Similar DSLR Processor Identification Using Compact Model Templates

Hong Cao[*†] and Alex C. Kot[*]
[*]Nanyang Technological University, Singapore
E-mail: {caoh0002, eackot}@ntu.edu.sg Tel: +65-67904946
[†]Institute for Infocomm Research, A*STAR, Singapore
E-mail: hcao@i2r.a-star.edu.sg Tel: +65-64082157

***Abstract*— Fast proliferation of digital images have aroused forensics needs to identify and verify the source class for a given image patch. Statistical forensics features are known to be useful to fingerprint traditional source classes, such as device models and brands. The key challenge is to construct compact templates based on the features that are discriminant for different classes and stable for devices of the same class. While dissimilar camera sources are relatively easy to discriminate, this paper proposes to identify very similar digital single-lens reflex (DSLR) cameras on a new source class, i.e. the digital processor used. By computing several types of derivative correlation features, an eigenfeature extraction technique is employed to learn the compact model templates. The source processor is identified through finding the nearest neighbor model template. Our compact templates achieve an average accuracy of 96.0% in identifying five DSLR processors from Canon brand using small image patches.**

I. INTRODUCTION

With advances in digital photography technology and the cost reduction, the high-end digital single-lens reflex (DSLR) cameras have become affordable and popular among the worldwide photo enthusiasts. The growing trend of images acquired by DSLR cameras has also triggered urgent forensics needs on the DSLR images such as identifying the image source class and detecting possible forgeries as both the source information and content of digital images can be electronically altered with ease for malicious purposes through the ubiquitous computers and the cheap yet powerful image editing tools.

Image source identification is the technology to identify the image source based on its content, which typically involves investigation on the hidden links between the image content and its corresponding source. Depending on the outcomes, we divide image source identification into two categories: individual device identification and identification of a class of devices sharing similar attributes, e.g. devices of the same model or of the same type. Prior works on individual device identification often extract certain defect pattern, such as sensor noise pattern [4] and sensor dust pattern [5], which are intrinsic to each individual device. Through correlating the pattern extracted from a test image with the synchronized reference patterns, the source device can be identified based on the highest correlation score. Since devices of the same model, brand, digital image processor or type often share similar software and hardware components, prior works on identification of device models or types rely on extracting some processing regularities e.g. demosaicing regularity [2-3, 6-8, 10] and color features [14], which are associated with the in-device software modules, or regularities such as noise statistics [3, 11] and image quality metrics [13, 14], which are related with the image sensor size and quality. These statistical forensics features can be used in conjunction with machine learning techniques to identify the source class. In the scenarios that the forensics analyst does not have the target camera device or do not have plenty of images from a device for learning the reference defect pattern, tracing the individual source device would be impossible. Alternatively, it would be desirable to identify the source class instead using source class templates, which are constructed using images from similar cameras. The statistical forensics regularities for source class identification are expected to generalize well so that the class template learned is applicable to numerous unseen cameras within the same class.

Previous works [6-8] have shown that the accurately detected correlation features perform well in identification of diverse image sources including different demosaicing algorithms, RAW conversion software tools, dissimilar digital still camera models and the low-end mobile camera models. However, it remains insufficiently explored among existing works for identifying very similar sources and on how to define the source classes sensibly in such a challenging context. In this paper, we study the identification of five similar high-end Canon DSLR camera processors using compact model templates. By first extracting three sets of derivative correlation features, we combine the different feature sets and construct compact model templates through learning a discriminative transformation. Comparison results show that our compact model templates achieve better classification performance than the pixel correlation features using cropped image patches.

The remainder of this paper is organized as follows. Section II details the feature extraction and reduction techniques used. Section III discusses some experimental results. Section IV concludes this paper.

II. PROPOSED METHOD

A. Derivative Correlation Features

Proprietary demosaicing algorithms are often implemented in the single-sensor DSLR cameras and each demosaicing

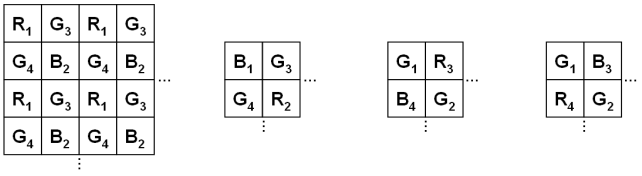

Fig. 1. Bayer color filter array and its three shifted versions, where the numbers (1-4) indicate the relative position in 2×2 periodical CFA lattice

technique can introduce some unique and persistent correlation throughout the image. Since demosaicing is not the last process in a camera processing pipeline, the correlation can be further distorted by the post-demosaicing processes such as white balancing, color space transformation, gamma correction, etc. The interplay of demosaicing and the different post-processes in a camera can leave unique image correlation patterns, which can be detected for identification of source camera classes.

