

ARL-TN-1044 • Nov 2020



Survey of Fault Detection and Classification in Power Conversion Electronics

by Charles W Tipton IV

Approved for public release; distribution is unlimited.

NOTICES

Disclaimers

The findings in this report are not to be construed as an official Department of the Army position unless so designated by other authorized documents.

Citation of manufacturer's or trade names does not constitute an official endorsement or approval of the use thereof.

Destroy this report when it is no longer needed. Do not return it to the originator.



Survey of Fault Detection and Classification in Power Conversion Electronics

Charles W Tipton IV

Sensors and Electron Devices Directorate, DEVCOM Army Research Laboratory

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing the burden, to Department of Defense, Washington Headquarters Services, Directorate for Information Operations and Reports (0704-0188), 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to any penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number.

PLEASE DO NOT RETURN YOUR FORM TO THE ABOVE ADDRESS.

1. REPORT DATE (DD-MM-YYYY) November 2020		2. REPORT TYPE Technical Note		3. DATES COVERED (From - To) 28 May–14 October 2020	
4. TITLE AND SUBTITLE Survey of Fault Detection and Classification in Power Conversion Electronics				5a. CONTRACT NUMBER	
				5b. GRANT NUMBER	
				5c. PROGRAM ELEMENT NUMBER	
6. AUTHOR(S) Charles W Tipton IV				5d. PROJECT NUMBER	
				5e. TASK NUMBER	
				5f. WORK UNIT NUMBER	
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) DEVCOM Army Research Laboratory ATTN: FCDD-RLS-RP Adelphi, MD 20783-1138				8. PERFORMING ORGANIZATION REPORT NUMBER ARL-TN-1044	
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)				10. SPONSOR/MONITOR'S ACRONYM(S)	
				11. SPONSOR/MONITOR'S REPORT NUMBER(S)	
12. DISTRIBUTION/AVAILABILITY STATEMENT Approved for public release; distribution is unlimited.					
13. SUPPLEMENTARY NOTES					
14. ABSTRACT This work summarizes the research trends in the area of anomaly detection and classification within electrical power conversion systems with special emphasis on motor drive inverters. To begin, an example of Army interest in and brief overview of the concepts and terminology of this field are given. These are followed with a general description of fault detection and classification methods and a summary of selected published works to provide the reader with a sense of the state of these methods as applied to power electronics. The analytical framework for anomaly detection and classification has been under development since the early years of the 20th century and all of the algorithms discussed in this review may be readily found in software libraries. It is, therefore, a reasonable expectation that machine learning (ML)-enabled products should be widely available. However, a significant oversight or underappreciation is the ability to produce an optimized feature vector that is statistically significant to the available training data. In order to successfully apply ML to a given problem, an optimally-selected feature vector and statistically significant training data are required. Based on this review, a study of optimal feature selection for fault detection in power conversion systems is warranted.					
15. SUBJECT TERMS anomaly detection, fault classification, power conversion, machine learning, feature selection					
16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT UU	18. NUMBER OF PAGES 28	19a. NAME OF RESPONSIBLE PERSON Charles W Tipton IV
a. REPORT Unclassified	b. ABSTRACT Unclassified	c. THIS PAGE Unclassified			19b. TELEPHONE NUMBER (Include area code) (301) 394-5209

Contents

List of Figures	iv
List of Tables	iv
1. Introduction	1
2. Background	1
3. Anomaly Types	2
4. Fault Detection and Classification Techniques	4
4.1 Data Preprocessing	5
4.2 Feature Creation, Selection, and Extraction	7
4.3 Classification	9
5. Recent Work	15
6. Summary	18
7. References	19
List of Symbols, Abbreviations, and Acronyms	21
Distribution List	22

List of Figures

Fig. 1	Simplified schematic of DC-AC motor drive inverter.....	2
Fig. 2	Anomaly class examples: a) point, b) contextual, and c) collective.....	3
Fig. 3	Wavelet coefficient generation algorithm.....	7
Fig. 4	Comparison of crisp and fuzzy membership functions.....	11
Fig. 5	Examples of hyperplane in a) 2-D and b) 3-D space.....	11
Fig. 6	Classifying a new data point (green circle) in a two-class data set using k-NN.....	12
Fig. 7	Diagram of ANN.....	13
Fig. 8	Nonlinear transformation used to identify a linear hyperplane.....	14
Fig. 9	Diagram of a hybrid adaptive neuro-fuzzy inference algorithm.....	14

List of Tables

Table 1	Summary of selected power inverter fault detection and classification works.....	16
---------	--	----

1. Introduction

This technical note summarizes the research trends in the area of anomaly (fault) detection and classification within electrical power conversion systems with special emphasis on motor drive inverters. To begin, an example of Army interest and a brief overview of the basic concepts and terminology used in this field are given. Next, a general description of fault detection and classification methods is presented with details provided about the methods most frequently used in power conversion. The third section summarizes selected published works to provide the reader with a sense of the state of these methods as applied to power electronics.

2. Background

Based on a detected fault or anomalous condition, we would like a future military system to inform the commander or operator, as well as provide operational recommendations. For example, we may predict that a future, hybrid-electric fighting vehicle has multiple motor-drive inverters powering the main traction drive sprocket, fuel pump, environmental conditioning unit, and so on. A simplified schematic of one of these inverters is shown in Fig. 1. Here a DC power source (battery, V_{in}) feeds the power inverter, composed of semiconductor switches $S1-S6$, which drive an AC motor. Each switch pair ($S1-S4$, $S3-S6$, and $S2-S5$) are operated in coordination to produce the three phase currents (i_a , i_b , and i_c). If, for example, a fault is detected in $S1$, we would like the machine learning (ML)-based control system to inform the commander of how long the present operational tempo may be maintained or how this fault could impact mission success. These results would be based on accurately classifying the fault and assessing the fault's functional impact based on historical data from fleet usage or simulation studies. Additionally, and with more difficulty, we would like the ML system to provide a justification for the recommendation given.

