *11*, 186–191.

Torrence, C., & Compo, G. P. (1998). A practical guide to wavelet analysis. *Bulletin of the American Meteorological Society*.

Wang, S., Shao, D., Wang, L., Zhang, Y., & Wang, B. (2018). Faults classification for manipulators based on time frequency characteristics of residual momentum and support vector machines. *Journal of China University of Metrology*.

Wickerhauser, M. V. (2010). *Adapted wavelet analysis from theory to software*. A.K. Peters.

Xu, F., Peng, R., Zheng, T., & Xu, X. (2019). Development and validation of numerical magnetic force and torque model for magnetically levitated actuator. *IEEE Transactions on Magnetics*, *55*(1), 4900109.

Zhu, F., Yang, J., Gao, J., & Xu, C. (2016). Extended nearest neighbor chain induced instance-weights for SVMs. *Pattern Recognition*, *60*, 863–874.

Zhu, F., Yang, J., Xu, S., Gao, C., Ye, N., & Yin, T. (2017). Incorporating neighbors' distribution knowledge into support vector machines. *Soft Computing*, *21*, 6407–6420.