We compute the derivative correlation features based on an accurate detection model [7], which is briefly described in the following steps:

1. As Bayer color filter array (CFA) has been widely used [1], we assume one of the Bayer CFAs in Fig. 1 to be the underlying CFA and separate the sensor color samples and the demosaiced samples accordingly;
2. A reverse classification is then performed to precisely divide all demosaiced samples from three color channels and at different periodical 2×2 CFA lattice positions into sixteen categories with known demosaicing axes. Each category contains the samples demosaiced by a similar formula;
3. Knowing that demosaicing a sample along an edge direction is equivalent to estimate its partial second-order derivative along that axis, we write one derivative equation for the *jth* demosaiced sample of the *ith* category,

$$e_{ij} = d''_{ij} - \mathbf{s}''^{T}_{ij}\mathbf{w}_i \qquad (1)$$

where e_{ij} is the prediction error, d''_{ij} is the partial derivative of the demosaiced sample, $\mathbf{s}''_{ij} \in \mathbb{R}^{m\times 1}$ contains the neighboring supporting derivatives computed from the sensor samples, $\mathbf{w}_i \in \mathbb{R}^{m\times 1}$ is the demosaicing weight vector and m denotes the number of weights selected. Pooling all the demosaicing equations for the *ith* category together, we can solve the optimal weights by minimizing $L2$-norm of the prediction errors using a regularized least square solution;

4. With the weights solved for each category, we compute three types of features including 316 weights (WT) for 16 categories, 64 error cumulants (EC) including mean, variance, skewness and kurtosis of the absolute errors, and 8 normalized group sizes (NGS), which characterize the distribution of demosaiced samples;
5. Steps 1-4 are repeated to cater for three other Bayer CFAs in Fig. 1. As the outcome, we have a total of 1248 WTs, 64 ECs and 32 NGSs.

Our features capture the image correlation pattern originated by demosaicing and these features also contain the distortion signature for differentiating various post-demosaicing processes for a fixed demosaicing algorithm [7].

B. Learning Compact Model Templates

We choose to learn the model template since devices of the same model are similar in terms of both hardware and software processing. In order to construct compact DSLR model templates, we learn an eigenfeature transformation for extracting a number of eigenfeatures from our combined correlation features. Our early work in [16] has shown that eigenfeature extraction technique typically results in better identification accuracy with less number of required features than the sequential forward floating feature selection technique used in [7]. Let $\{\mathbf{x}_{pq} \in \mathbb{R}^{M\times 1}$, where $1\le p\le P$ and $1\le q\le Q_p\}$ be the training feature vectors from P DSLR camera models, where Q_p feature vectors are available for the p^{th} model and M denotes dimension of the combined features. Learning of the model templates is summarized in the following steps.

1. Compute the within-model covariance matrix using

$$\mathbf{S}^{(w)} = \frac{1}{P}\sum_{p=1}^{P}\frac{1}{Q_p}\sum_{q=1}^{Q_p}\left(\mathbf{x}_{pq} - \bar{\mathbf{x}}_p\right)\left(\mathbf{x}_{pq} - \bar{\mathbf{x}}_p\right)^T \qquad (2)$$

where $\bar{\mathbf{x}}_p = \frac{1}{Q_p}\sum_{q=1}^{Q_p}\mathbf{x}_{pq}$;

