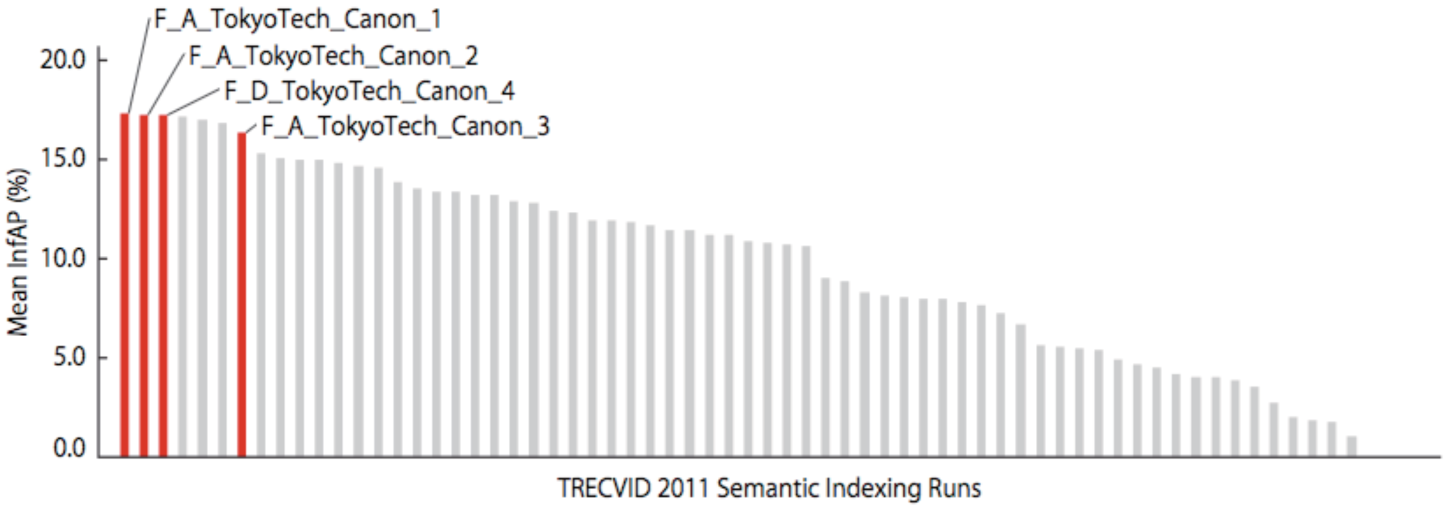
Semantic Indexing Using GMM Supervectors and Tree-structured GMMs

Nakamasa Inoue, Koichi Shinoda,
Department of Computer Science,
Tokyo Institute of Technology