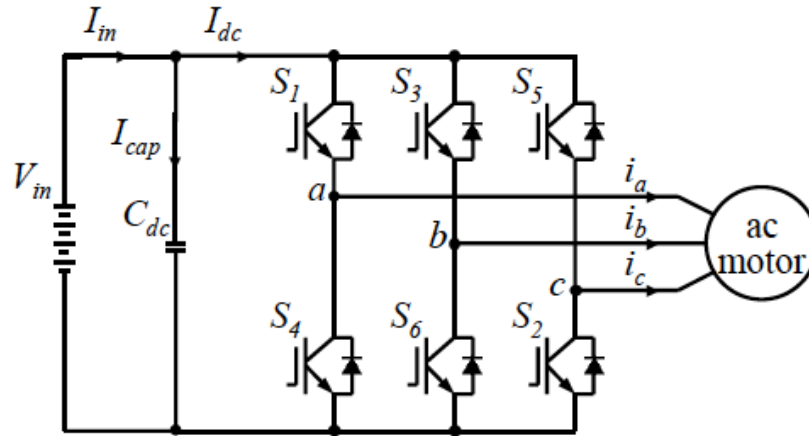


Fig. 1 Simplified schematic of DC-AC motor drive inverter

3. Anomaly Types

Anomalies (the terms anomaly and fault will be used interchangeably) may be classified into three categories: *point*, *contextual*, or *collective*. Point anomalies can occur in any data set and may be evaluated at each data instance. Collective anomalies can occur only in those data sets in which multiple data instances are related. Identifying contextual anomalies requires an evaluation of the data instance in the light of other determining attributes. Note that a point or collective anomaly can also be a contextual anomaly if analyzed with respect to a context. As shown in Fig. 2a, point anomaly (point *P*) lies well outside the two other data clusters. In this data, no relationships or context are assumed among the data points. Fig. 2b is a representation of the average high temperature over the course of several years. We note the data instance, which is contextually anomalous with respect to the nature of the data set. The observed temperature is very low for that expected in a typical summer. Therefore, that temperature instance is not anomalous because of its value per se, but because of the context of that value. Finally, in Fig. 2c, a collective anomaly is observed because a collection of related data instances is anomalous with respect to the entire data set. Here, there is a period of time over which the signal is three times its nominal frequency. Again, note that the values of specific data instances in a collective anomaly may not be anomalies when considered individually, but their occurrence as a collection may be anomalous.

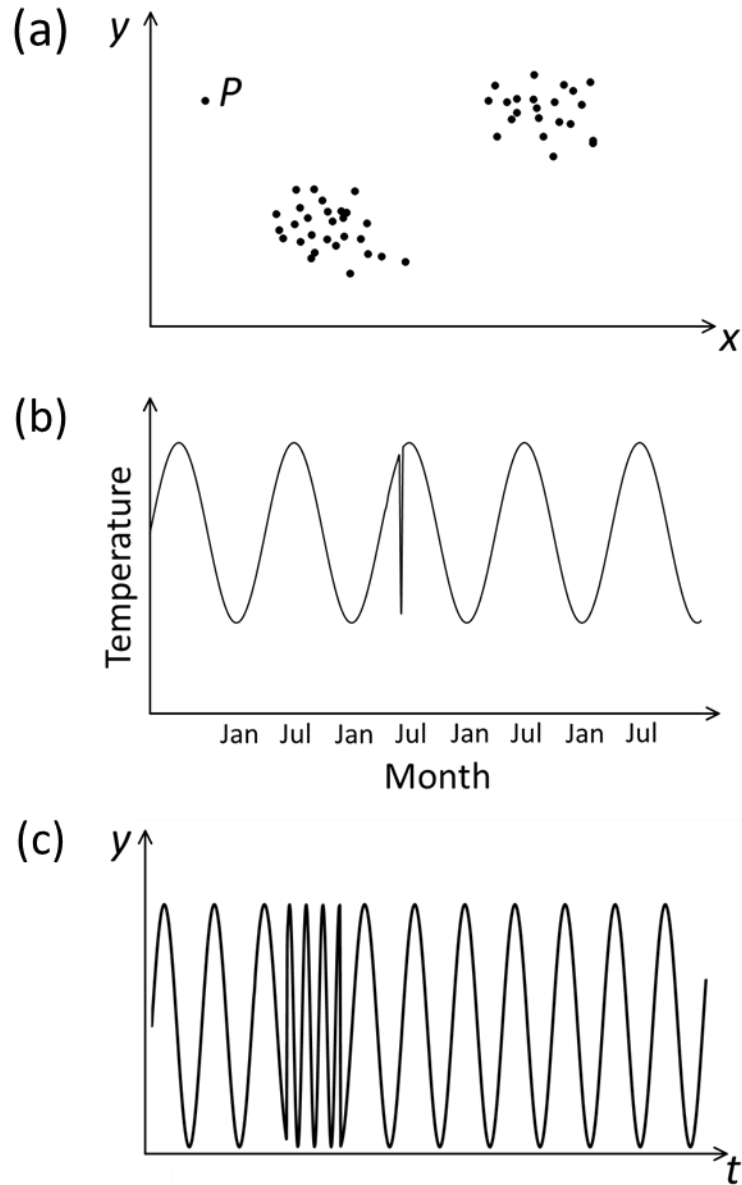


Fig. 2 Anomaly class examples: a) point, b) contextual, and c) collective

With knowledge of the system and operating conditions, one is able to label each data instance as normal or abnormal. Obtaining a labeled set of data, covering all possible type of anomalous behavior, is more difficult than obtaining labels for normal behavior. In either case, the process by which data is labeled is often performed by a human expert and therefore requires substantial effort to obtain. Based on the availability of labeled (or training) data, detection algorithms operate in one of four modes: supervised, semi-supervised, unsupervised, or reinforced. Algorithms trained in the supervised mode require the availability of a training data set that has labeled instances for normal and anomalous classes by which the

classifier is trained. Algorithms that operate in a semi-supervised mode assume that the training data has labeled instances for only the normal class and from which a normal model is derived. Based on this model, a discrimination of the unknown data set may be performed. Algorithms that operate in unsupervised mode do not require training data and make the implicit assumption that normal instances are far more frequent than anomalies in the test data. Anomalies, therefore, are those data instances that do not conform to the population centers (clusters). Reinforcement algorithms do not use training data per se. Labels are assigned based on past success (exploitation) and also by “guesses” (exploration). Here, the labeling process uses trial and error guided by a success-oriented reward policy.

