Automatic Hand Sign Recognition: Identify Unusuality Through Latent Cognizance

Pisit Nakjai and Tatpong Katanyukul(✉)

Khon Kaen University, Khon Kaen, Thailand
mynameisbee@gmail.com, tatpong@kku.ac.th

Abstract. Sign language is a main communication channel among a hearing disability community. Automatic sign language transcription could facilitate better communication and understanding between a hearing disability community and a hearing majority.

As a recent work in automatic sign language transcription has discussed, effectively handling or identifying a non-sign posture is one of the key issues. A non-sign posture is a posture unintended for sign reading and does not belong to any valid sign. A non-sign posture may arise during a sign transition or simply from an unaware posture. Confidence ratio (CR) has been proposed to mitigate the issue. CR is simple to compute and readily available without extra training. However, CR is reported to only partially address the problem. In addition, CR formulation is susceptible to computational instability.

This article proposes alternative formulations to CR, investigates an issue of non-sign identification for Thai Finger Spelling recognition, explores potential solutions and has found a promising direction. Not only does this finding address the issue of non-sign identification, it also provide an insight behind a well-learned inference machine, revealing hidden meaning and new interpretation of the underlying mechanism. Our proposed methods are evaluated and shown to be effective for non-sign detection.

Keywords: Hand sign recognition · Thai Finger Spelling
Open-set detection · Novelty detection · Zero-shot learning
Inference interpretation

1 Introduction

Sign language is a main face-to-face communication channel in a hearing disability community. Like spoken languages, there are many sign languages, e.g., American Sign Language (ASL), British Sign Language (BSL), French Sign Language (LSF), Spanish Sign Language (LSE), Italian Sign Language (LIS), Chinese Sign Language (CSL), Indo-Pakistani Sign Language (IPSL), Thai Sign Language (TSL), etc. A sign language usually has two schemes: a semantic sign

L. Pancioni et al. (Eds.): ANNPR 2018, LNAI 11081, pp. 255–267, 2018.
https://doi.org/10.1007/978-3-319-99978-4_20

scheme and a finger spelling scheme. A semantic sign scheme uses hand gestures, facial expressions, body parts, and actions to communicate meaning, tone, and sentiment. A finger spelling scheme uses hand postures to represent alphabets in its corresponding language. Automatic sign language transcription would allow better communication between a deaf community and hearing majority. Sign language recognition has been subjects of various studies [2,7,11]. A recent study [7], investigating hand sign recognition for Thai Finger Spelling (TFS), has discussed issues and challenges in automatic transcription of TFS. Although the discussion is based on TFS, some issues are general across languages or even general across domains beyond sign language recognition. One of the key issues discussed in the study [7] is an issue of a non-sign or an invalid TFS sign, which may appear unintentionally during a sign transition or from unaware hand postures.

The appearance of non-signs may undermine the overall transcription performance. Nakjai and Katanyukul [7] proposed a light-weight computation approach to address the issue. Sign recognition is generally based on multi-class classification, whose output is represented in softmax coding. That is, a softmax output capable of predicting one of K classes is noted $\boldsymbol{y} = [y_1 y_2 \ldots y_K]^T$, whose coding bit $y_i \in [0, 1], i = 1, \ldots, K$ and $\sum_{i=1}^{K} y_i = 1$. A softmax output $\boldsymbol{y}$ represents predicted class k when y_k is the largest component: $k = \arg\max_i y_i$. Their approach is based on the assumption that the ratio between the largest value of the coding bit and the rest shows the confidence of the model in its class prediction. Softmax coding values have been normalized so that it can be associated to both probability interpretation and cross-entropy calculation. Despite the benefits of normalization, they use the penultimate values instead of the softmax values for rationale that some information might have been lost during the softmax activation. Penultimate values are inference values before going through softmax activation (i.e., a_k in Eq. 1). Specifically, to indicate a non-sign posture, they proposed a confidence ratio (CR), $cr = \frac{a}{b}$, where a and b are the largest and second largest penultimate values, respectively: $a = a_m$ and $b = a_n$ where $m = \arg\max_i a_i$ and $n = \arg\max_{i \neq m} a_i$. Their CR has been reported to be effective in identifying a posture that is likely to get a wrong prediction. However, on their evaluating environment, they reported that CR could hardly distinguish the cause of the wrong prediction whether it was a misclassified valid sign or it was a forced prediction on an invalid sign. In addition, generally each penultimate output is a real number, $a_i \in \mathbb{R}$. This nature poses a risk on CR formulation for when there is zero or a negative number, CR can be misleading or its computation can even collapse (when the denominator is zero).