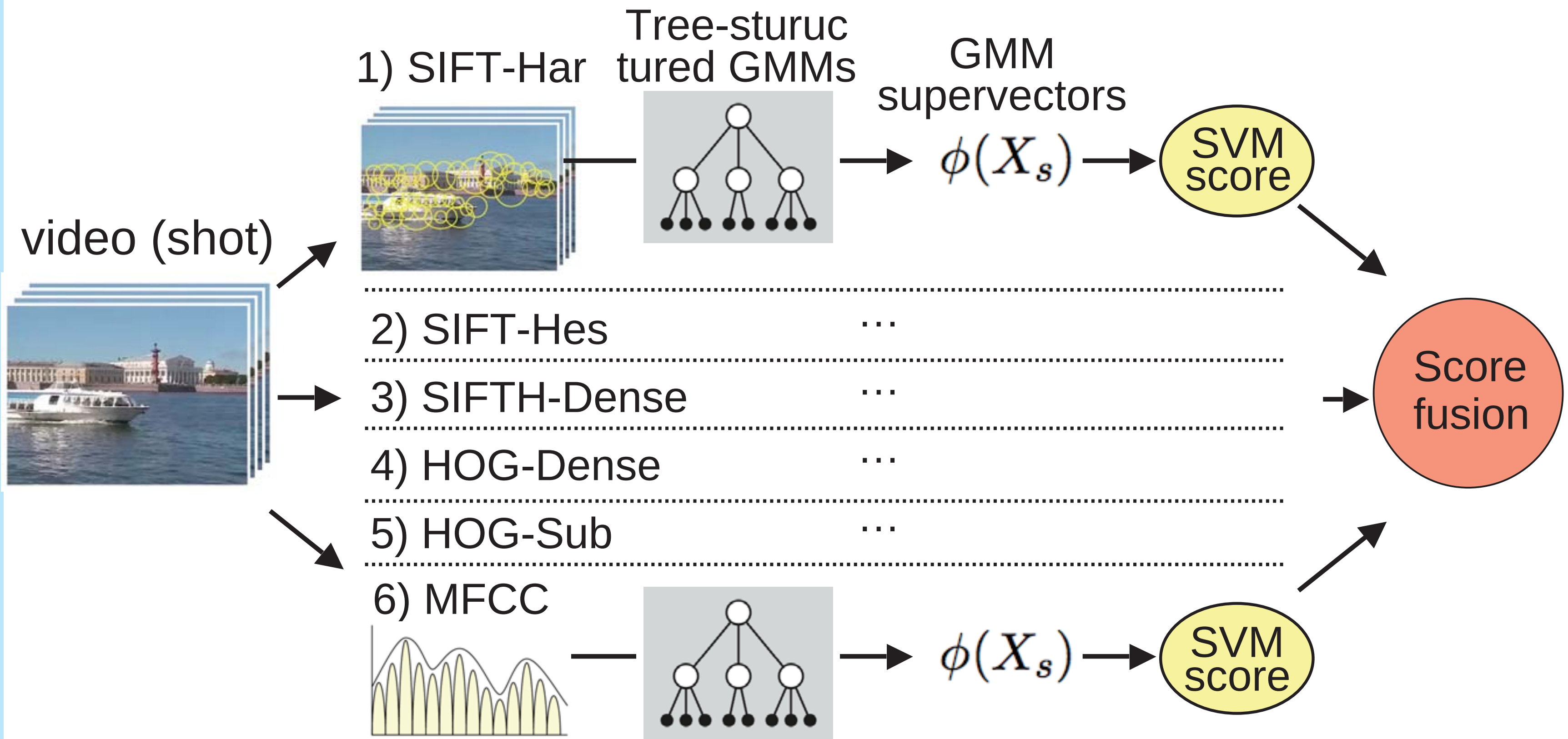
Outline

- System overview
- Fast and high-performance semantic indexing system
  - 6 types of audio and visual features
  - Gaussian mixture model (GMM) supervectors
  - Tree-structured GMMs
- Best result: Mean InfAP = 17.3%