4. Fault Detection and Classification Techniques

There are many anomaly detection techniques discussed in the literature, which are focused, primarily, on detecting point anomalies. Included, by category, are the following:

- 1) Classification
 - Neural networks
 - Bayesian networks
 - Support vector machines
 - Rule-based
- 2) Nearest neighbor
 - Distance to k th nearest neighbor
 - Relative density
- 3) Clustering
- 4) Statistical
- 5) Spectral
- 6) Contextual

It is beyond the scope of this technical note to fully describe all of these techniques; however, several will be described to give background information to support the discussion of cited works.

In comparison to the rich literature on point anomaly detection techniques, research on contextual anomaly detection has been limited. Contextual anomalies require that the data has a set of contextual attributes (context), and a set of behavioral attributes through which anomalies may be identified. Broadly, these techniques fall into two categories: those that reduce the contextual anomaly detection problem

to a point anomaly detection problem, and those that model the structure within the data and use this model to detect anomalies.

Collective anomaly detection is more challenging than point and contextual anomaly detection because it involves exploring structures within the data for anomalous regions. A primary requirement for collective anomaly detection is the existence of a relationship between data instances. Three types of relations that are frequently exploited are sequential, spatial, and graph.

As we consider fault detection or performance degradation in power conversion systems, anomalies may occur within a specific context. For example, a commanded voltage or expected efficiency might not be attained. Or, it might occur as an anomalous sequence of observations (collective anomalies), such as abnormalities in switching waveforms. The trend has been to reduce both scenarios to point-anomaly detection problems.

The fault detection and classification process may be described, generally, by four steps:

- 1) Modeling/characterization/data collection
- 2) Preprocessing
- 3) Feature creation, selection, and extraction
- 4) Classification

In the first step, a model describing the behavior of the system is developed. This model may be analytical (i.e., based on physical laws), linguistic (i.e., based on linguistic description), or empirical (i.e., based on data collected from the system). The model should be capable of generating all relevant behavior, including all expected conditions such as steady-state operating, transient loading, noise, and fault conditions. As this step is focused, primarily, on system modeling, and not fault detection and classification, no further details will be provided here.

4.1 Data Preprocessing

In the data preprocessing step, the operational data set is enhanced and/or converted to the appropriate processing domain with the goal of making the anomalies of interest easily recognizable by the detection algorithm. Some examples include the following:

- Outlier removal, offset removal, data clipping, and value normalization.
- Noise reduction by filtering or smoothing.

- Transformations within or between time and frequency domains.
- Advanced, time-frequency signal processing such as Fourier and wavelet transformations.

Consider, for example, processing time-based phase current data from an induction motor drive system. Time-frequency based approaches, such as the Fourier transform, are suitable if the speed of the drive remains constant for significant periods of time. That is, the statistical properties of the phase current do not change over time (stationary process¹). In many situations where variable-speed drives are employed or where speed and/or loading vary considerably with time, nonstationary approaches, such as the short-time Fourier transform (STFT) and wavelet transform (WT), are used. The STFT uses the standard Fourier transform over a small subset of the data (window) and can provide some information about both time and frequency. The primary shortcoming of the STFT is its limited time-frequency resolution, which is subject to the uncertainty principle. Simply stated, low frequencies cannot be resolved using short-time windows and short pulses cannot be accurately localized in time with long-time windows. This limitation was a strong motivation for the development of wavelet theory.

Just as the Fourier transform decomposes the input signal by using sinusoidal waveforms of many frequencies, the WT decomposes the input signal using a basis waveform called the mother wavelet and is given by:

$$CWT_x^\psi(\tau, s) = \frac{1}{\sqrt{|s|}} \int x(t) \psi^* \left(\frac{t-\tau}{s} \right) dt, \quad (1)$$

where x is the time-dependent input function, ψ is the mother wavelet function, s is the scaling parameter, and τ is the translation parameter. In this equation, the transform is simply the inner-product of the input function and amplitude-scaled, time-translated version of the mother wavelet. Although Eq. 1 is the continuous wavelet transform (CWT), in this note any reference to the WTs implies a practical implementation using the discrete wavelet transform (DWT), which is the dominant form used in digital signal processing applications. The WT uses a variable-size windowing technique to analyze regions of the data set. Long-time windows are used where precise, low-frequency information is needed. In this case, one may still obtain information about high frequency components; however, the position (in time) cannot be accurately determined. By using short-time intervals, the time position of high-frequency information can be precisely determined. Localized analysis in the time-frequency or time-scale domains is one of the significant advantages of the WT and make it an effective approach for condition monitoring and fault detection of both stationary and nonstationary signals.²

A depiction of a 3-level DWT is shown in Fig. 3. The input signal (x), contains n sample points that are passed through a high pass filter (HPF) and a low pass filter (LPF). The filters' outputs are then down-sampled by a factor of 2. The output of the LPF path is referred to as the *approximation coefficients* (A_i) and the output of the HPF path is referred to as the *detail coefficients* (D_i). This decomposition process may then be repeated on the detail coefficients to obtain the next-level transform. As the frequency content is reduced by a factor of 2 per level, decomposition is usually performed until the frequency of interest is obtained. To effectively use the DWT, it is critical that proper mother and scaling functions and decomposition level be chosen such that the original signal can be accurately reproduced.

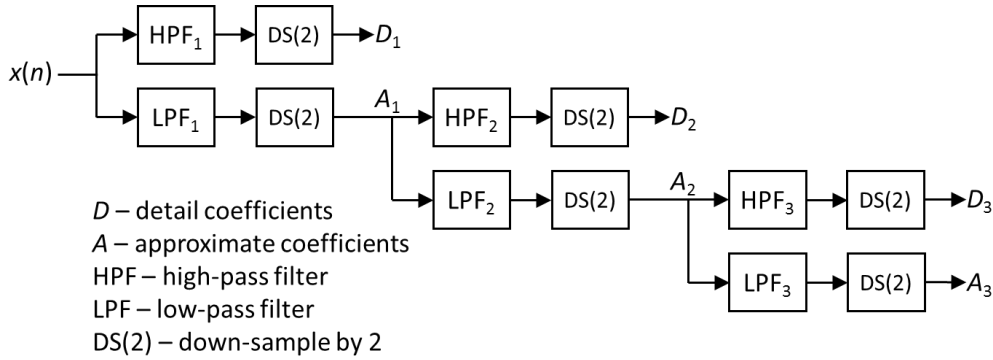


Fig. 3 Wavelet coefficient generation algorithm

4.2 Feature Creation, Selection, and Extraction

The goal of the third step is to select, from the processed data, a feature vector that has the highest probability of indicating, and perhaps classifying, an anomaly. Here, the feature vector is a set of parameters used to detect and classify anomalies. These parameters may include Fourier or wavelet coefficients, phase current values, or any other parameter relevant to anomaly detection. It is very important to note that classification performance using a large number of features is, however, not optimal from both the perspective of computational cost and classification accuracy.