Our study investigates development of an automatic hand sign recognition for Thai Finger Spelling (TFS), alternative formulations to CR, a non-sign issue and potential mitigations for a non-sign issue. TFS has 25 hand postures to represent 42 Thai alphabets using single-posture and multi-posture schemas [7]. Single-posture schema directly associates a hand posture to a corresponding alphabet. Multi-posture schema associates a series of 2 to 3 hand postures to a corresponding alphabet. Based on probability interpretation of an inference output, Bayes theorem, and examining an internal structure of a commonly adopted inference model, various formulations alternative to CR are investigated (Sect. 3). Sections 2, 4, and 5 provide related background, methodologies and experimental results, and discussion and conclusions, respectively.

2 Background

TFS Hand Sign Recognition. A recent visual-based state-of-the-art in TFS sign recognition A-TFS [7] frames hand sign recognition as a pipeline of hand localization and sign classification problem. A-TFS is an approach based on a color scheme and a contour area using Green's theorem for hand localization. Then, an image region dominated by a hand is scaled to a pre-defined size (i.e., 64×64) and passed through a classifier, implemented with a convolution neural network. The classifier predicts the most likely class out of the 25 pre-defined classes, each corresponding to a valid TFS sign.

Most visual-based TFS sign recognition studies [7,11] focus on static images. However, a practical system should anticipate video and streaming data, where unintended postures may be passed through the pipeline and cause confusion to the final transcription result. Unintended postures can accidentally match valid signs. This challenging case is worth a dedicated study and could be addressed through a language model. However, even when the unintended postures do not match any of the valid signs, a classifier is forced to predict one out of its pre-defined classes. No matter which class it predicts, the prediction is wrong. This could cause immediate confusion on its recognition result or undermine performance of its subsequence process when using this recognition as a part of a larger system. Confidence ratio (CR) [7] was proposed to address the issue, but reported to be marginally effective.

Novelty Detection. A conventional classifier specifies a fixed number of classes that it can predict and is forced to predict. This constraint allows it to be efficiently optimized to its classification task, but it has a drawback, which is more apparent when the assumption of all-inclusive classes is strongly violated. The concept of flagging out an instance belonging to a class that an inference machine has not seen at all in the training phase is a common issue and a general concern beyond sign language recognition. The issue has been extensively studied under various terms[1], e.g., novelty detection, anomaly detection, outlier detection, zero-shot learning, and open-set recognition.

Pimentel et al. [9] summarize a general direction in novelty detection. That is, a detection method usually builds a model using training data containing no examples or very few examples of the novel classes. Then, somehow depending on approaches, a novelty score s is assigned to a sample under question $\boldsymbol{x}$ and the

[1] Definition of novelty, anomaly, outlier, and zero-shot may be slightly different. Approaches may be various [9,13], but they are generally addressing a similar concern.

final novelty judgement is decided by thresholding, i.e., the sample $\boldsymbol{x}$ is judged a novelty (belonging to a new class) when $s(\boldsymbol{x}) > \tau$ for τ is a pre-defined threshold.