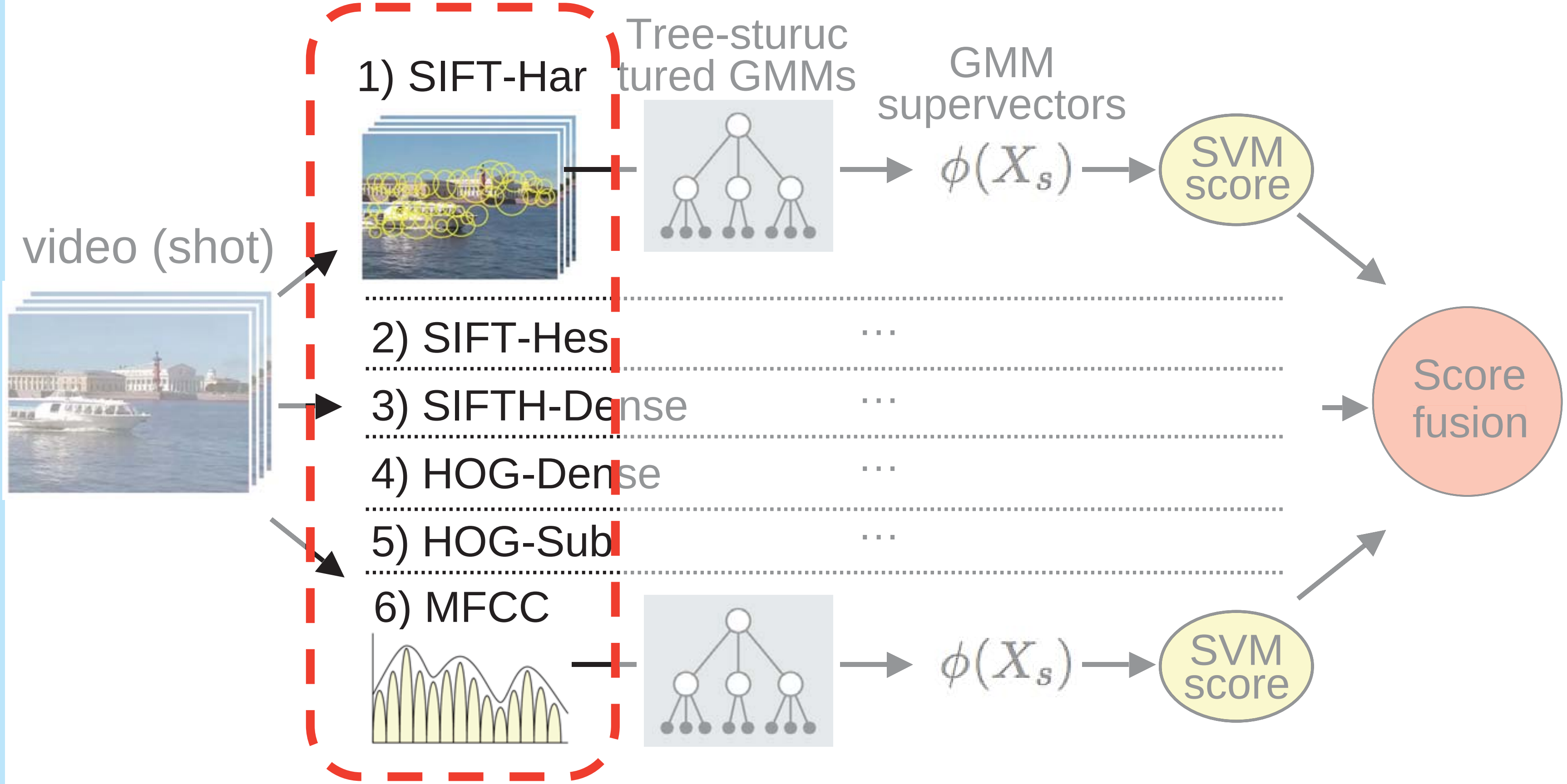
System Overview

- Fast and high-performance semantic indexing system



System Overview

- Fast and high-performance semantic indexing system



Local Feature Extraction

1) SIFT-Har

- Harris-affine detector: extension of Harris corner detector [Mikolajczyk, 2004]
- Multi-frame (every other frame)

2) SIFT-Hes

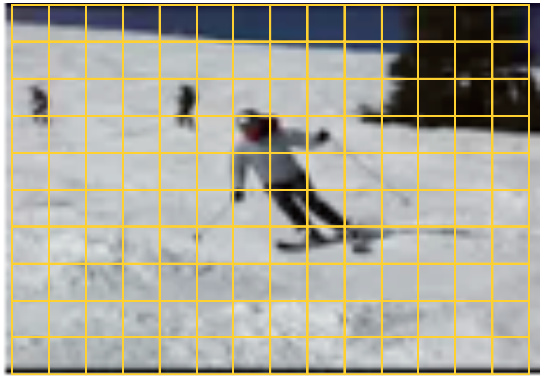
- Hessian-affine detector
- Multi-frame (every other frame)

Feature type	avg. #features per frame	avg. #features per shot
SIFT-Har	247	19,536
SIFT-Hes	240	18,986

