The SYSU System for the Interspeech 2015 Automatic Speaker Verification Spoofing and Countermeasures Challenge

Shitao Weng[1], *Shushan Chen*[1], *Lei Yu*[1], *Xuewei Wu*[1], *Weicheng Cai*[2], *Zhi Liu*[1], *Ming Li*[1,2]

[1]SYSU-CMU Joint Institute of Engineering, Sun Yat-Sen University, Guangzhou, China
[2]SYSU-CMU Shunde International Joint Research Institute, Guangdong, China

liming46@mail.sysu.edu.cn, mingli1@cmu.edu

Abstract

Many existing speaker verification systems are reported to be vulnerable against different spoofing attacks, for example speaker-adapted speech synthesis, voice conversion, play back, etc. In order to detect these spoofed speech signals as a countermeasure, we propose a score level fusion approach with several different i-vector subsystems. We show that the acoustic level Mel-frequency cepstral coefficients (MFCC) features, the phase level modified group delay cepstral coefficients (MGDCC) and the phonetic level phoneme posterior probability (PPP) tandem features are effective for the countermeasure. Furthermore, feature level fusion of these features before i-vector modeling also enhance the performance. A polynomial kernel support vector machine is adopted as the supervised classifier. In order to enhance the generalisation of countermeasure, we also adopted the cosine scoring and PLDA as one-class classifications. By combining the proposed i-vector systems with an OpenSMILE baseline which covers the acoustic and prosodic information further improves the final performance. The proposed fusion system achieves 0.29% and 3.265% EER on the development and test set of the database provided by the INTERSPEECH 2015 automatic speaker verification spoofing and countermeasures challenge.

Index Terms: speaker verification, spoofing, countermeasures

1. Introduction

The goal of speaker verification is to automatically verify the claimed speaker identity given a segment of speech. In the past decade, speaker verification has attracted significant research attention with promising results [1]. However, recently it is reported that many existing speaker verification systems are vulnerable against different spoofing attacks, e.g. speaker-adapted speech synthesis, voice conversion, play back, etc.[2, 3, 4, 5, 6].

Compared to text independent speaker verification, text dependent speaker verification is more robust against the play back spoofing since the speech content is constrained or predefined. Speaker-adapted speech synthesis and voice conversion are the most common spoofing methods that can convert arbitrary text or speech inputs towards the target speaker [2]. To enhance the robustness against spoofing attacks, different countermeasures have been proposed. In [7], higher-level dynamic features and voice quality assessment are used to detect those artificial signals. Furthermore, modified group delay cepstral coefficients (MGDCC) feature has been proposed to distinguish between the original and spoofed speech signals in the phase domain [8]. This approach is based on the fact that the phase information of synthetic spoofing speech is typically different from real human articulated speech while the human auditory system is less sensitive to this difference. Long term temporal modulation feature derived from magnitude or phase spectrum has also been proposed to detect the synthetic speech [9].

Total variability i-vector modeling has been widely used in speaker verification due to its excellent performance, compact representation and small model size [10, 11]. In this work, we apply the recently proposed generalized i-vector framework [12, 13, 14, 15] with both the acoustic and phonetic features to the countermeasure task.

Figure 1 shows an overview of our anti-spoofing countermeasure system. First, there are several i-vector subsystems using different features, namely acoustic level Mel-frequency cepstral coefficients (MFCC) features, the phase level MGDCC features, the phonetic level phoneme posterior probability (PPP) tandem features [14, 16] and their feature level combinations. Second, we also applied the openSMILE toolkit [17] to perform the utterance level acoustic and prosodic feature extraction. We believe that the spoofed speech signal may have different prosodic patterns. Third, after the feature normalization, multiple classification methods, e.g. cosine scoring, K-nearest neighbor (KNN), simplified PLDA [18] and Support Vector Machine (SVM), are employed as the back end. Finally, score level fusion is adopted to further enhance the overall system performance.

The remainder of the paper is organized as follows. The corpus and the proposed algorithms are explained in Sections 2 and 3, respectively. Experimental results and discussions are presented in Section 4 while conclusions are provided in Section 5.