To obtain the novelty score, various approaches have been examined. Pimentel et al. [9] categorize novelty detection into 5 approaches: probabilistic, distance-based, reconstruction-based, domain-based, and information-theoretic based techniques. A probabilistic approach relies on estimating a probability density function (pdf) of the data. A sample $\boldsymbol{x}$ is tested by thresholding the value of its pdf: $pdf(\boldsymbol{x}) < \tau$ indicates $\boldsymbol{x}$ being novel. Training data is used to estimate the pdf. Although this approach has a strong theoretical support, estimating a pdf in practice requires a powerful generative model along with an efficient mechanism to train it. A generative model at its fullest potential could provide greater inference capabilities on data, such as expressive representation, reconstruction, speculation, generation, and structured prediction. Its applicability is much beyond novelty detection. However, high-dimension structured data, e.g., images, render this requirement very challenging. A computationally traceable generative model is a subject of highly active research. Another related issue is to determine a sensible value for τ, in which many studies [1,3] have resorted to extreme value theory (EVT) [8]. A distance-based approach is presumably [9] based on an assumption that data seen in a training process is tightly clustered and data of new types locate far from their nearest neighbors in the data space. Either a concept of nearest neighbors [14] or of clustering [6] is used. Roughly speaking, a novelty score is defined by a distance either between a sample $\boldsymbol{x}$ and its nearest neighbors or between $\boldsymbol{x}$ and its closest cluster centroids. The distance is often measured with Euclidean or Mahalanobis distance. The approach relies on a mechanism to identify the nearest neighbors or the nearest clusters. This usually is computationally intensive and becomes a key factor attributed to its scalability issue in terms of data size and data dimensions. A reconstruction-based approach involves building a re-constructive model, often called "auto-encoder," which learns to find a compact representation of input and reproduce it as an output. Then, to test a sample, the sample is put through a reconstruction process and a degree of dissimilarity between the sample and its reconstructed counterpart is used as a novelty score. Hawkins et al. [4] used a 3-hidden-layer artificial neural network (ANN) learned to reproduce its input. As an auto-encoder, a number of input nodes is equal to a number of output nodes and a number of nodes in at least one hidden layer is smaller than a number of input nodes in order to force ANN to learn a compressed representation of the data. Any sample that cannot be reconstructed well is taken for novelty, as this infers that its internal characteristics do not align with the compressed structure fine tuned to the training data. This approach may also resort to distance measurement for a degree of dissimilarity, but it does not require to search for the nearest neighbors. Therefore, once an auto-encoder is tuned, it is easier to scale up than a distance-based approach. A domain-based approach associates building a boundary of the data domain in a feature space. Any sample $\boldsymbol{x}$ is considered novelty if its location on the feature space lies outside the boundary. Schölkopf et al. [10] proposed one-class support vector machine (SVM) for novelty detection. SVM learns to build a boundary in a feature space to adequately cover most training examples, while having a user-defined parameter to control a degree to allow some training samples to be outside the boundary. This compromising mechanism is a countermeasure to outliers in the training data. The last approach—information-theoretic—involves measurement of information content in the data. It assumes that samples of novelty increase information content in the dataset significantly. As their task was to remove outliers from data, He et al. [5] used a decrease in entropy of a dataset after removal of the samples to indicate a degree of the samples being outliers. The samples were heuristically searched. Pimentel et al. [9] note that this approach often requires an information measure that is sensitive enough to pick up the effect of novelty samples, especially when a number of these samples is small. Noted that most approaches do not scale well to high-dimension structured data, like images. Novelty detection in high-dimension structured data is still in an early stage.

Based on this categorization [9], a probabilistic approach is closest to the direction we are taking. However, unlike many early works, firstly, rather than requiring a dedicated model, our proposed method builds upon a well-adopted classifier. It can be used with an already-trained model without requirement for re-training. Secondly, most works including a notable work of OpenMax [1]—whose performance achieves F-measure[2] of 0.595—determine a degree of novelty by how unlikely the sample belongs to any seen class. Another word, most previous works have to examine every probability of sample $\boldsymbol{x}$ being seen class i, $Pr[class = i|\boldsymbol{x}]$, for $i = 1, \ldots, K$, when K is a number of all seen classes. Our work follows our interpretation of a softmax output, i.e., $y_i \equiv Pr[class = i|s, \boldsymbol{x}]$, where s represents a state of being a seen class (not novelty). How likely sample $\boldsymbol{x}$ is novel then can be directly deduced.

3 Prediction Confidence and Non-sign Identification

Confidence Score (cs). To quantify confidence in classification output, our study investigates various candidates (shown in Table 1) based on that y_k associates to a probability of being class k and y_k is generally obtained through a softmax mechanism (Eq. 1). Formulation cs_1 is straightforward. Formulation cs_2 associates to a logarithm of probability. Formulation cs_3 is similar to confidence ratio [7] (CR), but with an attempt to link an empirical utility to a theoretical rationale. In addition, formulation cs_3 is preferable in terms of computational cost and stability. Formulation cs_4—a logit function—has a more direct interpretation of the starting assumption that the confidence is high when probability of the predicted class is much higher than the rest.

Latent Cognizance. Given the input image $\boldsymbol{x}$, the predicted sign in softmax coding $\boldsymbol{y} \in \mathbb{R}^K$, where K is a number of the pre-defined classes, is derived through a softmax activation: for $k = 1, \ldots, K$,

[2] Tested on $80,000$ images (including $15,000$ unknown images).