Local Feature Extraction

3) SIFTH-Dense

- SIFT + Hue histogram
- 30,000 samples from a key-frame



4) HOG-Dense

- 32 dimensional HOG
- 10,000 samples from a key-frame

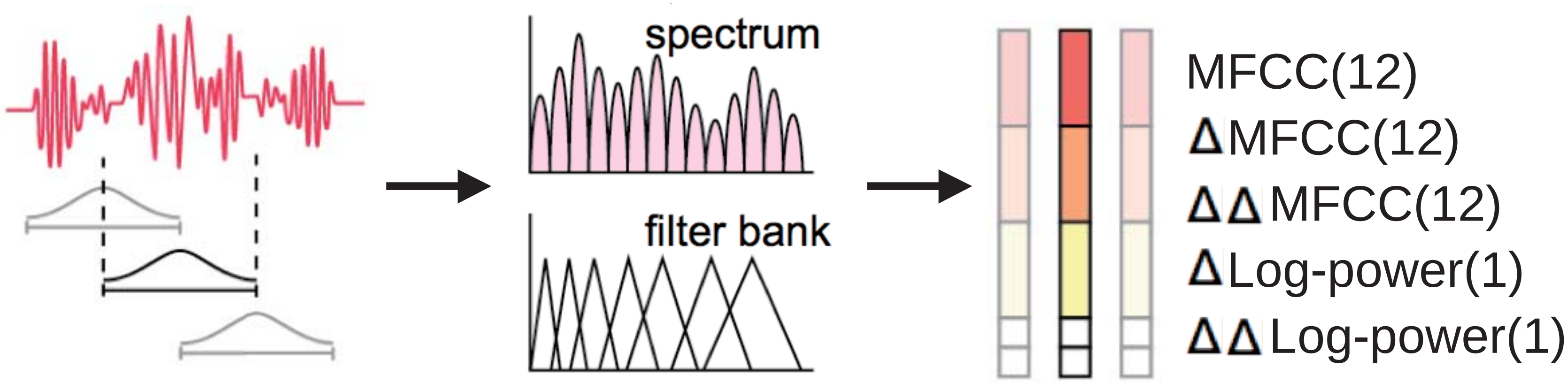
5) HOG-Sub

- Dense HOG features extracted from temporal subtraction images
- Capture movement

