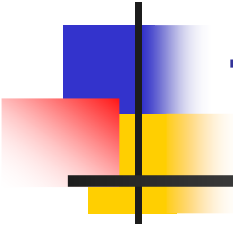


Static and Dynamic Spectral Features: Their Noise Robustness and Optimal Weights for ASR



Author : Chen Yang, Frank K. Soong, Tan Lee

Reporter: 邱聖權

Professor: 陳嘉平



Introduction

- Dynamic cepstral coefficients are more noise resistant than the static ones under various noise conditions.



Introduction

- The likelihoods contributed by static and dynamic features are weighted exponentially in the decoding process.
- A discriminative training algorithm is devised to learn the feature weights automatically from a small set of development data.

Performance comparison of static and dynamic features

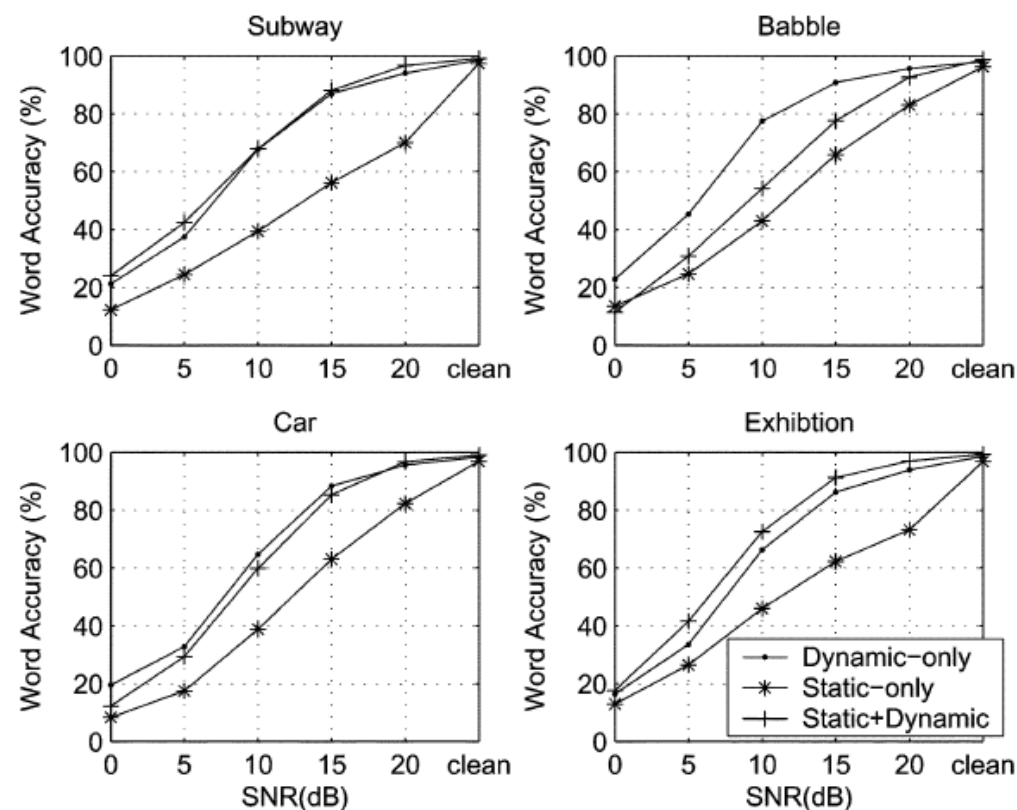


Fig. 1. Performance comparison of “static-only,” “dynamic-only,” and “static + dynamic” systems (Aurora 2).



Performance comparison of static and dynamic features

TABLE I
RECOGNITION ACCURACY AND ERROR PATTERNS OBTAINED WITH DIFFERENT
FEATURES (AURORA 2, BABBLE NOISE, 5 DB SNR)

System	Corr	Acc	Del	Sub	Ins
Static+dynamic	61.70%	30.80%	347	920	1022
Dynamic-only	45.53%	45.41%	1600	202	4
Static-only	38.09%	24.67%	973	1075	444

TABLE II
RELATIVE PERFORMANCE DIFFERENCE FROM “STATIC -ONLY” TO
“DYNAMIC-ONLY” (AURORA 2, BABBLE NOISE, 5 DB SNR)