Table 1. Formulations under investigation for confidence score (cs). Softmax value $y_l = \frac{e^{a_l}}{\sum_{i=1}^{K} e^{a_i}}$, where K is a number of predefined classes; a_l is a penultimate value; k and j are indices of the largest and the second largest components, respectively

Confidence score	$cs_1 = y_k$	$cs_2 = a_k$	$cs_3 = \log\left(\frac{y_k}{y_j}\right) = a_k - a_j$	$cs_4 = \log\left(\frac{y_k}{1-y_k}\right)$

Table 2. Formulations under investigation for cognizance function ($\tilde{g}$). Term a represents a penultimate value

Cognizance function	$\tilde{g}_0(a) = a$	$\tilde{g}_1(a) = e^a$	$\tilde{g}_2(a) = a^2$	$\tilde{g}_3(a) = a^3$	$\tilde{g}_4(a) = \|a\|$

$$y_k = \frac{e^{a_k}}{\sum_{i=1}^{K} e^{a_i}}, \tag{1}$$

where a_k is the k^{th} component of penultimate output. Each y_k can be interpreted as a probability that the given image belongs to sign class k, or more precisely a probability that the given valid input belongs to class k. That is, $y_k \equiv Pr[k|s, \boldsymbol{x}]$ where k indicates one of the K valid classes, $\boldsymbol{x}$ is the input under question, and s indicates that $\boldsymbol{x}$ is representing one of the valid classes (being a sign). For conciseness, conditioning on $\boldsymbol{x}$ may be omitted, e.g., $y_k \equiv Pr[k|s, \boldsymbol{x}]$ may be written as $y_k = Pr[k|s]$. Noted that, this insight is distinct to a common interpretation [1] that a softmax coding bit y_k of a well-learned inference model estimates probability of being in class k, i.e., $y_k = Pr[k|\boldsymbol{x}]$. This common notion does not emphasize its conditioning on an inclusiveness of all pre-defined classes.

Identifying a non-sign can be achieved through determining the probability of a sample $\boldsymbol{x}$ not belonging to any of the sign classes: $Pr[\bar{s}|\boldsymbol{x}] = 1 - Pr[s|\boldsymbol{x}]$. To deduce $Pr[s|\boldsymbol{x}]$, or concisely $Pr[s]$, consider Bayesian relation: $Pr[k|s] = \frac{Pr[k,s]}{\sum_{i=1}^{K} Pr[i,s]}$ where $Pr[k, s]$ is a joint probability. Given the Bayesian relation, the inference mechanism (Eq. 1), and our new interpretation of y_k, the following relation is found:

$$\frac{e^{a_k}}{\sum_{i=1}^{K} e^{a_i}} = \frac{Pr[k, s]}{\sum_{i=1}^{K} Pr[i, s]}. \tag{2}$$

Based on Eq. 2, it should be easier to find an appropriate mapping between e^{a_k} and $Pr[k, s]$ for the interpretability of the equation. Here, we draw the assumption that penultimate value a_k relates to joint probability $Pr[k, s]$ through an unknown function $u : a_k(\boldsymbol{x}) \mapsto Pr[k, s|\boldsymbol{x}]$. Theoretically, this unknown function is difficult to exactly characterize. In practice, even without exact characteristics of this mapping, a good approximate is enough to accomplish a task of identifying a non-sign. Supposed there exists an approximate mapping g, i.e., $g(a_k) \approx Pr[k, s]$, therefore given $g(a_i)$'s (for $i = 1, \ldots, K$), a non-sign can be identified by $Pr[s|\boldsymbol{x}] = \sum_i Pr[i, s|\boldsymbol{x}] \approx \sum_i g(a_i(\boldsymbol{x}))$. Further refining, to lessen burden on enforcing proper probability properties on g, define a "cognizance" function $\tilde{g}$ such that $\tilde{g}(a_i(\boldsymbol{x})) \propto g(a_i(\boldsymbol{x}))$. Consequently, define primary and secondary latent cognizance as the following relations, respectively:

$$\tilde{g}\left(a_i(\boldsymbol{x})\right) \propto Pr[i, s|\boldsymbol{x}], \tag{3}$$

$$\sum_i \tilde{g}\left(a_i(\boldsymbol{x})\right) \propto Pr[s|\boldsymbol{x}]. \tag{4}$$

Various formulations (Table 2) are investigated for an effective cognizance function. Identity $\tilde{g}_0$ is chosen for its simplicity. Exponential $\tilde{g}_1$ is chosen for its immediate reflection on Eq. 2. It should be noted that a study on a whole family of $\tilde{g} = m \cdot e^a$, where m is a constant, is worth further investigation. Other formulations are intuitively included on an exploratory purpose.

4 Experiments

Various formulations of confidence score and choices of cognizance function are evaluated on TFS sign recognition system. Our TFS sign recognition follows the current state-of-the-art in visual TFS sign recognition [7] with a modification of convolution neural network (CNN) configuration and its input resolution. Instead of a 64 × 64 gray-scale image, our work uses a 128 × 128 color image as an input for CNN. Our CNN configuration uses a VGG-16 [12] with the 2 fully-connected layers each having 2048 nodes, instead of 3 fully-connected layers in the original VGG-16. Figure 1 illustrates our processing pipeline.