2. Corpus

The database used to evaluate the proposed methods is based upon a standard dataset of both genuine and spoofed speech. Genuine speech is without significant channel or background noise effect and includes 106 speakers (45 male, 61 female), while spoofed speech is obtained through applying several spoofing algorithms on on the genuine speech [19]. The training data set (25 speakers, 3750 genuine utterances and 12635 spoofed utterances) is for model training while the development data (35 speakers, 3497 genuine utterances and 49875 spoofed utterances) is used to evaluate the system performance and tune the parameters. Finally, the testing data set (46 speakers, 193404 utterances) with unknown types of spoofing attacks is provided to obtain the official submission scores. The details of the database and evaluation protocol are provided in [19].

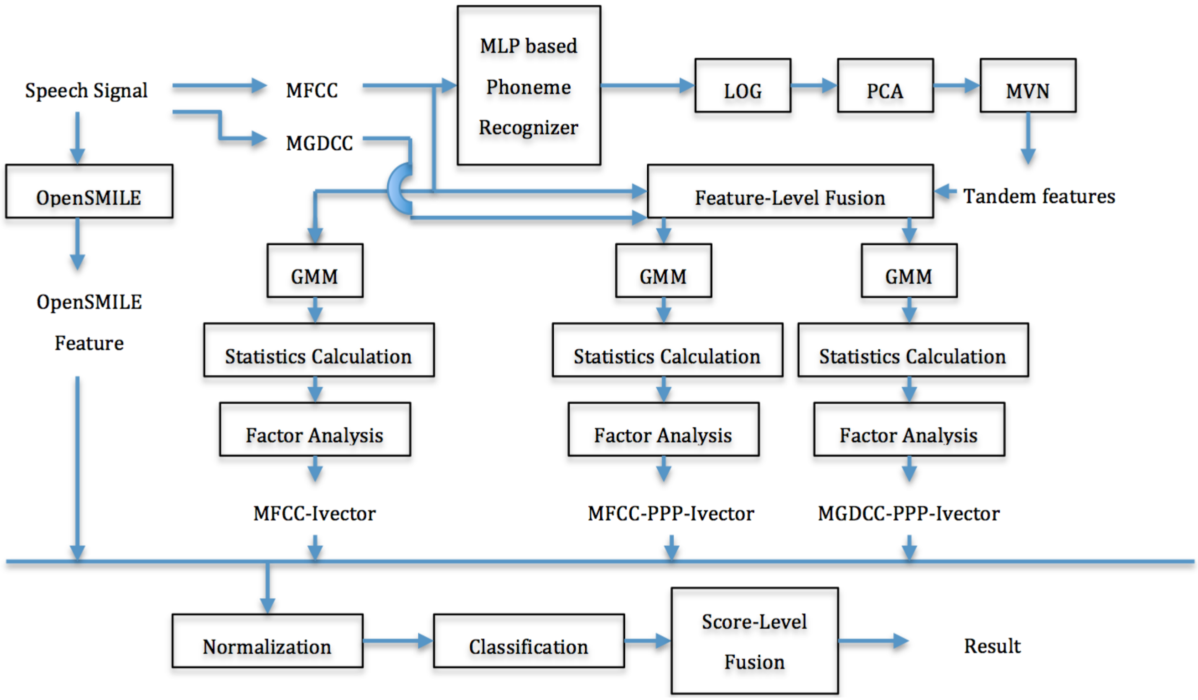

Figure 1: The system overview

3. Methods

From Figure 1, we can see that there are four different features, namely MFCC i-vectors, MFCC-PPP i-vectors, MGDCC-PPP i-vectors and openSMILE feature vectors followed by the same feature normalization, classification and score level fusion pipeline. We first present the proposed features in section 3.1. Then section 3.2 describes the supervised classification and score level fusion methods, respectively.

3.1. Features

3.1.1. The i-vector Framework

In the total variability space, there is no distinction between the speaker effects and the channel effects. Rather than separately using the eigenvoice matrix $\mathbf{V}$ and the eigenchannel matrix $\mathbf{U}$ [20], the total variability space simultaneously captures the speaker and channel variabilities [11]. Given a C component GMM UBM model λ with $\lambda_c = \{p_c, \mu_\mathbf{c}, \mathbf{\Sigma_c}\}, c = 1, \cdots, C$ and an utterance with a L frame feature sequence $\{\mathbf{y_1}, \cdots, \mathbf{y_L}\}$, the zero-order and centered first-order Baum-Welch statistics on the UBM are calculated as follows:

$$N_c = \sum_{t=1}^{L} P(c|\mathbf{y_t}, \lambda) \tag{1}$$

$$\mathbf{F_c} = \sum_{t=1}^{L} P(c|\mathbf{y_t}, \lambda)(\mathbf{y_t} - \mu_\mathbf{c}) \tag{2}$$

where $c = 1, \cdots, C$ is the GMM component index and $P(c|\mathbf{y_t}, \lambda)$ is the occupancy posterior probability for $\mathbf{y_t}$ on λ_c. The corresponding centered mean supervector $\tilde{\mathbf{F}}$ is generated by concatenating all the $\tilde{\mathbf{F}}_\mathbf{c}$ together:

$$\tilde{\mathbf{F}}_\mathbf{c} = \frac{\sum_{t=1}^{L} P(c|\mathbf{y_t}, \lambda)(\mathbf{y_t} - \mu_\mathbf{c})}{\sum_{t=1}^{L} P(c|\mathbf{y_t}, \lambda)}. \tag{3}$$

Then the centered mean supervector $\tilde{\mathbf{F}}$ can be projected as follows:

$$\tilde{\mathbf{F}} \rightarrow \mathbf{Tx}, \tag{4}$$

where $\mathbf{T}$ is a rectangular total variability matrix of low rank and $\mathbf{x}$ is the so-called i-vector [11].

3.1.2. The MFCC i-vector

The MFCC i-vector is extracted by the aforementioned i-vector framework with the acoustic level MFCC features. For cepstral feature extraction, a 25ms Hamming window with 10ms shifts was adopted. Each utterance was converted into a sequence of 36-dimensional feature vectors, each consisting of 18 MFCC coefficients and their first derivatives. We employed the English phoneme recognizer [21] to perform the voice activity detection (VAD) by simply dropping all frames that are decoded as silence or speaker noises.

3.1.3. The MFCC-PPP i-vector

It is reported in [22, 15] that by combining the phonetic level phoneme posterior probability tandem features with the acoustic level MFCC features at the feature level, the performances on speaker verification and language identification are significantly enhanced. In this work, the MFCC-PPP i-vector is extracted the same way as in [22] following the generalized i-vector framework. We employed the multilayer perceptron (MLP) based phoneme recognizer [21] with a provided English acoustic model trained on the TIMIT database to perform the phoneme decoding. The GMM model size and the tandem feature dimensionality are 512 and 32, respectively.

3.1.4. The MGDCC-PPP i-vector

The MGDCC-PPP i-vector is calculated the same way as the MFCC-PPP i-vector except that here we replace the acoustic

System	FeatureEER [dir=SE]classification method	LIBLINEAR	LIBPOLY	COSINE SCORING	KNN	Simplified PLDA	two stage PLDA
1	MFCC i-vector	8.46	6.63	16.1	9.95	12.01	17.84
2	PPP i-vector	1.72	1.26	3.6	3.4	2.29	
3	MFCC-PPP i-vector	1.86	**1.06**	2.86	2.46	1.89	10.18
4	MGDCC-MFCC-PPP i-vector	2.97	2.06	6.52	3.43	3.95	17.79
5	OPENSmile	2.03	1.57	43.41			
6	Fusion 1+2+3+4			1.63	1.37	1.09	
7	Fusion 1+2+3+4+5	0.54	**0.29**				

Table 1: Performance on development set

	S1	S2	S3	S4	S5	S6	S7	S8	S9	S10	Average
Fusion 1+2+3+4+5-LIBPOLY	0.1137	1.0332	0.0482	0.0412	0.6614	0.7112	0.2297	0.0108	0.1336	29.6649	3.265

Table 2: Performance on test set

level MFCC features with the phase domain MGDCC features. The MGDCC feature is a kind of frame-level feature focusing on the speech phase characteristics. It has been shown that phase domain features are effective for anti-spoofing countermeasures [9]. In order to calculate the MGDCC feature, we need to obtain the modified group delay function phase spectrum (MGDFPS) [23] first.