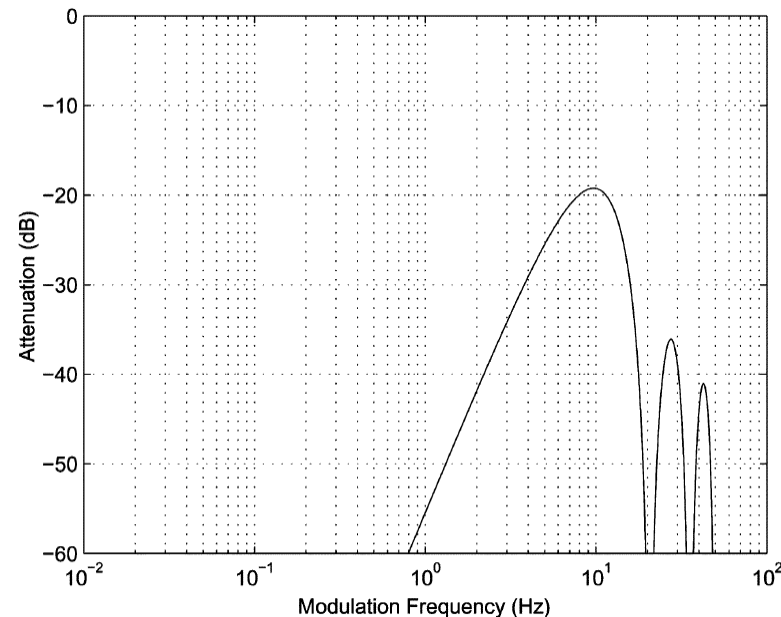
Deletion	Substitution	Insertion	Word error rate (WER)
64.44%	-81.21%	-99.10%	-27.53%

Dynamic Feature Extraction as a Filtering Process

$$\mathbf{o}_t^d = \frac{(\mathbf{o}_{t+1}^s - \mathbf{o}_{t-1}^s) + 2(\mathbf{o}_{t+2}^s - \mathbf{o}_{t-2}^s) + 3(\mathbf{o}_{t+2}^s - \mathbf{o}_{t-2}^s)}{2(1^2 + 2^2 + 3^2)}$$

$\mathbf{o}_t^d, \mathbf{o}_t^s$ denote the dynamic and static feature vectors.

$$H(z) = \frac{z^3(3 + 2z^{-1} + z^{-2} - z^{-4} - 2z^{-5} - 3z^{-6})}{2(1^2 + 2^2 + 3^2)}$$



Exponential Weighting of Likelihoods

- The output pdf at a particular HMM state is expressed as a Gaussian mixture

$$b_j(\mathbf{o}_t) = \sum_{k=1}^K c_{jk} \mathbf{N}(\mathbf{o}_t; \boldsymbol{\mu}_{jk}, \boldsymbol{\Sigma}_{jk})$$

k is mixture index, c_{jk} is mixture weight.

- Assume dynamic and static features are conditionally independent, and apply weights to each.

$$b_j(\mathbf{o}_t) = \sum_{k=1}^K c_{jk} \exp[\alpha \log \mathbf{N}(\mathbf{o}_t^d; \boldsymbol{\mu}_t^d, \boldsymbol{\Sigma}_t^d) + \beta \log \mathbf{N}(\mathbf{o}_t^s; \boldsymbol{\mu}_t^s, \boldsymbol{\Sigma}_t^s)]$$

α is dynamic feature weight, β is static feature weight, $\alpha + \beta = 1$.

Discriminative training of feature weights

- Given an utterance $\mathbf{O}_u = \{\mathbf{o}_{u1}, \mathbf{o}_{u2}, \dots, \mathbf{o}_{uT}\}$
- Using the log likelihood difference (lld) between the recognized and the correct word sequences as the optimization criterion,

$$lld(\mathbf{O}_u) = g^r(\mathbf{O}_u) - g^c(\mathbf{O}_u)$$

where $g^r(\mathbf{O}_u)$ is the log likelihood of the recognized word sequence for \mathbf{O}_u ,

$g^c(\mathbf{O}_u)$ is the log likelihood computed by forced alignment with the correct answer.

when $lld(\mathbf{O}_u) > 0$, a recognition error occurs.