2. Perform eigen decomposition on $\mathbf{S}^{(w)}$ where $\{\mathbf{v}_m^{(w)}, 1\le m\le M\}$ are the eigenvectors and $\lambda_1^{(w)} \ge \ldots \ge \lambda_M^{(w)}$ correspond to the eigenvalues;
3. Due to high feature dimension and the relatively small set of learning samples in practice, a large number of very small eigenvalues are prone to estimation errors. As the inverse of these eigenvalues are commonly used for feature scaling and measuring discriminant power of an eigen axis, their poor estimation can severely affect our selected subspace and quality of eigenfeatures. To circumvent this problem, we use an eigen spectrum regularization method to divide the entire eigen spectrum $\{\lambda_n^{(w)}\}$ into three regions, namely *signal*, *noise* and *null*. The reliable eigen values in the *signal* region are for computing parameters of an eigen spectrum model [9], which is subsequently used to regularize the unreliable eigen values in the *noise* region to improve numerical stability and discriminant performance. Eigen values for *null* region are replaced with a constant spectrum that smoothly transits from the regularized spectrum in the *noise* region;

4. Based on the regularized eigenvalues $\{\tilde{\lambda}_m^{(\mathrm{w})}\}$, we perform whitening transformation using

$$\mathbf{y}_{pq} = \boldsymbol{\Psi}_M^{(\mathrm{w})T} \mathbf{x}_{pq} \tag{3}$$

where $\boldsymbol{\Psi}_M^{(\mathrm{w})} = \left[\mathbf{v}_1^{(\mathrm{w})} \Big/ \sqrt{\tilde{\lambda}_1^{(\mathrm{w})}}, \ldots, \mathbf{v}_M^{(\mathrm{w})} \Big/ \sqrt{\tilde{\lambda}_M^{(\mathrm{w})}} \right]$.

5. Compute the total covariance matrix $\mathbf{S}^{(\mathrm{t})}$ on $\{\mathbf{y}_{pq}\}$ and perform its eigen decomposition to have the eigenvectors $\{\mathbf{v}_m^{(\mathrm{t})}\}$ with descending corresponding eigenvalues;

6. Perform principal component analysis transformation as

$$\mathbf{z}_{pq} = \boldsymbol{\Psi}_L^{(\mathrm{t})T} \mathbf{y}_{pq} = \left(\boldsymbol{\Psi}_M^{(\mathrm{w})} \boldsymbol{\Psi}_L^{(\mathrm{t})} \right)^T \mathbf{x}_{pq} \tag{4}$$

where $\boldsymbol{\Psi}_L^{(\mathrm{t})} = \left[\mathbf{v}_1^{(\mathrm{t})}, \ldots, \mathbf{v}_L^{(\mathrm{t})} \right]$ and $L \ll M$. $\mathbf{z}_{pq} \in \mathbb{R}^{L \times 1}$ is a eigenfeature vector. Note that after whitening transformation, our selected eigen axes have the large total covariance energy (i.e. eigen values of $\mathbf{S}^{(\mathrm{t})}$), which is contributed by the between-model feature differences;

7. We compute $\overline{\mathbf{z}}_p = \dfrac{1}{Q_p} \sum_{q=1}^{Q_p} \mathbf{z}_{pq}$ as the *pth* model template.

By regularizing the within-class eigen spectrum, the above procedures can efficiently address the common issues of high feature dimensionality and relatively small training sample size. Extensive experiments in [9] have shown that the regularized eigenfeatures perform more reliably than a number of conventional subspace methods in classification tasks. Our compact model templates constructed using the eigenfeatures hence reliably captures the uniqueness of each model through suppressing the within-model covariance.

C. Source Processor Identification

A simple one nearest neighbor classifier (1NNK) is used in conjunction with a correlation similarity metric to identify the source model for a test eigenfeature vector **z**. The correlation similarity between **z** and a model template **g** is computed below

$$S(\mathbf{z}, \mathbf{g}) = \left(\mathbf{z}^T \mathbf{g} \right) \Big/ \left(\|\mathbf{z}\| \cdot \|\mathbf{g}\| \right) \tag{5}$$

where $\|\cdot\|$ denotes Frobenius norm. The processor of the identified DSLR model is regarded as the processor class. Here, each of the processor class is represented as a number of DSLR model templates, which share the same processor.

III. Experiments

We first prepare a set of JPEG photos from 18 camera models of 6 brands with a total of 31 cameras. As tabulated in Table 1, we have 13 DSLR models with 26 cameras for Canon brand. Five types of DIGIC processors are employed in these 13 Canon DSLR models. The remaining models are Nikon D2X, Fujifilm Finepix S5Pro, Panasonic DMC LX1, Konica Minolta DYNAX 7D and Sony DSC T100 with one camera in each model. For each camera, we crop 200 image blocks of 512×512 within the fixed central image region. Overall, these blocks cover a large variety of scenes under different lighting conditions such as indoor, outdoor, studio and night. After performing feature extraction, we randomly select 150 blocks per camera for learning the model template and the remaining are for testing. The random apportion is repeated for five times and we report the test identification results averaged over the five different apportions in the following section.