Many methods for dimensionality reduction, based in the time or frequency domains, have been proposed during recent decades and a survey of 25 of these is presented in Phinyomark et al.³ As an example, “energy” is a common feature derived from the DWT coefficients and is defined as:

$$E_j = \sum_n |c_j(n)|^2, \quad (2)$$

where there are n , DWT coefficients (c) at decomposition level j .

More frequently, statistical calculations are performed on the preprocessed data to obtain features. Most notably used are the first four statistical moments:

mean (μ) representing the average value,

$$\mu = \sum_i x_i p(x_i), \quad (3)$$

where $p(x_i)$ is the probability of the occurrence of the i th value (x_i),

variance (σ^2) represents how closely values under consideration are spread about the mean,

$$\sigma^2 = E[(X - \mu)^2] = \sum_i (x_i - \mu)^2 p(x_i), \quad (4)$$

skewness (γ_1) that reflects the dissymmetry of the distribution of values,

$$\gamma_1 = E \left[\left(\frac{X - \mu}{\sigma} \right)^3 \right], \quad (5)$$

and *kurtosis* (κ) that reflects the degree of flatness of the distribution of values,

$$\kappa = E \left[\left(\frac{X - \mu}{\sigma} \right)^4 \right]. \quad (6)$$

Although statistical features are often used in the detection algorithms, it has generally been observed they are not sufficient to detect with high confidence levels. Moreover, fault classification is nearly impossible without the consideration of additional features.

As previously indicated, it is possible to generate an overwhelming number of features from the preprocessed data. In order to interpret such data sets, methods are required to drastically reduce the dimensionality, while preserving relevant information contained in the data. Most of the references cited in this note did not explicitly discuss the formal process by which features were selected. That is to say, features were selected until an acceptable result was achieved. However, it is unknown if the selected set is the optimal set. Feature selection is a field of study unto itself and introductory information on this topic may be found in the work of Kuhn and Johnson.⁴

Principal component analysis (PCA) is one of the oldest and most widely used selection methods. Its goal is simple—to reduce the dimensionality of the feature vector, while preserving as much “variability” (i.e., statistical information) as possible. This translates into finding new features that are linear combinations of those in the original vector, that maximize variance, and that are uncorrelated with each other. The PCA linear transformation maps features to a new coordinate

system such that the greatest variance lies on the first coordinate (called the first principal component), the second greatest variance on the second coordinate, and so on. PCA can be thought of as fitting an m -dimensional ellipsoid to the feature vector, where each axis of the ellipsoid represents a principal component. If an axis of the ellipsoid is small, then the variance along that axis is also small, and an equally small amount of information is lost by omitting this axis and corresponding principal component from the representation of the data set. An assumption in this discussion is that the most important features are those with the highest variance.

4.3 Classification

The fourth step is classification, in which the selected features are analyzed to detect and classify faults. A classification model must be developed through which unlabeled data objects are assigned to one of several predefined categories or classes. Nearly all of the common classification methods have been applied to power conversion applications. These include the following:

- 1) Decision tree (DT)
- 2) Fuzzy inference system (FIS)
- 3) Support vector machine (SVM)
- 4) k th nearest neighbor (k-NN)
- 5) Artificial neural network (ANN)
- 6) Adaptive neuro-fuzzy inference system (ANFIS)

The decision tree (DT) classifier is a supervised learning algorithm. Recall that, in supervised learning, the existing data is already labeled and, therefore, one knows which features and rules will allow the unknown data set to be classified with little or no error. The classification model is constructed by building a DT based on training data that includes all possible features and their values. Each node in the tree specifies a test on a feature, each branch descending from that node corresponds to one of the possible values of that feature. Each leaf represents class labels associated with the instance. Instances in the training set are classified by navigating them from the root of the tree to a leaf, according to the outcome of the tests along the path. Starting from the root node of the tree, each node splits the instance space into two or more subspaces according to the feature test condition. Then moving along the tree branch corresponding to the value of the feature, a new node is created. This process is then repeated for the subtree rooted at the new node, until all records of the training set have been classified. There are two types of DTs: classification and regression. Classification trees sort based on qualitative sorting,

whereas regression trees are based on quantitative sorting. In power conversion fault classification, regression trees are often used because features are based on specific parameter values.

Fuzzy inference systems (FISs), based on fuzzy logic, are similar to DTs in that both are supervised algorithms in which a fixed (static) set of rules are obtained from training data. Unlike the “crisp” data used in the regression trees, the FIS uses fuzzy rules and, therefore, fuzzification and defuzzification processes are needed in a practical classifier system. To illustrate the fuzzification–defuzzification process, consider a simple cooling system. In addition to the fuzzy controller, this system has a temperature sensor, motor controller, and fan. When the digital temperature sensor reports a “crisp” value of 97.4 °F to the fuzzification process, a membership function maps a fuzzy value to the crisp input value—in this case, 97.4 is converted to a value that indicates the degree to which this temperature should be considered “HOT”. Membership functions usually have an output range of [0,1] and, therefore, let us say that we map a value of 0.87 to the input value of 97.4. Selection of the membership function type (e.g., triangle, trapezoid, Gaussian) is based on the expert knowledge of the system designer. Figure 4 compares membership functions for crisp and fuzzy systems. In the present example, assume that Fig. 4 is the membership function for the set of “WARM” temperatures and that point “a” is 70 °F and point “b” is 80 °F. In a crisp controller, only temperatures between 70 °F and 80 °F could be labeled warm. Furthermore, an input temperature of 72 °F will have the same label as an input of 79 °F. The fuzzy controller considers the entire range of input values to be “WARM” and applies a strength value based on the membership function. We can assume that there is also an overlapping “COLD” membership function (not shown) and that point “a” will also be considered to be a member of this class, but to a lesser strength. Having assigned membership strengths to the input value, the fuzzy “classification” rules in the controller are applied to obtain a fuzzy speed value for the cooling fan. Finally, a defuzzification process is used to obtain a crisp value that can be used by the fan motor controller, say 103 rpm. When compared to a crisp controller that uses precise thresholds, the imprecise nature of the FIS allows more flexibility in defining the classification rules and optimizing their effectiveness.