Discriminative training of feature weights

- The empirical cost averaged over a set of U training utterances is given by

$$LLD = \frac{1}{U} \sum_{u=1}^U lld(\mathbf{O}_u)$$

- Using the gradient descent method to find optimal feature weights

$$\alpha(n+1) = \alpha(n) - \varepsilon \left(\frac{\partial LLD}{\partial \alpha} \right)$$

$$\beta(n+1) = \beta(n) - \varepsilon \left(\frac{\partial LLD}{\partial \beta} \right)$$



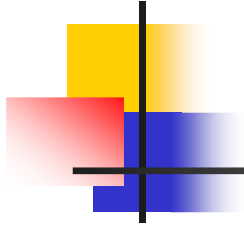
Discriminative training of feature weights

- the gradient term is computed as

$$\frac{\partial lld(\mathbf{O}_u)}{\partial \alpha} = \frac{\partial g^r(\mathbf{O}_u)}{\partial \alpha} - \frac{\partial g^c(\mathbf{O}_u)}{\partial \alpha}$$

$$\frac{\partial g(\mathbf{O}_u)}{\partial \alpha} = \sum_{t=1}^T \frac{\partial \{\log b_{q_t^*}(\mathbf{o}_{ut})\}}{\partial \alpha}$$

q_t^* denote the HMM state associated with \mathbf{o}_{ut} ,
 $b_{q_t^*}(\mathbf{o}_{ut})$ is corresponding acoustic probability.



$$\frac{\partial \{\log b_{q_t}^*(\mathbf{o}_{ut})\}}{\partial \alpha} = \frac{1}{b_{q_t}^*(\mathbf{o}_{ut})} \cdot \frac{\partial \{b_{q_t}^*(\mathbf{o}_{ut})\}}{\partial \alpha}$$

$$= \frac{1}{b_{q_t}^*(\mathbf{o}_{ut})} \times$$

$$\sum_{k=1}^K \left\{ c_{jk} \exp \left[\alpha \log N(\mathbf{o}_{ut}^d; \boldsymbol{\mu}_{q_t^*k}^d, \boldsymbol{\sigma}_{q_t^*k}^d) + \beta \log N(\mathbf{o}_{ut}^s; \boldsymbol{\mu}_{q_t^*k}^s, \boldsymbol{\sigma}_{q_t^*k}^s) \right] \cdot \log \left[N(\mathbf{o}_{ut}^d; \boldsymbol{\mu}_{q_t^*k}^d, \boldsymbol{\sigma}_{q_t^*k}^d) \right] \right\}$$



Experimental setup

- Aurora2
 - Multi-condition training data used to train feature weights
 - Clean training model used for recognition
- features with equal weighting used for baseline

Recognition Results With Condition-Specific Weights

- the weights are used on the condition that they were trained

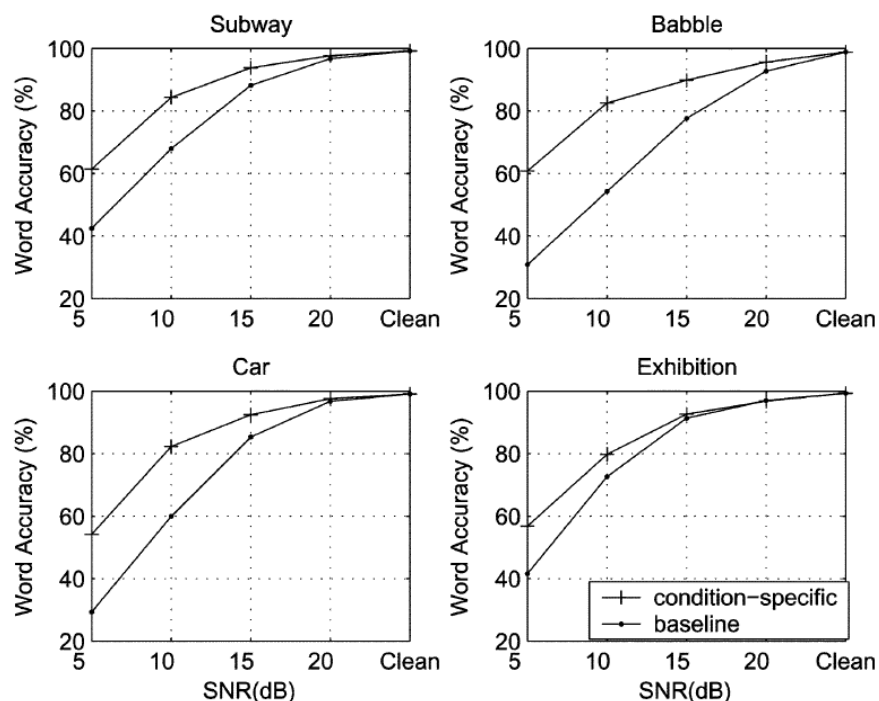


Fig. 6. Recognition performance with condition-specific feature weights (Test Set A of Aurora 2).