TABLE I
Canon DSLR camera processors with different Models Used

Processor	Model	#Cameras	Resolution
DIGIC	10D	1	3072×2048
	300D	1	3072×2048
DIGIC 2	1D mark 2	2	3504×2336
	5D	5	4368×2912
	20D	1	3504×2336
	30D	1	3504×2336
	350D	1	3456×2304
	400D	3	3888×2592
Dual DIGIC 3	1D mark 3	1	3888×2592
DIGIC 3	40D	4	3888×2592
	450D	4	4272×2848
DIGIC 4	5D mark 2	1	5616×2592
	500D	1	4752×3168

A. Identification Scenarios and Feature Comparison

As shown in Fig. 2, we compare our proposed templates with those generated from the color interpolation coefficient (CIC) features [2] in three different identification scenarios: 1) identification of cameras from different brands; 2) identification of cameras of the same brand with different processor types and 3) identification of cameras using the same processor type but of different models. Note that CIC features also model the image correlation but are computed based on pixel correlation model within each of the three color channels. From our results in Fig. 2, we can observe the following: 1) For all three identification scenarios, our test error rate tends to drop quickly at the initial few eigenfeatures and stabilizes to a low level within about 10-30 eigenfeatures; 2) Based on the same number of eigenfeatures, our derivative correlation features give lower test error rate in all three identification scenarios; 3) the best achievable test error rates using our eigenfeature are 1.2%, 2.9% and 12.3% for the three scenarios in Fig. 2(a), (b) and (c), respectively. As each scenario identifies similar number of cameras, these results show that our eigenfeatures can distinguish dissimilar cameras of different brands with a high confidence. When DSLR cameras become incrementally more similar, e.g. of different brands, of the same brand and using the same

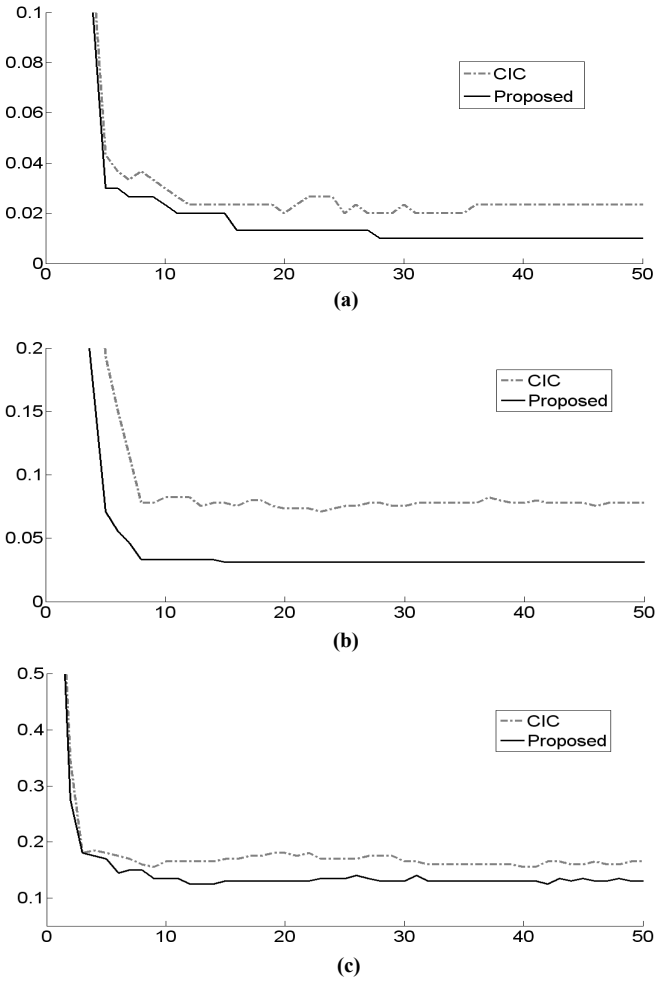

Fig. 2. Test error rate versus number of eigenfeatures in identification of (a) 6 cameras of different brands; (b) 4 Canon cameras (10D, 400D, 450D, 500D) associated with 4 different DIGIC processors; (c) 4 Canon cameras of different models (1D mark2, 5D, 30D and 400D) associated with the same DIGIC2 processor