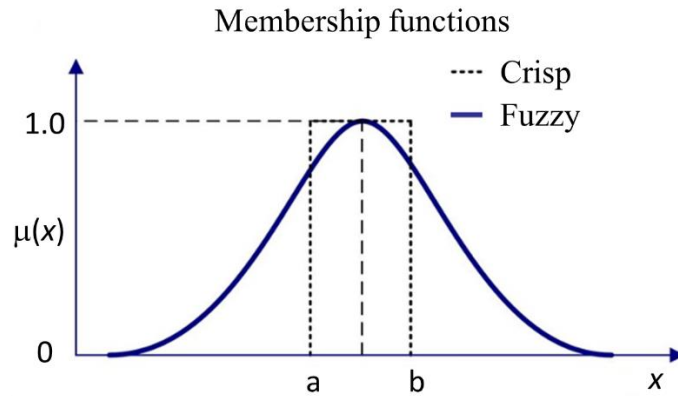


Fig. 4 Comparison of crisp and fuzzy membership functions

A very common algorithm used to distinguish two classes is the support vector machine (SVM). The objective of the SVM algorithm is to identify the hyperplane that delineates the maximum margin (distance) between the data points of the two classes. Hyperplanes are decision boundaries that are used to classify the data points while the support vectors are those data points that have the greatest influence on the position and orientation of the hyperplane. By definition, the dimension of the hyperplane is one less than the feature vector dimension. So, as shown in Fig. 5, if the number of input features is 2, then the hyperplane is just a line. Although any separating hyperplane could be chosen, the one maximizing margin within the training data provides some assurance that classification algorithm will be successful when applied to unknown data sets.

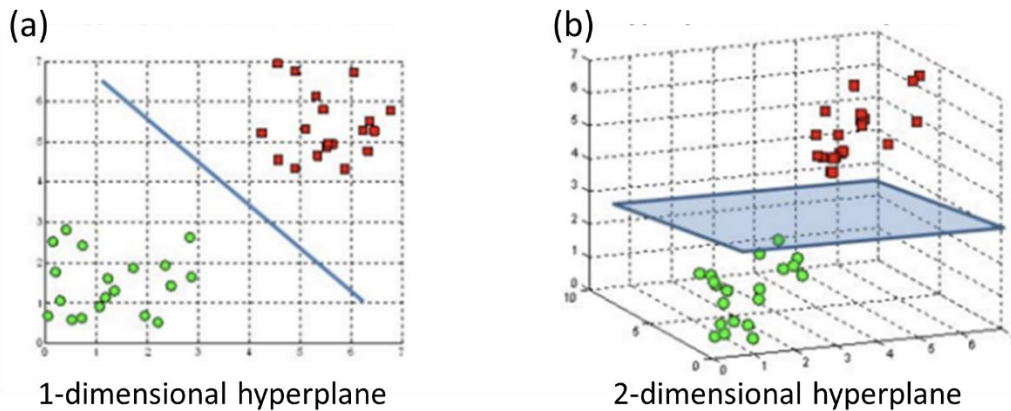


Fig. 5 Examples of hyperplane in a) 2-D and b) 3-D space

The k -nearest neighbor (k -NN) algorithm is based on majority voting and uses locally stored, labeled training data to classify unknown data points based on how their neighbors are labeled. Of course, in this scheme, we must define how

neighbors are selected. Figure 6 shows an example of a k-NN system with an unlabeled data point (green circle) being classified using a value of k being 3 and 5. When k is equal to 3, the new data point is classified as a red triangle, whereas it is classified as a blue square when k equals 5. The nearest neighbors are usually selected by a distance measure such as Euclidean, Manhattan, and Hamming. The value of k is often determined by using a validation process. Here, a subset of the test data is selected and classified using a range of k values. The value of k selected is that which minimizes the classification error. Alternatively, k may be set to \sqrt{n} , where n is the number of samples in the training data set. Additionally, k should be odd to assure that a majority vote is guaranteed in a two-class system. The two obvious drawbacks of the k-NN classifier are that the training set must be locally stored and that distance must be computed for each training point.

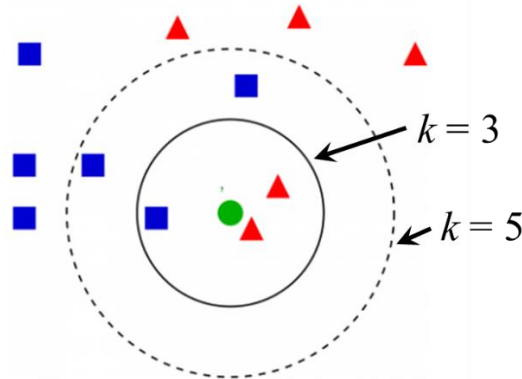


Fig. 6 Classifying a new data point (green circle) in a two-class data set using k-NN

Artificial neural networks (ANNs) are algorithms, modeled loosely after the human brain, which are designed to recognize patterns. After being trained with a labeled data set, an ANN can group unlabeled data based on similarities of the input features. Figure 7 shows a common depiction of an ANN. The network is formed of interconnected layers containing nodes. Each node performs a relatively primitive computation and, similar to a human neuron, fires when it encounters sufficient stimuli. The input layer (first layer) is normally considered a passive layer through which the input stimuli values are distributed to the hidden layer (second layer). Within the hidden layer, the starting input values are multiplied by coefficients (weights), either amplifying or dampening each value through the transformation function (f), which assigns significance to that input. These input-weight products are summed and then the sum is passed to the output layer. The output node performs an addition computation—the nonlinear, activation function (g). Here, the outputs of the hidden layer are multiplied by another set of weighting coefficients and summed; giving the final output value. In order for the ANN to be accurate, all of the weights must be optimized through a “training process” referred

to as back-propagation. This is an iterative process that compares the output of the network with the expected output, and using the difference, modifies the weight values to minimize error. The effectiveness of the ANN is determined, in part, by the sample size and quality of the training data set. As a side note, deep-learning networks are distinguished from the single-hidden-layer ANN by their depth, that is, the number of hidden layers. If a network has more than one hidden layer, it qualifies as a “deep” learning network. As the resultants of the input features propagate through the deep neural network, the nodes can “recognize” more complex features because they aggregate and recombine node firings of the previous layer. Training of deep-learning networks is not trivial and requires orders of magnitude more training data compared to a simple ANN.