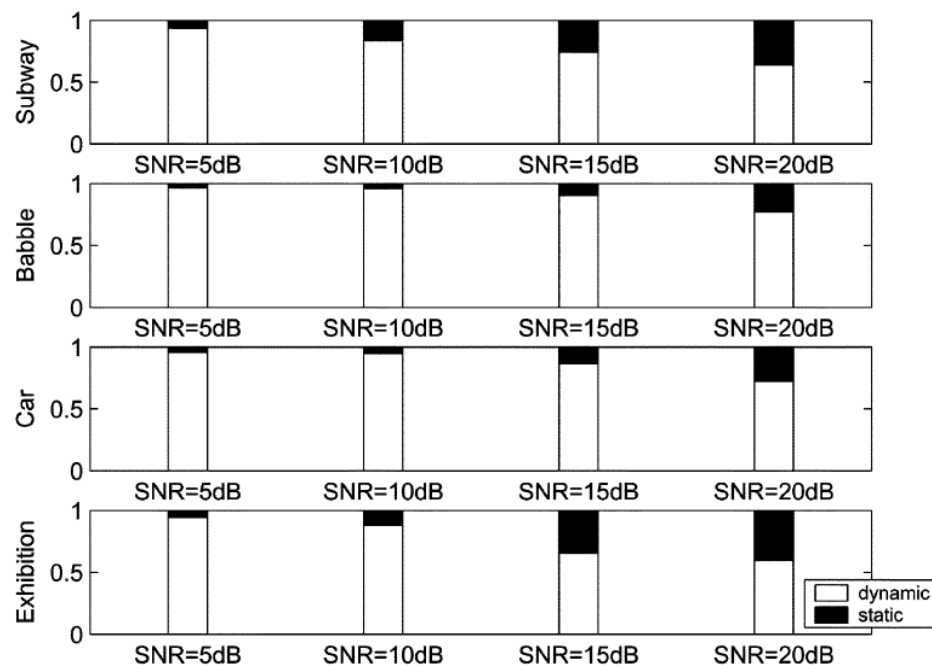


Fig. 7. Optimal condition-specific weights for different noise conditions (Test Set A of Aurora 2).

Universal Weights on Matched Conditions

- Test set A, WER reduction is 36.6%

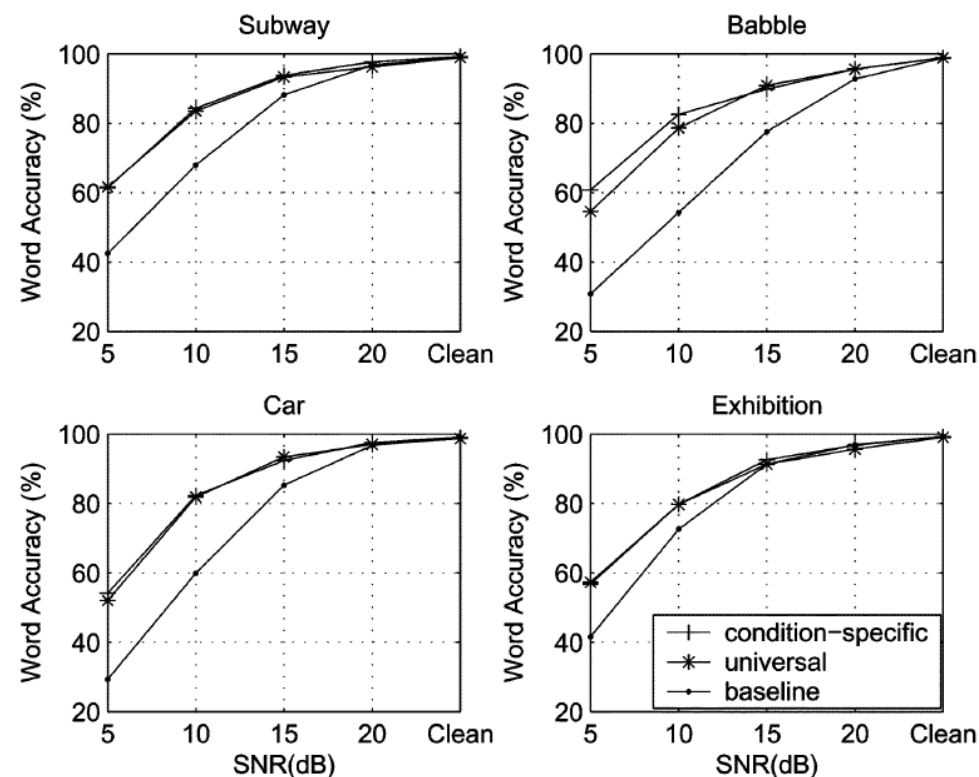
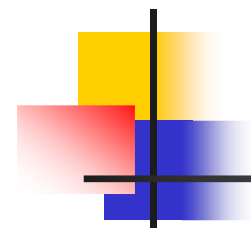


Fig. 8. Recognition performance with universal weights for matched conditions (Test Set A of Aurora 2).