Local Feature Extraction

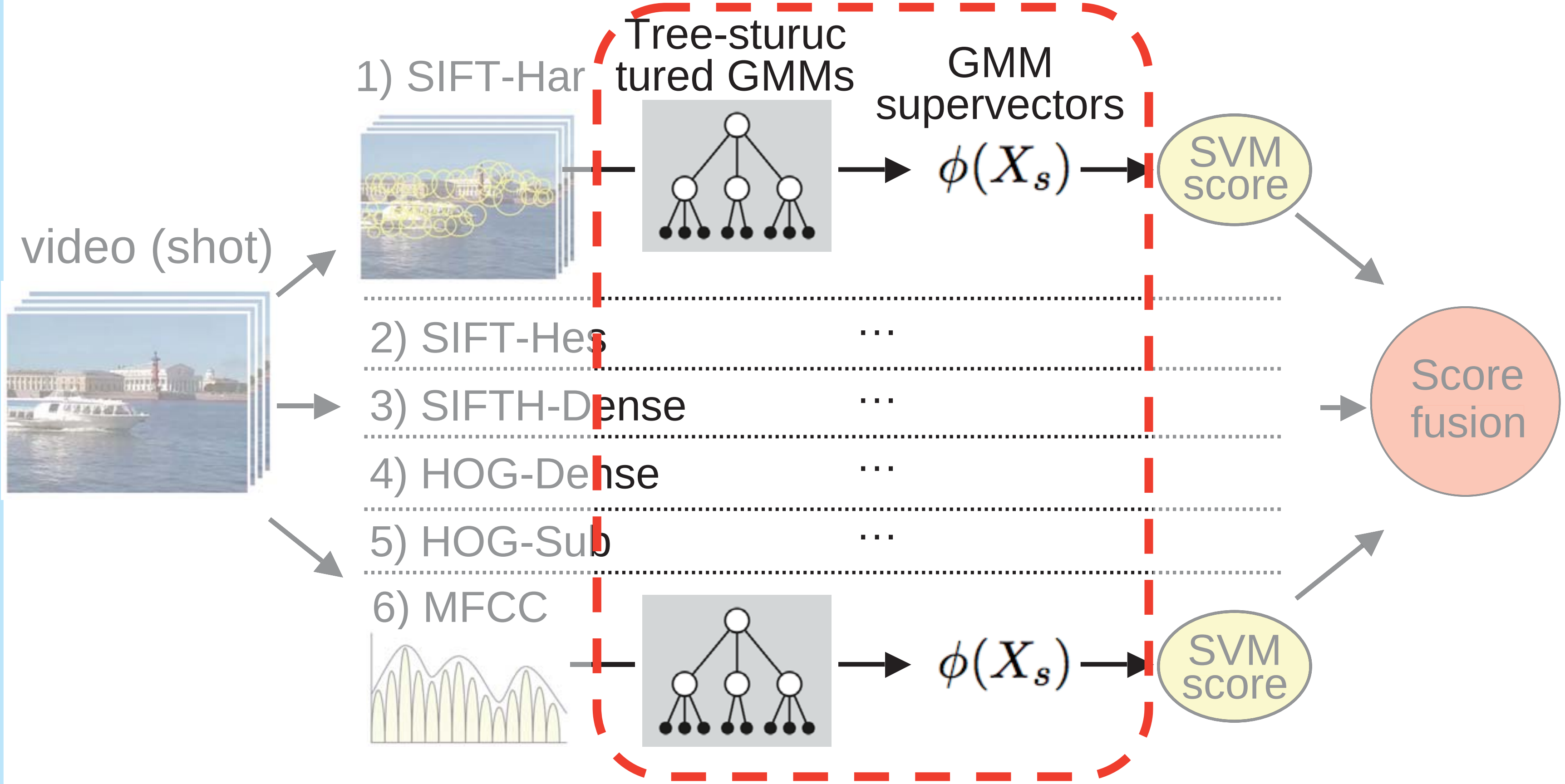
6) MFCC

- Mel-frequency cepstrum coefficients (MFCC)
- Audio features for speech recognition
- Targets: Speaking, Singing etc.



System Overview

- Fast and high-performance semantic indexing system



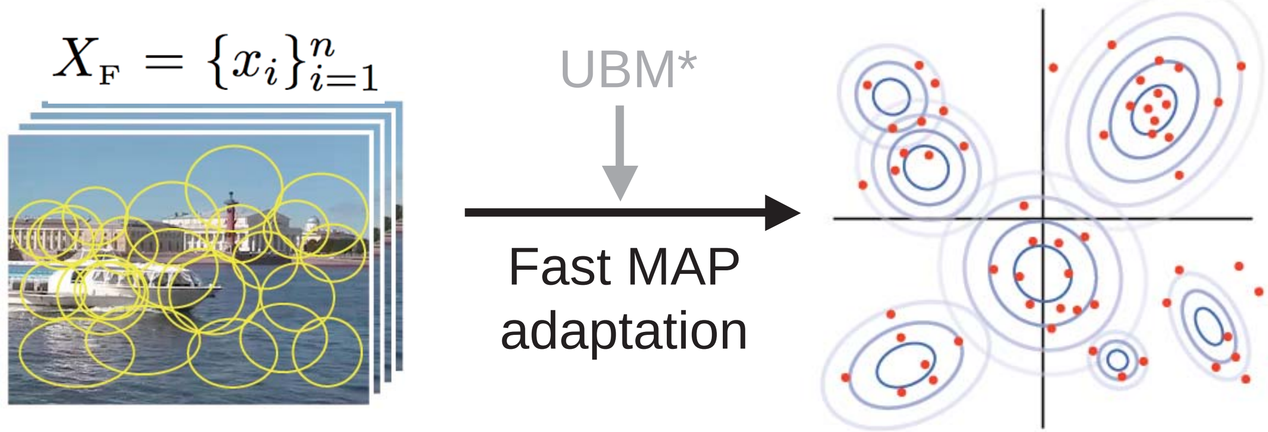
Gaussian Mixture Models (GMMs)