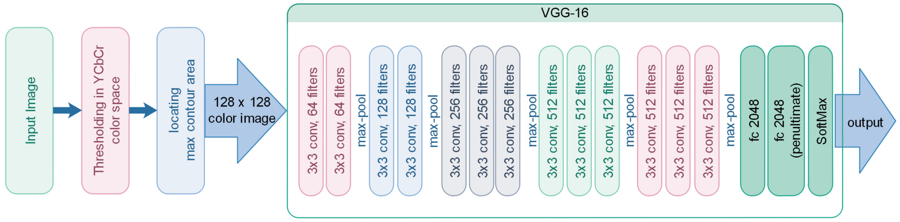


Fig. 1. Processing pipeline of our TFS sign recognition.

Sign Data. The main dataset contains images of 25 valid TFS sign postures. Twelve signers[3] were employed to perform TFS signs. Each signer performed all 25 valid TFS signs for 5 times. That resulted in a total number of 1500 images (5 times ×25 postures ×12 signers), which were augmented to 15000 images. The augmentation process generated new images from the original images using different image processing methods, e.g., skewing, scaling, rotating, and translating. All augmented images were visually inspected for human readability and semantic integrity. Every image is a color image with a resolution of approximately 800 × 600 pixels.

[3] A signer is an individual person who performs TFS signs.

Experimentation. The data was separated based on signers into a training set and a test set, i.e., 11250 images from 9 signers for training set (75%) and 3750 images from the other 3 signers for test set (25%). The experiments were conducted for 10 repetitions in a 10-fold manner. Specifically, each repetition separated data differently, e.g., the 1^{st} fold used data from signers 1, 2, and 3 for test and used the rest for training; the 2^{nd} fold used test data from signers 2, 3, and 4; and so on till the last fold using test data from signers 10, 1, and 2.

The mean Average Precision (mAP), commonly used in object detection [7], is a key performance measurement. Area under curve (AUC) and receiver operating characteristic (ROC) are used to evaluate effectiveness of various formulations for confidence score and latent cognizance. AUC is often referred to as an estimate area under Precision-Recall curve, while ROC is usually referred to an estimate area under Detection-Rate–False-Alarm-Rate curve. However, generally both areas are equivalent. We use them to differentiate the purpose of our evaluation rather than taking them as different metrics. AUC is used for identification of samples not to be correctly predicted[4]. It is more direct to measure a quality of a replacement for confidence ratio (CR) [7]. ROC is used for identifying non-sign samples[5]. It is more direct to the very issue of non-sign postures.

Non-sign Data. In addition to the sign dataset, a non-sign dataset containing images of various non-sign postures is used to evaluate non-sign identification methods. All non-sign postures were carefully choreographed to be perceivably different from any valid TFS sign and performed by a signer before augmented to 1122 images. All augmented images had been visually inspected that they all were readable and did not accidentally match to any of the 25 valid signs.

Results. Table 3 shows TFS recognition performance of the previous studies and our work. The high performing mAP (97.59%) indicates that our model is well-trained. The results were shown to be non-normal distributed, based on Lilliefors test at 0.05 level. Wilcoxon rank-sum test was conducted on each treatment for comparing (1) difference between correctly classified samples (`CP`) and misclassified samples (`IP`), (2) difference between `CP` and non-sign samples (`NS`), and (3) difference between `IP` and `NS`. At 0.01 level, Wilcoxon rank-sum test confirmed all 3 differences in all treatments. Figure 2 shows boxplots of all treatments. Y-axes show the treatment values, e.g., the top left plot has its Y-axis depicting values of $\frac{a_k}{a_j}$. The values are observed in 10 cross-validations each testing on 4872 images (3750 sign images and 1122 non-sign images). Hence, each subplot depicts 48720

[4] Positive is defined to be a sample of either a non-sign or an incorrect prediction.

[5] Positive is defined to be a non-sign.

data points[6] categorized into 3 groups. Although the significance tests confirm that the 3 groups are distinguishable using any of the treatments, the boxplots show a wide range of degrees of difficulty to distinguish each individual sample, e.g., cubic cognizance ($\sum_i a_i^3$) seems to be easier than others on thresholding the 3 cases. To measure a degree of effectiveness, Tables 4 and 5 provide AUC and ROC. Noted that, since treatment $\tilde{g}_0$ gives results in a different manner than others: a higher value associates to a non-sign (c.f. a lower value in others), the evaluation logic is adjusted accordingly.