Universal Weights on Mismatched Conditions



Test set B

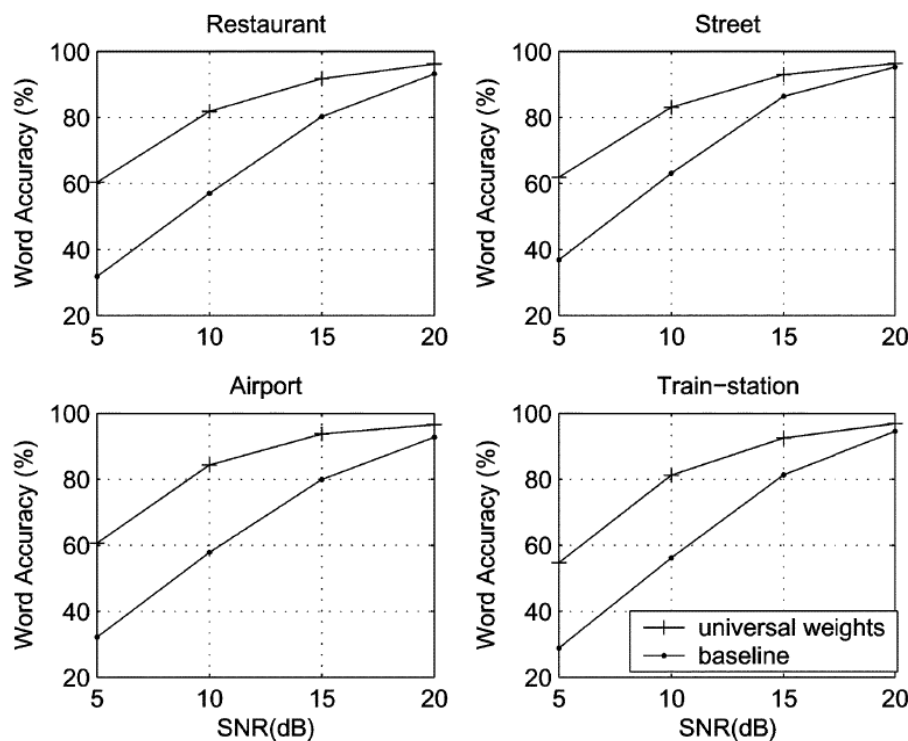


Fig. 9. Recognition performance with universal weights for mismatched noise types (Test Set B of Aurora 2).

Test set C (after CMN)

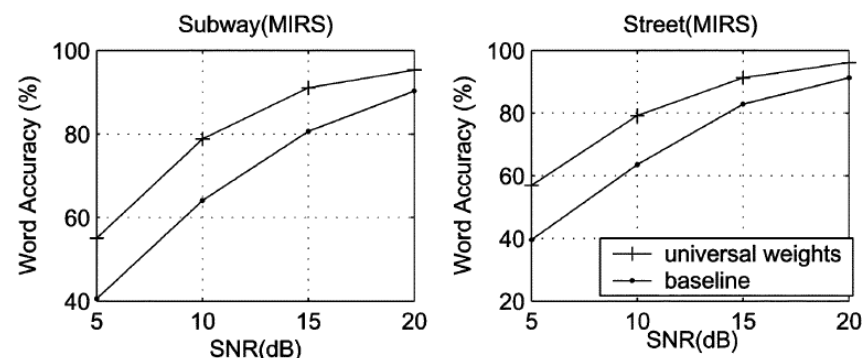


Fig. 10. Recognition performance with universal weights when there exists channel distortion (Test Set C of Aurora 2).