Given the data x_n of a short time window, the MGDFPS spectrum $\tau_{\rho,\gamma}(\omega)$ is calculated as follows [23]:

$$\tau_\rho(\omega) = \frac{X_R(\omega)Y_R(\omega) + Y_I(\omega)X_I(\omega)}{|S(\omega)^{2\rho}|} \tag{5}$$

$$\tau_{\rho,\gamma}(\omega) = \frac{\tau_\rho(\omega)}{|\tau_\rho(\omega)|}|\tau_\rho(\omega)|^\gamma \tag{6}$$

where $X(\omega)$ and $Y_(\omega)$ are the fourier transforms of speech signal $x(n)$ and $nx(n)$; $X_R(\omega)$ and $X_I(\omega)$ are the real and imaginary parts of $X(\omega)$; $Y_R(\omega)$ and $Y_I(\omega)$ are the real and imaginary parts of $Y(\omega)$, respectively. $|S(\omega)|^2$ is calculated by applying smoothing over $X(\omega)$ [23]. After applying the Mel-frequency filter banks and Discrete Cosine Transform, MGDCC feature is obtained. More details are provided in [9].

3.1.5. *The OpenSMILE feature vector*

The OpenSMILE feature is a 6373 dimensional utterance level feature vector extracted by the OpenSMILE toolkit [17] using the configuration file provided by the 2014 Paralinguistic Challenge [24]. Since various kinds of features, such as MFCC, loudness, auditory spectrum, voicing probability, F0, F0 envelop, jitter, and shimmer, etc., are included, this feature set can capture spoofing information at both the acoustic and prosodic levels. In our system, it served as a baseline as well as a supplement to those i-vector subsystems.

3.2. Back-end modeling

After feature vectors are extracted, we apply different classification methods as the back-end modeling.

3.2.1. *The K-nearest neighbor classification (KNN)*

KNN is a non-parametric multi-class classifier. The utterances in the training set are divided into human set and spoofed set. For each tested utterance x_t, K nearest neighbor samples are found in the training set and the score is calculated based on the class distribution of these K nearest neighbors.

3.2.2. *The cosine similarity scoring*

The cosine similarity between two vectors is calculated as follows:

$$similarity(\mathbf{x}, \mathbf{y}) = \frac{\mathbf{x}^t\mathbf{y}}{||\mathbf{x}||_2||\mathbf{y}||_2} \tag{7}$$

In our system, a mean vector of all the human utterances in the training data set is calculated. For each test utterance, the score is computed as the cosine similarity between itself and the mean vector.

3.2.3. *PLDA modeling*

We first applied the simplified PLDA modeling [18] as the back-end assuming that there are six special speakers (five spoofing channels plus one human channel), each represents a spoofing type or the original genuine speech. Second, we also adopted the two subspace (speaker subspace and spoofing subspace) PLDA presented in [25] to model the i-vectors. The scoring is based on the standard log likelihood ratio based hypothesis testing [18, 25].

3.2.4. *Support Vector Machine*

We formed the anti-spoofing countermeasure as a two class classification task for SVM modeling. The linear kernel LIBLINEAR [26] and its polynomial kernel extension LIBPOLY [27] are adopted as the back-end SVM classifiers and we applied the max/min normalization (range -1 to +1) for each feature dimension on the training, development and test sets with parameters computed only from the training data.

3.2.5. *Score fusion*

We simply employed the weighted summation fusion approach at the score level to further enhance the performance. The fusion weights were tuned on the development data set.

4. Experimental results

On the frontend, our experiments on different feature based Ivector system indicated that the PPP feature is the most efficient feature to detect spoofed uttrances in our systems. Table 3 shows that fusion on feature level can help achieve better performance. Fusing MFCC with PPP, we found that the EER was decreased from 6.63% to 1.06%. OpenSMILE works better than MFCC and MGDCC feature, the underlying reason maybe the its inclusion of prosody feature.

Methods	EER(%)
MFCC-Ivector	6.63
MFCC-PPP-Ivector	**1.06**
MGDCC-PPP-Ivector	2.23
OpenSmile	1.57

Table 3: Feature-level Fusion

polynomial kernel degree	1 (LIBLINEAR)	2 (LIBPOLY)	3	4	10
EER	1.86	1.06	1.03	1.00	2.32

Table 4: Performance of polynomial kernel SVM with different degree on MFCC-PPP-Ivector

We tried several classifications to obtain a stronger countermeasure. Cosine scoring and KNN were tried on the development data set and got a good performance. On the other hand, Support Vector Machine showed its efficience on the spoofing attack. Table 1 shows that LIBPOLY is the most efficient classification on the backend, which achieved 0.29% EER on the development data set. We also tried polynomial kernel SVM with different degrees. Table 4 shows that Libpoly is much better than liblinear, while increasing the degree will not improve the performance.