- Each shot is model by a GMM

  $X_{\mathrm{F}} = \{x_i\}_{i=1}^{n}$: local features

  $\theta = \{w_k, \mu_k, \Sigma_k\}_{k=1}^{K}$: GMM parameters

- GMM parameters are estimated by using fast maximum a posteriori (MAP) adaptation



*Universal background model (UBM): a prior GMM which is estimated by using all video data.

Gaussian Mixture Models (GMMs)

- (Basic) MAP adaptation for mean vectors:

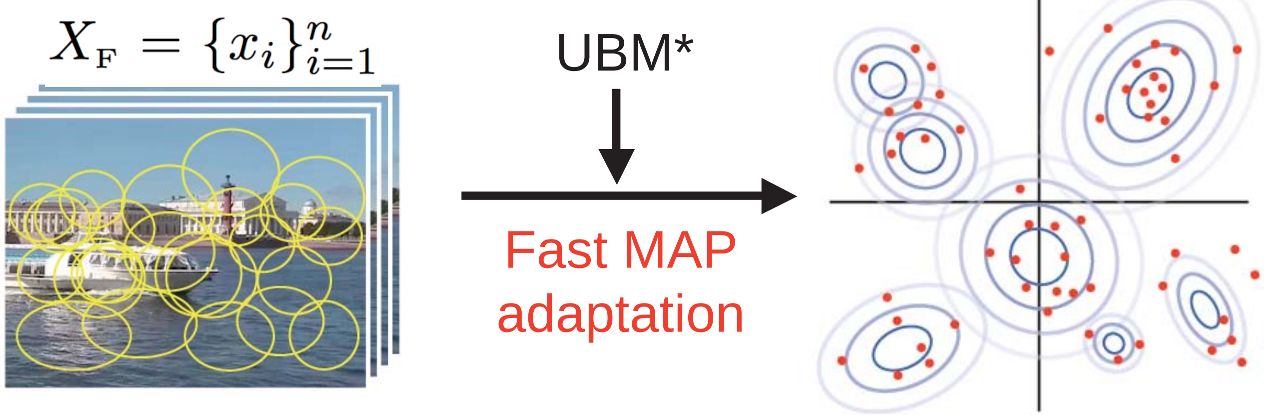
$$\hat{\mu}_k = \frac{\tau \hat{\mu}_k^{(\mathrm{U})} + \sum_{i=1}^{n} c_{ik} x_i}{\tau + C_k}$$

where

$$\underbrace{c_{ik} = \frac{w_k \mathcal{N}(x_i | \mu_k^{(U)}, \Sigma_k^{(U)})}{\sum_{k=1}^{K} w_k \mathcal{N}(x_i | \mu_k^{(U)}, \Sigma_k^{(U)})}}, \quad C_k = \sum_{i=1}^{n_s} c_{ik}$$