On finding an alternative to CR [7], maximal penultimate output a_k appears promising with the largest AUC (0.934) and it is simple to obtain (no extra computation, thus no risk of computational instability). On addressing a non-sign issue, cubic cognizance a^3 gives the best ROC (0.929). Its smoothed estimate densities[7] of non-sign samples (`NS`) and sign samples (combining `CP` and `IP`) are shown on Fig. 3a. Plots of detection rate versus false alarm rate of the 4 strongest candidates and CR are shown in Fig. 3b. Table 6 shows non-sign detection performance of the 4 strongest cognizance functions compared to a baseline, CR. Non-sign detection performance is measured with accuracy—a ratio of correctly classified sign/non-sign samples to all test samples—and F-measure—a common performance index for novelty detection [1]—at thresholds selected so that every treatment has its False Alarm Rate closest to 0.1.

Table 3. Performance of visual-based TFS sign recognition.

Method	TFS coverage	Data size (# images)	Key factors	Performance
Chansri and Srinonchat [2]	16 signs	320	Kinect 3D camera, HOG and ANN	83.33%
Silanon [11]	21 signs	2100	HOG and ANN	78.00%
A-TFS [7]	25 signs	1500	Hand Extraction and CNN	91.26%
Our work (V-TFS)	25 signs	15000	Hand Extraction and VGG-16	97.59%

Table 4. Evaluation of confidence score formulations.

	$cr = \frac{a_k}{a_j}$	$cs_1 = y_k$	$cs_2 = a_k$	$cs_3 = a_k - a_j$	$cs_4 = \log\left(\frac{y_k}{1-y_k}\right)$
AUC	0.814	0.919	0.934	0.900	0.919
ROC	0.740	0.879	0.921	0.847	0.879

[6] Extreme values—under 0.25 quantile and over 0.75 quantile—were removed.

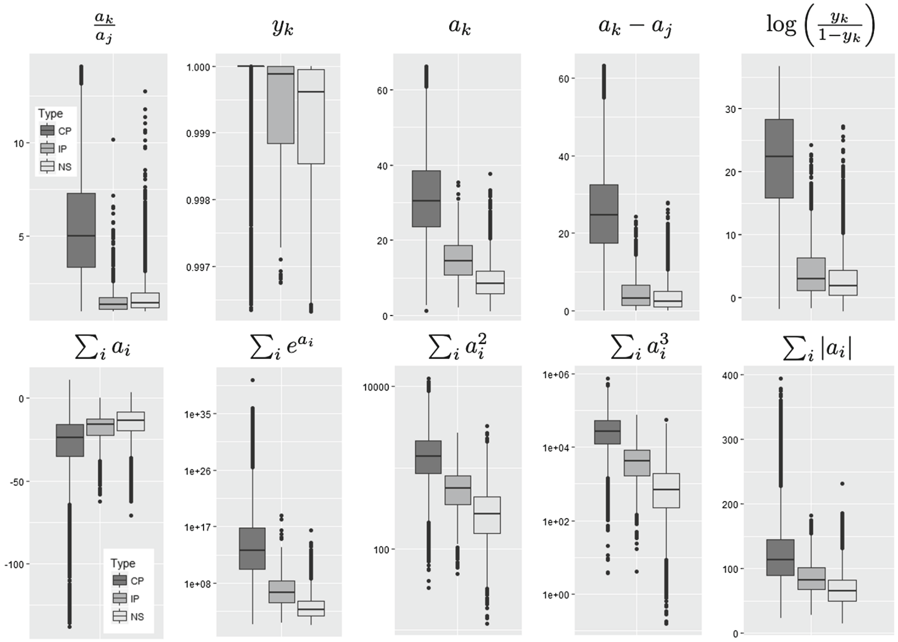
[7] A normalized Gaussian-smoothing version of histogram produced through smoothed density estimates of `ggplot2` (http://ggplot2.tidyverse.org) with default parameters.

Table 5. Evaluation of various $\tilde{g}$ formulations on $\sum_i \tilde{g}(a_i) \propto Pr[s]$.

	Identity $\tilde{g}_0(a) = a$	Exponential $\tilde{g}_1(a) = e^a$	Quadratic $\tilde{g}_2(a) = a^2$	Cubic $\tilde{g}_3(a) = a^3$	Absolute $\tilde{g}_4(a) = \|a\|$
AUC	0.437	0.930	0.855	0.934	0.737
ROC	0.419	0.920	0.845	0.929	0.726

Table 6. Non-sign detection performance of the cognizance functions c.f. CR. Thresholds were selected so that every treatment has its False Alarm Rate closest to 0.1.