We also found that Simplified PLDA has a potential stability on the unseen data testing. Its performance is worse than liblinear on the known spoofing attack. However, when we simulate an unseen attack on the training and development data set (using 4 kinds of spoofed uttrances for modeling and test on another spoored uttrances), we found that Simplified PLDA got a better performance than LIBLINEAR.

The two stage PLDA didn't work well in our experiments. A possible reason is that the two channels are not orthogonal.

Table 2 shows that our countermeasure works well on all of the known spoofing attack and most of the unknown spoofing attack. However, our system also failed to detect the spoofed uttrances from S10 data set as most of their participants. We guess it is playback attack.

Finally, after fusing results on the score level, we achieved 0.29% EER on the development set. On the test set, we achieves 0.3795% and 6.15% on the known and unknown attacks, respectively.

5. Conclusions

This paper presents an anti-spoofing countermeasure system based on a multi-feature and multi-subsystems fusion approach. By fusing the phonetic level phoneme posterior probability tandem features with the acoustic level MFCC features or the phase level MGDCC features, the system performance is significantly enhanced. Combining the proposed i-vector subsystems with the OpenSMILE baseline which covers the acoustic and prosodic information further improves the final performance. For the back-end modeling, two classes support vector machine outperforms the one class cosine similarity or PLDA scoring on the development data where the spoofing attack types are known. The one class scoring method achieves more robust performance on the unseen testing data with unknown types of spoofing techniques.

train set	test set	PLDA	LIBLINEAR
human+spoof[2,3,4,5]	human+spoof[1]	3.57	3.4
human+spoof[1,3,4,5]	human+spoof[2]	4.8	7.69
human+spoof[1,2,4,5]	human+spoof[3]	0.2	0.71
human+spoof[1,2,3,5]	human+spoof[4]	0.2	0.66
human+spoof[1,2,3,4]	human+spoof[5]	4.49	11.81

Table 5: Performance of simplified PLDA on unknown test

6. References

[1] T. Kinnunena and H. Li, "An overview of text-independent speaker recognition: From features to supervectors," *Speech Communication*, vol. 52, no. 1, pp. 12–40, 2010.

[2] Z. Wu, N. W. D. Evans, T. Kinnunen, J. Yamagishi, F. Alegre, and H. Li, "Spoofing and countermeasures for speaker verification: A survey," *Speech Communication*, vol. 66, pp. 130–153, 2015.

[3] J. Yamagishi, T. Kobayashi, Y. Nakano, K. Ogata, and J. Isogai, "Analysis of speaker adaptation algorithms for hmm-based speech synthesis and a constrained SMAPLR adaptation algorithm," *IEEE Transactions on Audio, Speech & Language Processing*, vol. 17, no. 1, pp. 66–83, 2009.

[4] Z. Wu, T. Virtanen, T. Kinnunen, E. Chng, and H. Li, "Exemplar-based unit selection for voice conversion utilizing temporal information," in *proceedings of INTERSPEECH*, 2013, pp. 3057–3061.

[5] T. Toda, A. W. Black, and K. Tokuda, "Voice conversion based on maximum-likelihood estimation of spectral parameter trajectory," *IEEE Transactions on Audio, Speech & Language Processing*, vol. 15, no. 8, pp. 2222–2235, 2007.

[6] A. Sizov, E. Khoury, T. Kinnunen, Z. Wu, and S. Marcel, "Joint speaker verification and anti-spoofing in the i-vector space," *IEEE Transactions on Information Forensics and Security*, 2015.

[7] F. Alegre, R. Vipperla, and N. Evans, "Spoofing countermeasures for the protection of automatic speaker recognition systems against attacks with artificial signals," in *proceedings of INTERSPEECH*, 2012.

[8] Z. Wu, C. E. Siong, and H. Li, "Detecting converted speech and natural speech for anti-spoofing attack in speaker recognition." in *proceedings of INTERSPEECH*, 2012.