responsibility of component k for x_i

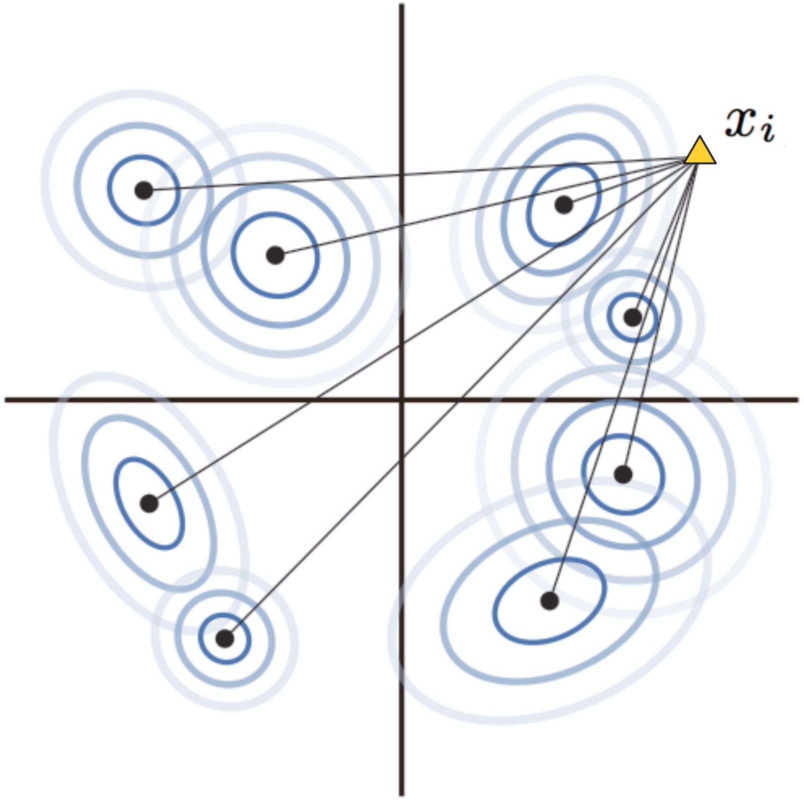
Computational cost: high



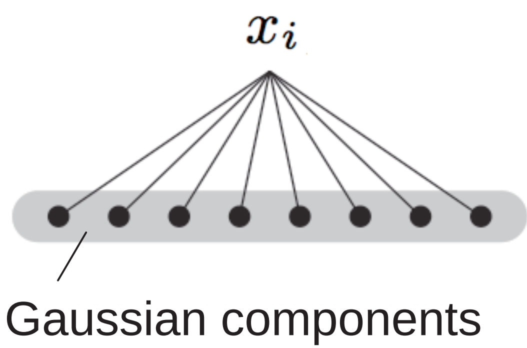
*Universal background model (UBM): a prior GMM which is estimated by using all video data.

Gaussian Mixture Models (GMMs)

- c_{ik} : *responsibility* of component k for x_i



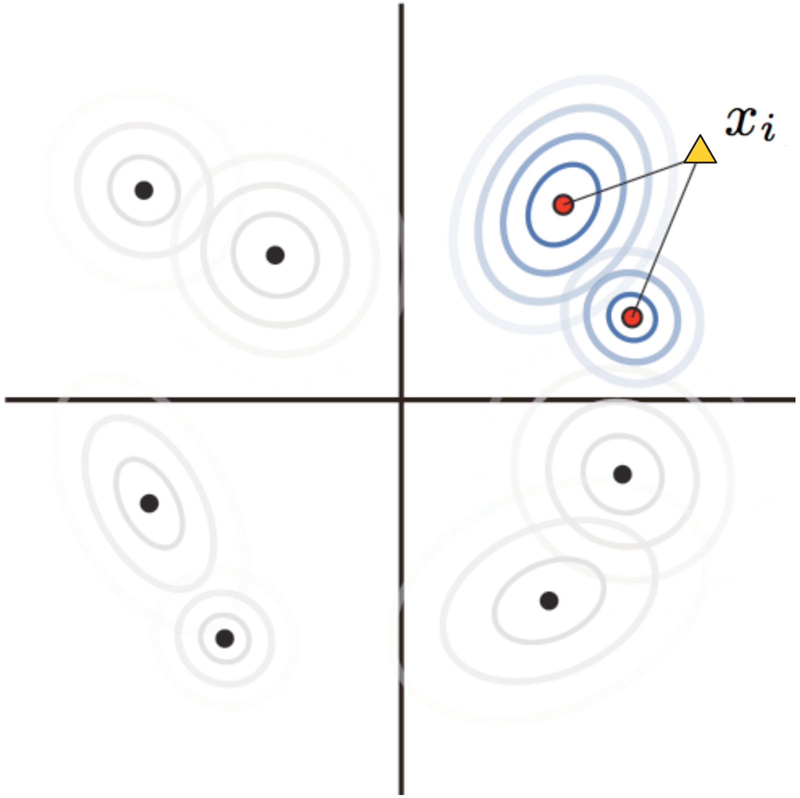
$$c_{ik} = \frac{w_k \mathcal{N}(x_i | \mu_k^{(U)}, \Sigma_k^{(U)})}{\sum_{k=1}^{K} w_k \mathcal{N}(x_i | \mu_k^{(U)}, \Sigma_k^{(U)})},$$