Treatment	Threshold	Accuracy	F-measure
CR [7]	1.02	0.769	0.029
e^a	100000.00	0.919	0.807
a^2	26.70	0.866	0.627
a^3	1700.63	0.926	0.831
$\|a\|$	50.88	0.825	0.425



$\frac{a_k}{a_j}$ y_k a_k $a_k - a_j$ $\log\left(\frac{y_k}{1-y_k}\right)$

$\sum_i a_i$ $\sum_i e^{a_i}$ $\sum_i a_i^2$ $\sum_i a_i^3$ $\sum_i |a_i|$

Fig. 2. Upper row: boxplots of confidence ratio and candidates for confidence score. A Y-axis shows values of confidence score in linear scale. The confidence score formulations are indicated in the subplot titles. Lower row: boxplots of 5 candidates for a cognizance function. A Y-axis shows $\sum_i \tilde{g}(a_i)$ values ($\sum_i a_i$ and $\sum_i |a_i|$ in linear scale; the rest in log scale). The confidence score or cognizance values are shown in 3 groups: `CP` for correctly classified samples; `IP` for misclassified samples; `NS` for non-sign samples.

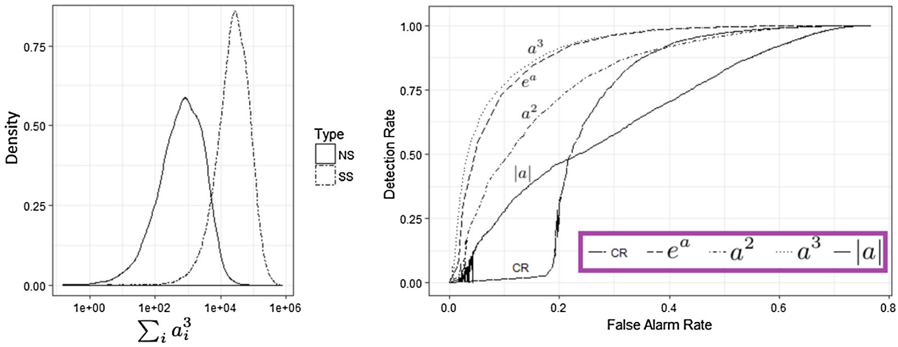


Fig. 3. (a) Left: illustration of smoothed estimated densities of sign (denoted SS) and non-sign (denoted NS) data over $\sum_i a_i^3$. (b) Right: detection rate versus false alarm rate curves of the 4 strongest candidates and the confidence ratio (CR).

5 Discussion and Conclusions

The cubic function has shown to be the best cognizance function among other candidates, including the exponential function. In addition, the cubic cognizance has ROC par to the max-penultimate confidence score. On the other hand, the max-penultimate confidence score also provide a competitive ROC and could be used to identify non-sign samples as well. Noted that OpenMax [1]—a state-of-the-art in open-set recognition—uses penultimate output as one of its crucial parts. Our finding could contribute to the development of OpenMax. A study of using cubic cognizance in OpenMax system seems promising, since it is shown to be more effective than a penultimate output. Another point worth noting is that the previous work [7] evaluated confidence score on identifying non-signs and could not confirm its effectiveness with the significance tests. Their results agree with our early experiments when using a lower resolution image, a smaller CNN structure, and training and testing on smaller datasets. In our early experiment, only a few of the treatments could be confirmed for non-sign identification. Those that were confirmed are consistent with ROC presented here. This observation implies a strong relation between state of the inference model and non-sign-identification effectiveness. This relation deserves a dedicated systematic study. Regarding applications of the techniques, thresholding can be used and a proper value for the threshold has to be determined. This can be simply achieved through tracing Fig. 3b with the corresponding threshold values. Alternatively, the proper threshold can be determined based on Extreme Value Theory, like many previous studies [1,3]. Another interesting research direction is to find a similar solution for other inference families. Our techniques target a softmax-based classifier, which is well-adopted especially in artificial neural network. However, Support Vector Machine (SVM), another well-adopted classifier, is built on a different paradigm. Application of latent cognizance to SVM might not work or might be totally irrelevant. Investigation into the issue on other inference paradigms could provide a unified insight of the underlying inference mechanism and benefits beyond addressing the novelty issue. Regarding starting assumptions, high ROC values of exponential and cubic cognizances support our new interpretation and its following assumptions. However, the penultimate output, according to our new interpretation, has relation $a_k(\boldsymbol{x}) = \log(Pr[k|s, \boldsymbol{x}]) + C$, where $C = -\log \sum_i a_i(\boldsymbol{x})$. This relation only partially agrees with our results. High value of AUC agrees with $\log(Pr[k|s, \boldsymbol{x}])$ that a class is confidently classified, but $Pr[k|s, \boldsymbol{x}]$ alone is not enough to determine a non-sign, which needs $Pr[\bar{s}|\boldsymbol{x}]$. This implies that our research is on a right direction, but it still needs more studies to complete the picture.