[9] Z. Wu, X. Xiao, E. Chng, and H. Li, "Synthetic speech detection using temporal modulation feature," in *Proceedings of ICASSP*, 2013, pp. 7234–7238.

[10] N. Dehak, P. Torres-Carrasquillo, D. Reynolds, and R. Dehak, "Language recognition via i-vectors and dimensionality reduction," in *Proc. INTERSPEECH*, 2011, pp. 857–860.

[11] N. Dehak, P. Kenny, R. Dehak, P. Dumouchel, and P. Ouellet, "Front-end factor analysis for speaker verification," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 19, no. 4, pp. 788–798, 2011.

[12] Y. Lei, N. Scheffer, L. Ferrer, and M. McLaren, "A novel scheme for speaker recognition using a phonetically-aware deep neural network," in *Proc. ICASSP*, 2014.

[13] P. Kenny, V. Gupta, T. Stafylakis, P. Ouellet, and J. Alam, “Deep neural networks for extracting baum-welch statistics for speaker recognition,” in *Proc. Odyssey*, 2014.

[14] M. Li and W. Liu, “Speaker verification and spoken language identification using a generalized i-vector framework with phonetic tokenizations and tandem features,” in *Proc. INTERSPEECH*, 2014.

[15] L. D’Haro, R. Cordoba, C. Salamea, and J. Echeverry, “Extended phone log-likelihood ratio features and acoustic-based i-vectors for language recognition,” in *Proc. ICASSP*. IEEE, 2014, pp. 5379–5383.

[16] M. Li, “Automatic recognition of speaker physical load using posterior probability based features from acoustic and phonetic tokens,” in *Proc. INTERSPEECH*, 2014.

[17] F. Eyben, M. Wöllmer, and B. Schuller, “Opensmile: the munich versatile and fast open-source audio feature extractor,” in *Proceedings of the international conference on Multimedia*. ACM, 2010, pp. 1459–1462.

[18] D. Garcia-Romero and C. Y. Espy-Wilson, “Analysis of i-vector length normalization in speaker recognition systems,” in *Proceedings of INTERSPEECH*, 2011, pp. 249–252.

[19] Z. Wu, T. Kinnunen, N. Evans, and J. Yamagishi, “Asvspoof 2015: Automatic speaker verification spoofing and countermeasures challenge evaluation plan,” *Training*, vol. 10, no. 15, p. 3750, 2014.

[20] P. Kenny, G. Boulianne, P. Ouellet, and P. Dumouchel, “Joint factor analysis versus eigenchannels in speaker recognition,” *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 15, no. 4, pp. 1435–1447, 2007.

[21] P. Schwarz, P. Matejka, and J. Cernocky, “Hierarchical structures of neural networks for phoneme,” in *Proc. ICASSP*, 2006, pp. 325–328, software available at http://speech.fit.vutbr.cz/software/phoneme-recognizer-based-long-temporal-context.

[22] M. Li and W. Liu, “Speaker verification and spoken language identification using a generalized i-vector framework with phonetic tokenizations and tandem features,” in *Proceedings of INTERSPEECH*, 2014, pp. 1120–1124.

[23] D. Zhu and K. K. Paliwal, “Product of power spectrum and group delay function for speech recognition,” in *Proceedings of ICASSP*, 2004, pp. 125–128.

[24] B. Schuller, S. Steidl, A. Batliner, F. Epps, J.and Eyben, F. Ringeval, E. Marchi, and Y. Zhang, “The interspeech 2014 computational paralinguistics challenge: Cognitive & physical load,” in *Proc. INTERSPEECH*, 2014.

[25] S. Prince and J. Elder, “Probabilistic linear discriminant analysis for inferences about identity,” in *Proc. ICCV*, 2007, pp. 1–8.

[26] R.-E. Fan, K.-W. Chang, C.-J. Hsieh, X.-R. Wang, and C.-J. Lin, “Liblinear: A library for large linear classification,” *Journal of machine learning research*, vol. 9, pp. 1871–1874, 2008.

[27] Y.-W. Chang, C.-J. Hsieh, K.-W. Chang, M. Ringgaard, and C.-J. Lin, “Training and Testing Low-degree Polynomial Data Mappings via Linear SVM,” *Journal of Machine Learning Research*, vol. 11, pp. 1471–1490, 2010.