Set B WER reduction:49.3%

Set C WER reduction:43.4%

Inclusion of the acceleration features

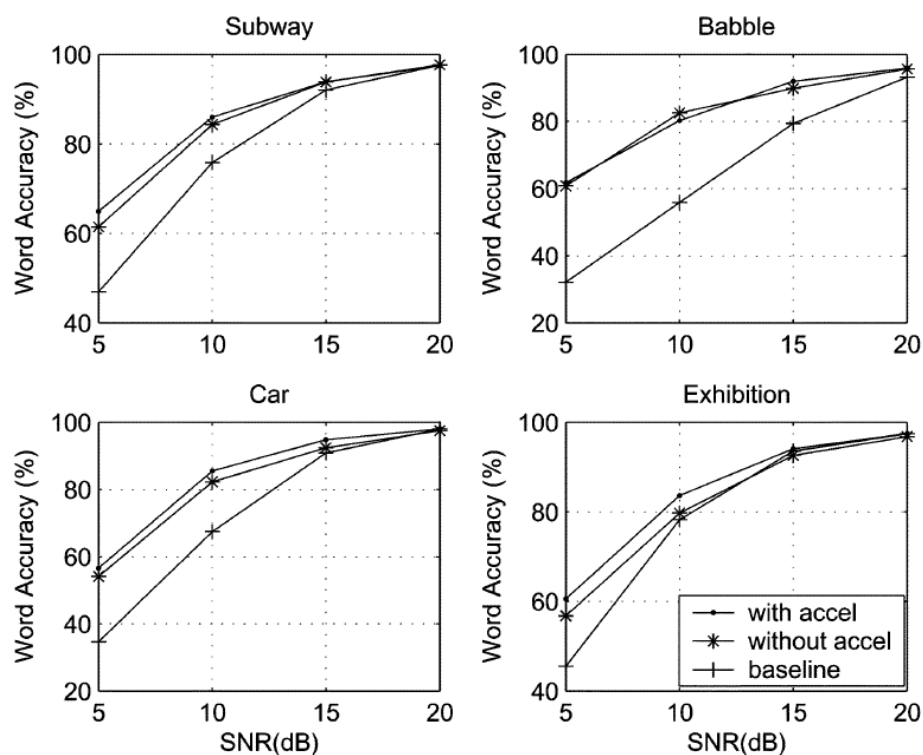


Fig. 14. Recognition performance with the inclusion of condition-specific weighted acceleration features (Test Set A of Aurora 2).

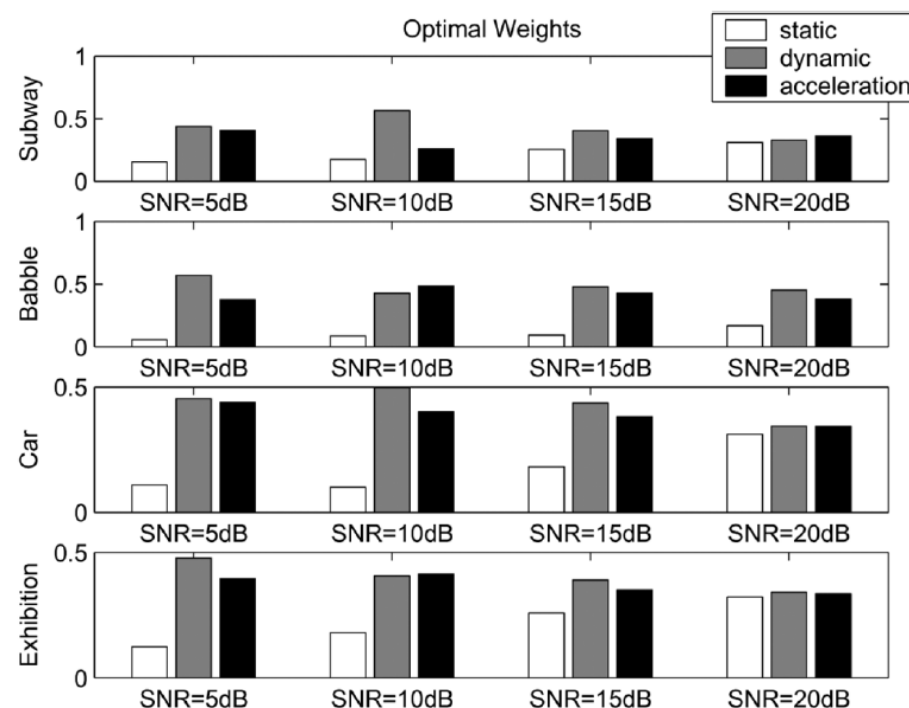


Fig. 15. Comparison of optimal condition-specific weights for static, dynamic, and acceleration features (Test Set A of Aurora 2).