In brief, our study investigates (1) alternatives to confidence ratio (CR) [7] and (2) methods to identify a non-sign. The max-penultimate output is shown to be a good replacement for CR in terms of detection performance and simplicity. Its large value associates to a sample likely to be correctly classified and vice versa. The cognizance $\sum_i a_i^3$ is shown to be a good indicator for a non-sign such that $\sum_i a_i^3(\boldsymbol{x}) \propto Pr[s|\boldsymbol{x}]$, i.e., a low value of $\sum_i a_i^3(\boldsymbol{x})$ associates to a non-sign sample. To wrap up, our findings give an insight into a softmax-based inference machine and provide a tool to measure a degree of confidence in the prediction result as well as a tool to identify a non-sign. The implications may go beyond our current scope of TFS hand-sign recognition and contribute to open-set recognition or other similar concepts. Latent cognizance is viable for its simplicity and effectiveness in identifying non-signs. These would help improve an overall quality of the translation, which in turn hopefully leads to a better understandingc among people of different physical backgrounds.

References

1. Bendale, A., Boult, T.E.: Towards open set deep networks. CoRR abs/1511.06233 (2015). http://arxiv.org/abs/1511.06233
2. Chansri, C., Srinonchat, J.: Hand gesture recognition for thai sign language in complex background using fusion of depth and color video. Procedia Comput. Sci. **86**, 257–260 (2016). https://doi.org/10.1016/j.procs.2016.05.113
3. Clifton, D., Hugueny, S., Tarassenko, L.: Novelty detection with multivariate extreme value statistics. J. Signal Process. Syst. **65**, 371–389 (2011)
4. Hawkins, S., He, H., Williams, G., Baxter, R.: Outlier detection using replicator neural networks. In: Kambayashi, Y., Winiwarter, W., Arikawa, M. (eds.) DaWaK 2002. LNCS, vol. 2454, pp. 170–180. Springer, Heidelberg (2002). https://doi.org/10.1007/3-540-46145-0_17
5. He, Z., Deng, S., Xu, X., Huang, J.Z.: A fast greedy algorithm for outlier mining. In: Ng, W.-K., Kitsuregawa, M., Li, J., Chang, K. (eds.) PAKDD 2006. LNCS (LNAI), vol. 3918, pp. 567–576. Springer, Heidelberg (2006). https://doi.org/10.1007/11731139_67
6. Kim, D., Kang, P., Cho, S., Lee, H., Doh, S.: Machine learning-based novelty detection for faulty wafer detection in semiconductor manufacturing. Expert. Syst. Appl. **39**(4), 4075–4083 (2011)
7. Nakjai, P., Katanyukul, T.: T. J Sign Process Syst (2018). https://doi.org/10.1007/s11265-018-1375-6

8. Pickands, J.I.: Statistical inference using extreme order statistics. Ann. Stat. **3**(1), 119–131 (1975). https://doi.org/10.1214/aos/1176343003
9. Pimentel, M., Clifton, D., Clifton, L., Tarassenko, L.: A review of novelty detection. Signal Process. **99**, 215–249 (2014)
10. Schölkopf, B., Williamson, R., Smola, A., Shawe-Taylor, J., Platt, J.: Support vector method for novelty detection. In: Proceedings of the 12th International Conference on Neural Information Processing Systems, NIPS 1999, pp. 582–588. MIT Press (1999)
11. Silanon, K.: Thai finger-spelling recognition using a cascaded classifier based on histogram of orientation gradient features. Comput. Intell. Neurosci. **11** (2017). https://doi.org/10.1155/2017/9026375
12. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. CoRR abs/1409.1556 (2014). http://arxiv.org/abs/1409.1556
13. Xian, Y., Schiele, B., Akata, Z.: Zero-shot learning - the good, the bad and the ugly. In: IEEE Computer Vision and Pattern Recognition (CVPR) (2017)
14. Zhang, J., Wang, H.: Detecting outlying subspaces for high-dimensional data: the new task, and performance. Knowl. Inf. Syst. **10**(3), 333–355 (2006)