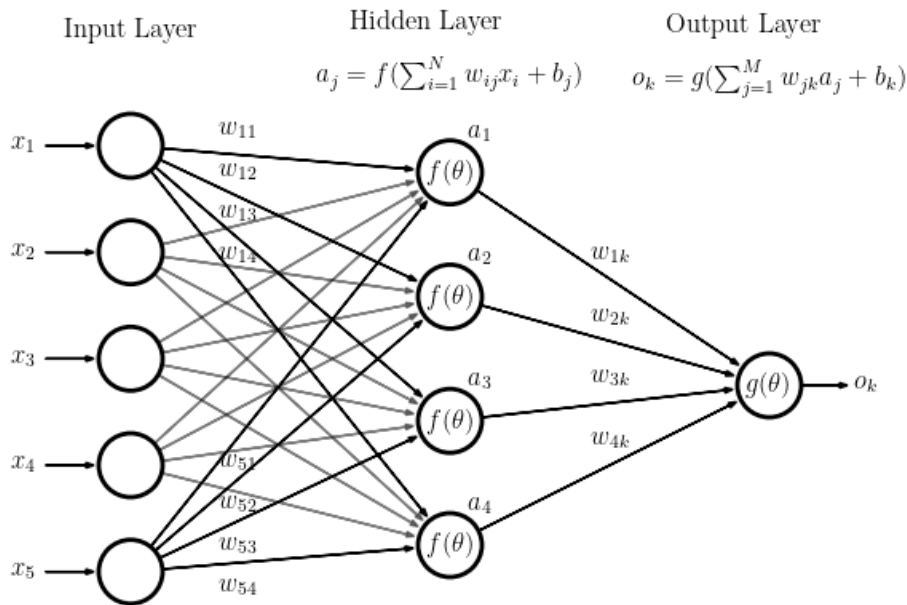


Fig. 7 **Diagram of ANN**

In Fig. 5, we saw how linear hyperplanes are used to classify data in its native space. Some data sets, however, are best separated by a nonlinear decision boundary. Because the ANN can perform nonlinear transformations, it has the added benefit of being able to project input features to a new vector space, as depicted in Fig. 8. By transforming the nonlinear (green curve) decision boundary of the input space into a linear boundary, previously discussed classification methods are more easily implemented.

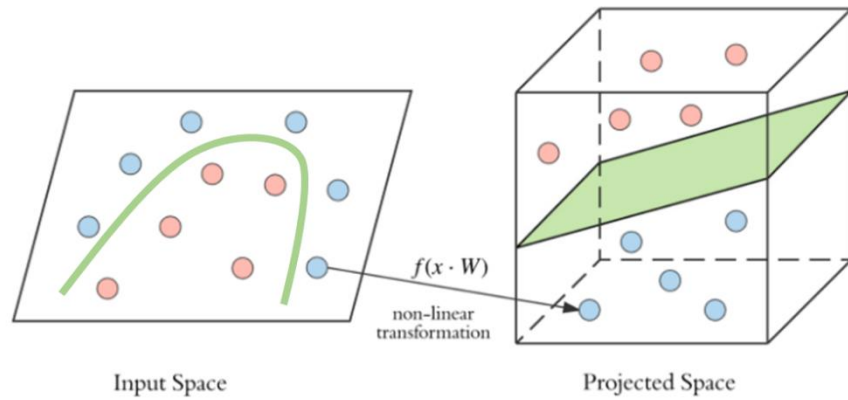


Fig. 8 Nonlinear transformation used to identify a linear hyperplane

The final method to be discussed in this review is the hybrid adaptive neuro-fuzzy inference system (ANFIS), which was introduced in the early 1990s.⁵ This approach combines the ability of the FIS to model complex problems while taking into consideration inherent uncertainties of human knowledge and exploiting the ability of ANNs to “learn” relationships and make generalized decisions from historical data. A notional diagram of the ANFIS structure is given in Fig. 9.

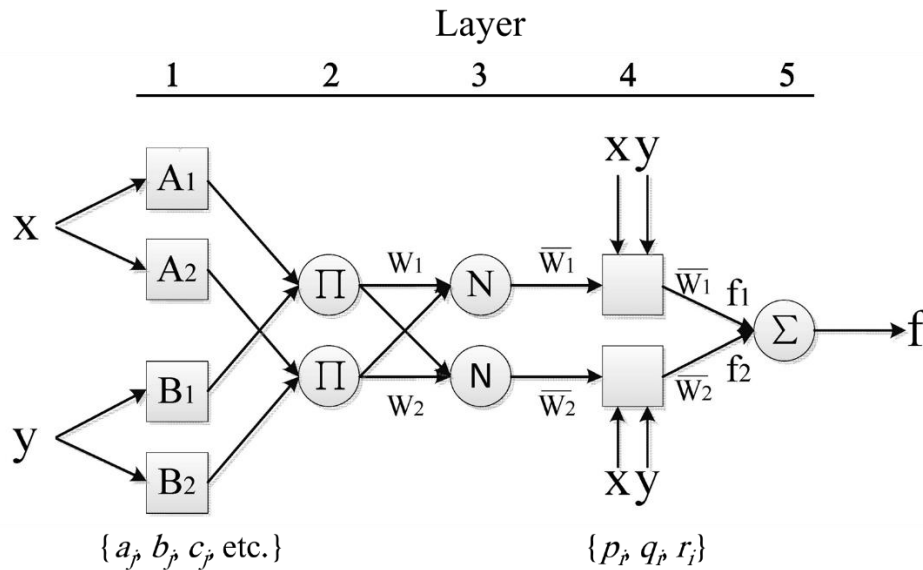


Fig. 9 Diagram of a hybrid adaptive neuro-fuzzy inference algorithm

Layer 1 is the fuzzification stage in which the feature data (x and y) from the preprocessing stage are converted to fuzzy values using membership functions with the premise parameters (a_j, b_j, c_j). In Layer 2, or the rule layer, pairs of outputs from Layer 1 are multiplied to generate terms that correspond to the “firing”

strengths of each rule. Layer 3, then, generates a set of normalized firing strengths (\bar{W}_i) wherein, the i th output is the ratio of the i th input to the sum of all the firing rules' strengths. Each node in Layer 4 (equivalent to the ANN hidden-layer) has the function $\bar{W}_i(p_i x + q_i y + r_i)$, where $\{p_i, q_i, r_i\}$ are referred to as the consequence parameters. The consequence and premise parameters are optimized/trained with back-propagation techniques. Layer 5 includes the activation (defuzzification) functions and provides the final, crisp output value.