version of DIGIC processor, our identification rate for each camera drops. Especially, we observe an increment of about 9.4% error rate in identifying four DSLR cameras of the same DIGIC2 processor. This is because the image correlation is largely associated with the in-camera software processing. Thus, cameras equipped with the same processor are expectedly harder to classify using statistical correlation features.

B. Processor and Model Identification

We further test the identification of all 13 Canon DSLR models by incrementally increasing the number of training cameras per model by one in each phase for a total of five phases. For instance, in Phase 1, 13 training cameras from 13 models with one from each model are used to learn the respective model template. In Phase 5, all 26 cameras are used where the 5D model in Table I contributing 5 cameras for learning the 5D model template and the remaining models contributing their maximally available cameras. In all 5 phases, we use the same set of test images from all 26

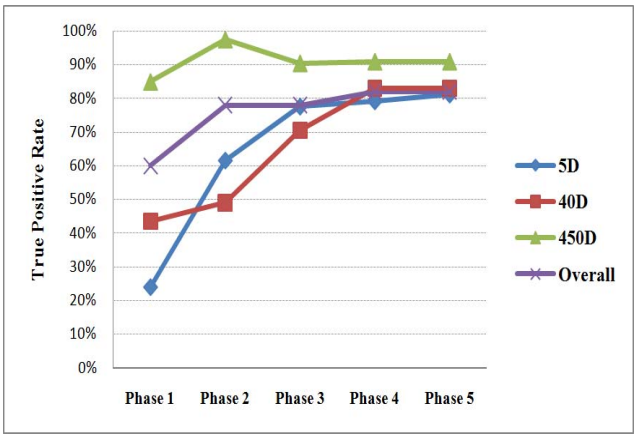

Fig. 3. True positive model identification rates when the number of cameras per model for training is incrementally increased in five phases

cameras to evaluate the performance. From the results in Fig. 3 using 25 eigenfeatures, both the overall identification rate and the rates for three separate models, for which we have 4-5 cameras each, increase when more cameras per model are used in the training. Especially, a significant increment of about 18% on the overall rate is observed from Phase 1 to Phase 2. This huge increment is because that by including more cameras per model, our computed within-model covariance better represents the common statistical differences between different cameras of the same model. Also, the increased number of training images leads to more representative model templates. The model identification errors are mainly caused by the confusion between the Canon DSLR camera models that employ the same type of DIGIC processor. We also note that 40D and 450D models, both using DIGIC 3 processor, behave differently in Fig. 3. This is due to the different learning quality at the initial phases which depends on representativeness and the content variety of our available training images. Visually, we find that the training images from our initial two 40D cameras contain less variety than the images from the first two 450D cameras. Our 450D template is hence better learnt at the initial phases with a significant portion of 40D test images being misclassified as the 450D model.

After the Phase-5 learning, we show in Table 2, the confusion matrix where the $(i,j)th$ field denotes the accuracy rate of predicting the *ith* processor class as the *jth* output processor class. We achieve a good average processor identification accuracy of 96.0% considering that these five different DIGIC processor types are all from the same Canon brand. Compared with our overall model identification rate of about 80% in Fig. 3, our good processor identification rate suggest that it is more feasible to define source classes based on the digital processor used in the context that available camera devices are very similar.

IV. CONCLUSIONS

In this paper, we proposed to construct compact model templates using eigenfeatures, which are extracted from a set

TABLE II
CONFUSION MATRIX (%) OF OUR IDENTIFICATION ACCURACY FOR 5 CANON DIGIC PROCESSORS USING 25 EIGENFEATURES, WHERE ONE TEMPLATE IS LEARNED FOR EACH MODEL AND A PROCESSOR CLASS CAN SEVERAL SEVERAL LEARNED MODEL TEMPLATES

Ave. Accuracy = 96.0%	**DIGIC**	**DIGIC 2**	**Dual DIGIC 3**	**DIGIC 3**	**DIGIC 4**
DIGIC (2 models with 2 cameras)	100	0	0	0	0
DIGIC 2 (6 models with 13 cameras)	0	99.2	0	0.5	0.3
Dual DIGIC 3 (1 model with 1 camera)	0	0	100	0	0
DIGIC 3 (2 models with 8 cameras)	0	2.5	0	97.0	0.5
DIGIC 4 (2 models with 2 cameras)	1.0	14.0	0	1.0	84.0

of high-dimensional image derivative correlation features. A processor class was then constructed as a group of the model templates, where these models share a same digital processor. Comparison results show that our features perform better than the pixel-level image correlation features. Our identification accuracy is dependent on the processing similarity inside the source cameras. The identification error rate could increase from 1.2% for identifying six cameras of different brands to 12.3% for identifying four cameras using the same processor. In identification of five similar processors, we achieved an overall accuracy of 96.0%, which is significantly higher than 81% accuracy for identifying 13 DSLR models. This suggests that using processor as the source classes is more reliable choice in the challenging source-class identification task for very similar DSLR cameras. Our results also show that the model identification accuracy can be largely improved when more training cameras per model are involved and the improvement is particularly significant when this number increases from one to two.

REFERENCES

[1] X. Li, B. Gunturk, and L. Zhang, "Image demosaicing: a systematic survey, " in *Proc. of SPIE*, vol. 6822, 2008.

[2] A. Swaminathan, M. Wu and K.J.R. Liu, "Nonintrusive component forensics of visual sensors using output images," *IEEE Trans .on Information Forensics and Security*, vol. 2(1), pp. 91-106, 2007.

[3] C. McKay, A. Swaminathan, H. Gou, and M. Wu, "Image acquisition forensics: forensic analysis to identify imaging source," in *Proc. of ICASSP*, pp. 1657-1660, 2008.

[4] M. Chen, J. Fridrich, M. Goljan and J. Lucas, "Determine image origin and integrity using sensor noise," *IEEE Trans. on Information Security and Forensics*, vol. 3(1), pp. 74-90, 2008.

[5] A.E. Dirik, H.T. Sencar and N. Memon, "Digital Single Lens Reflex Camera Identification From Traces of Sensor Dust," *IEEE Trans. on Information Security and Forensics*, vol. 3(3), pp. 539-552, 2008.

[6] H. Cao and A.C. Kot, "Raw tool identification through detected demosaicing regularity," in *Proc. of ICIP*, pp. 2885-2888, 2009.

[7] H. Cao and A.C. Kot, "Accurate detection of demosaicing regularity for digital image forensics," *IEEE Trans. on Information Security and Forensics*, vol. 4(4), pp. 899-910, 2009.

[8] H. Cao and A.C. Kot, "Mobile camera identification using demosaicing features," in *Proc. of ISCAS*, pp. 1683-1686, 2010.

[9] X. Jiang, B. Mandal and A.C. Kot, "Eigenfeature Regularization and Extraction in Face Recognition," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 30(3), pp. 383-394, 2008.

[10] S. Bayram, H.T. Sencar and N. Memon, "Classification of digital camera-models based on demosaicing artifacts," *Digital Investigations*, vol. 5(1-2), pp. 49-59, 2008.

[11] Celiktutan, B. Sankur, and I. Avcibas, "Blind Identification of Source Cell-Phone Model," *IEEE Trans. on Information Forensics and Security*, vol. 3(3), pp. 553-566, 2008.

[12] T. Filler, J. Fridrich and M. Goljan, "Using sensor pattern noise for camera model identification," in *Proc. of ICIP*, pp. 1296-1299, 2008.

[13] G. Xu, Y.Q. Shi and W. Su, "Camera brand and model identification using moments of 1-D and 2-D characteristic functions," in *Proc. of ICIP*, pp. 2917-2920, 2009.

[14] Y. Fang, A.E. Dirik, X. Sun and N. Memon, "Source class identification for DSLR and compact cameras," in *Proc. of MMSP*, pp. 1-5, 2009.

[15] T. Gloe, K. Borowka and A. Winkler, "Feature-based camera model identification works in practice*," Lecture Notes in Computer Science*, vol. 5806, pp. 262-276, 2009.

[16] H. Cao, "Statistical image source model identification and forgery detection", *Ph.D thesis*, Nanyang Technological University, 2010.