5. Recent Work

Fault diagnosis in motor drive inverters has been an active area of research since the mid-1990s. In the early years, fault detection was based on the analyses of time-series data with collective classification: relatively little signal processing was performed in this process. Drive faults have been divided into three classes:

- 1) *Abrupt faults* that occur suddenly and produce significant changes in system behavior.
- 2) *Degradation faults* that produce changes in system behavior over a given interval of time.
- 3) *Intermittent faults* that produce short-term perturbations but, no perceivable degradation over time.

Due to its time-invariant nature, abrupt fault detection in electrical machines and drives has been widely studied through experimentation and simulation. Detection methods for degradation faults is becoming a hot topic in academic research. Here, it is crucial to detect the fault at the earliest possible stage (lowest detectable level) when the fault is still considered to be incipient. The intermittent fault is the most difficult to detect because it occurs randomly and with differing durations, furthermore, its severity can vary from incipient to severe.⁶

Along with the dynamic differences, there are numerous fault sources within an inverter drive system; included are the following:

- DC bus to earth short
- DC link capacitor short
- Transistor base/gate open
- Transistor short/open
- Diode short/open
- Line-to-line short

- Line-to-ground short

In addition, electric machine faults such as insulation failure and broken rotor bars also impact fault diagnosis. It is estimated that among all faults types in variable-speed AC drives, about 38% are due to failures of the power semiconductor devices. Most of these inverters are based on insulated-gate bipolar transistors (IGBTs) because of their high-voltage and -current ratings and ability to handle short-circuit currents. Nevertheless, they often suffer failures due to excess electrical and thermal stress that are experienced in this application. IGBT failures can be broadly classified as open-circuit, short-circuit, and misfiring faults.⁷

Although thermal and acoustic sensors have been used as diagnostic tools for electric motor failures, the most common method to assess the health of a motor (and its drive) is to monitor the input phase currents. Moreover, most motor current signature analysis (MCSA) is performed in the frequency domain due to its simplicity and accuracy. Table 1 presents a summary of selected articles on inverter fault detection and classification. Though not an exhaustive list, it does indicate the trends (or lack thereof) and methods used. In the following several paragraphs, details on other relevant works are given.

Table 1 Summary of selected power inverter fault detection and classification works

Year	Reference	Input	Preprocessing	Feature generation	Primary feature	Classification
2001	11	Phase currents	None	DWT (Daubenchies 1)	D_1	FIS
2004	12	Phase currents	Clarke-Park transform	Mean current	Mean current	ANFIS
2008	13	Phase currents	Normalization	DWT (Daubenchies 2)	$D_{1,2}$	SVM
2010	14	Phase currents	None	DWT (Symlet 2)	Mean energy	DT
2013	15	Phase currents, torque, speed	None	DWT (Haar)	D_{1-3}	ANN
2015	16	Phase currents	None	DWT (Daubenchies 2)	D_{1-4}	FIS
2016	7	Phase currents	None	THD	THD	DT
2016	17	Phase currents	Clarke transform	DWT (Daubenchies 6)	Mean, Total energy	ANN

In 2001, one of the first articles proposing the use of a wavelet transform (and fuzzy logic) in inverter fault detection was published.⁸ The authors consider a three-phase induction motor drive in which only a single transistor open-circuit fault could

occur. Failure features were extracted from the phase currents using DWT with the Daubenchies-1 mother wavelet. At each time step, data in a sliding window was used to compute DWT coefficients. A “significant” change in the stator current waveforms was flagged if any wavelet coefficient exceeded or fell below an empirically determined range of values. On detecting a change, the system determined the DC-offset in the sliding window of stator currents. Then, the system waited for four cycles of the drive current before checking the DC-offset for a second time. A transistor open-circuit fault was considered to have occurred if the DC-offset persisted. If the DC-offset disappeared in the second reading, however, an intermittent misfiring was considered to have occurred. Experimental verification of this technique was performed at a constant speed and supply voltage. Although not implemented, the authors suggest that an FIS may be used as a fault classifier based on the polarity of each of the offset levels of the three phases. Again, only long-term, severe faults were addressed in this note.

In 2016, Kumar,⁷ performed a circuit simulation study on a 3-phase induction motor drive. Here, the total harmonic distortion of the phase currents was calculated as the simulated fault was introduced. These faults were severe and long-term in nature as the steady-state fault condition was reached. Several fault scenarios were simulated: open- and shorted-diode, open- and shorted-switch, open- and shorted-gate. Although there was a clear distinction between the total harmonic distortion (THD) in normal and fault states, the specific fault type could not be determined.

In 2017, Mousavi⁹ investigated electro-mechanical fault detection of an induction motor. Here, phase current data were obtained from finite element analysis and experimentation for three fault modes: broken rotor bars, bearing failure, and rotor eccentricity. DWT analysis, using the Daubenchies-4 mother-wavelet with five levels of decomposition, was performed on the data and a feature vector was extracted from the wavelet coefficients, which included minimum, maximum, standard deviation, and energy of the signals. The final step in fault detection was to calculate the Euclidian distance between the known good and unknown feature vectors. This value was compared with an empirically determined threshold to detect a fault. Unfortunately, the authors did not include the same operational-fault conditions of both the simulated and experimental data presentation. There is a clear relationship between features and the number of broken rotor bars in the experimental results. However, the reader is unable to ascertain if those results are predictable by simulation. In this work, no attempt was made to develop a classification algorithm to identify specific fault types.

In 2019, Baghli¹⁰ investigated the detection of short-lived (100- μ s to 1-ms) fault conditions using statistical detection techniques. It was found that although the skewness and kurtosis were able to discriminate current faults, they were not

sufficient, especially in the case of the incipient faults. Therefore, the authors incorporated the Kullback–Leibler divergence (KLD) technique, which utilizes the fault- and normal-state probability density functions, to derive a fifth detection criterion—the KLD “distance”. Feature analysis used PCA to study the amount of variation of the five criteria under different operational and fault conditions. The application of PCA allowed the classification of incipient, open-switch faults using only the first and second principal components at higher shaft speeds. Faults were properly classified according to their duration and operating condition (speed, load, and noise level).

In the research cited, motor phase current data was used as the basis for fault detection and classification. In the majority of cases, the DWT components were used in the formation of a feature vector. None of these works suggested that other signals, within the power electronics, were considered as part of the feature selection process.

6. Summary

The analytical framework for anomaly detection and classification have been under development since the early years of the 20th century, and all of the algorithms discussed in this review have been implemented in most computational software products and libraries. Although the pace of research in the area of fault detection and classification for electric motor drives is increasing, the author has been unable to find examples of fielded products. Perhaps a significant oversight, or underappreciation, is our ability to produce an optimized feature vector that is statistically significant to the available training data. At one extreme, *high variance*, a model has too many features in relation to the amount of training data. Here the system is overdefined. In the opposite case, *high bias*, there are too few features and too simple a model. In order to successfully apply ML to a given problem, an optimally selected feature vector and statistically significant training data are required.

In 2019, as a possible indicator of future trends, Renesas Electronics Corp. introduced a design suite for embedded artificial intelligence. This development system is intended for home appliance applications and allows for the detection of electric motor failures using MCSA and rotational information. Of particular note, the algorithm learns “normal” behavior of the system in real-time; therefore, training data are not required.

7. References

1. Palachy S. Stationarity in time series analysis: A review of the concept and types of stationarity. Medium [accessed 2020 Sep 22]. <https://towardsdatascience.com/stationarity-in-time-series-analysis-90c94f27322>.
2. Chandola V, Banerjee A, Kumar V. Anomaly detection: A survey. *ACM Computing Surveys*. July 2009;41(3):article 15.
3. Phinyomark A, Nuidod A, Phukpattaranont P, Limsakul C. Feature extraction and reduction of wavelet transform coefficients for EMG pattern classification. *Electronics and Electrical Engineering*. 2012;6(122):27–32.
4. Kuhn M, Johnson K. Feature engineering and selection: A practical approach for predictive models. 2019 June 21 [accessed 2020 Sep 22]. <http://www.feats.engineering/index.html>.
5. Jang J-SR. ANFIS: adaptive-network-based fuzzy inference system. *IEEE Transactions on Systems, Man and Cybernetics*. 1993;23(3):665–685.
6. Estima JO, Marques Cardoso AJ. A fault-tolerant permanent magnet synchronous motor drive with integrated voltage source inverter open-circuit faults diagnosis. *Proceedings of the 2011 14th European Conference on Power Electronics and Applications*; 2011.
7. Kumar MD, Kodad SF, Sarvesh B. Fault analysis of three phase inverter with induction motor load. *3rd International Conference on Electrical, Electronics, Engineering Trends, Communication, Optimization and Sciences (EEECOS)-2016*; 2016.
8. Khanniche MS, Mamat-Ibrahim MR. Fault detection and diagnosis of 3-phase inverter system. *Rev Energ Ren: Power Engineering*. 2001:69–75.
9. Mousavi S SM, Mollaeian A, Kar NC, Timusk M. A novel parallel modelling-wavelet based mechanical fault detection using stator current signature of induction machine under variable load conditions. *J Electr Eng Electron Technol*. 2017;6(2).
10. Baghli M, Delpha C, Diallo D, Hallouche A, Mba D, Wang T. Three-level NPC inverter incipient fault detection and classification using output current statistical analysis. *Energies*. 2019;12:1372.
11. Khanniche MS, Mamat-Ibrahim MR. Fault detection and diagnosis of 3-phase inverter system. *Rev Energ Ren: Power*.

12. Park J-H, Kim D-H, Kim S-S, Lee D-J, Chun M-G.. C-ANFIS based fault diagnosis for voltage-fed PWM motor drive systems. IEEE Annual Meeting of the Fuzzy Information, 2004, Processing NAFIPS '04; 2004; Alberta, Canada, vol. 1, p. 379–383.
13. Kim DE, Lee DC. Fault diagnosis of three-phase PWM inverters using wavelet and SVM. 2008 IEEE International Symposium on Industrial Electronics; Cambridge, UK; 2008, p. 329–334.
14. Aktas M, Turkmenoglu V. Wavelet-based switching faults detection in direct torque control induction motor drives. IET Science, Measurement and Technology. 2010;4(6):303–310.
15. Qiang S, Li Y. Motor inverter fault diagnosis using wavelets neural networks. 2013 IEEE International Conference on Systems, Man, and Cybernetics, Manchester; 2013, p. 3168–3173.
16. Dhumale RB, Lokhande SD, Thombare ND, Ghatule MP. Fault detection and diagnosis of high speed switching devices in power inverter. International Journal of Research in Engineering and Technology. 2015;4(2):53–257.
17. Rohan A, Kim SH. Fault detection and diagnosis system for a three-phase inverter using a DWT-based artificial neural network. International Journal of Fuzzy Logic and Intelligent Systems. 2016;16(4):238–245.

List of Symbols, Abbreviations, and Acronyms

2-D	2-dimensional
3-D	3-dimensional
AC	alternating current
ANFIS	adaptive neuro-fuzzy inference system
ANN	artificial neural network
ARL	Army Research Laboratory
CWT	continuous wavelet transform
DC	direct current
DEVCOM	US Army Combat Capabilities Development Command
DT	decision tree
DWT	discrete wavelet transform
FIS	fuzzy inference system
HPF	high-pass filter
IGBT	insulated-gate bipolar transistor
KLD	Kullback–Leibler divergence
k-NN	<i>k</i> th-nearest neighbor
LPF	low-pass filter
MCSA	motor current signature analysis
ML	machine learning
PCA	principal component analysis
THD	total harmonic distortion
STFT	short-time Fourier transform
SVM	support vector machine
WT	wavelet transform

1 DEFENSE TECHNICAL
(PDF) INFORMATION CTR
DTIC OCA

1 DEVCOM ARL
(PDF) FCDD RLD DCI
TECH LIB

11 DEVCOM ARL
(PDF) FCDD RLS D
E SHAFFER
FCDD RLS DP
M BERMAN
B ARGENIS
B GEIL
M HINOJOSA
R JANE
H O'BRIEN
A OGUNNIYI
D PORSCHE
R THOMAS
D URCIUOLI

2 DEVCOM GVSC
(PDF) J HEUVERS
D MCGREW