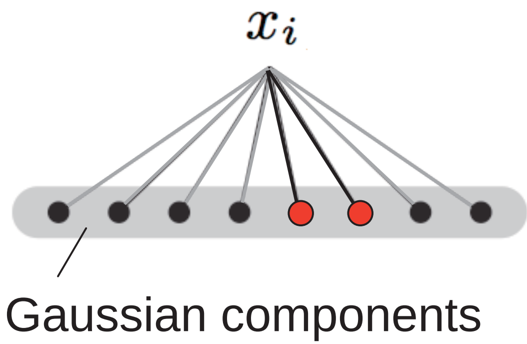
- Tree-structured GMMs calculate c_{ik} quickly!

Gaussian Mixture Models (GMMs)

- c_{ik} : *responsibility* of component k for x_i



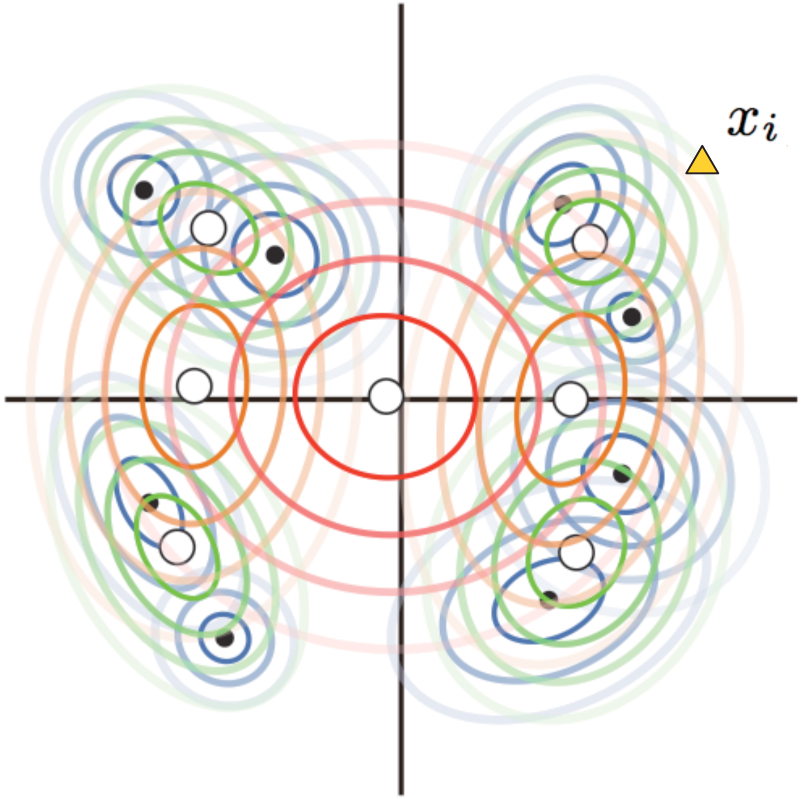
$$c_{ik} = \frac{w_k \mathcal{N}(x_i | \mu_k^{(U)}, \Sigma_k^{(U)})}{\sum_{k=1}^{K} w_k \mathcal{N}(x_i | \mu_k^{(U)}, \Sigma_k^{(U)})},$$



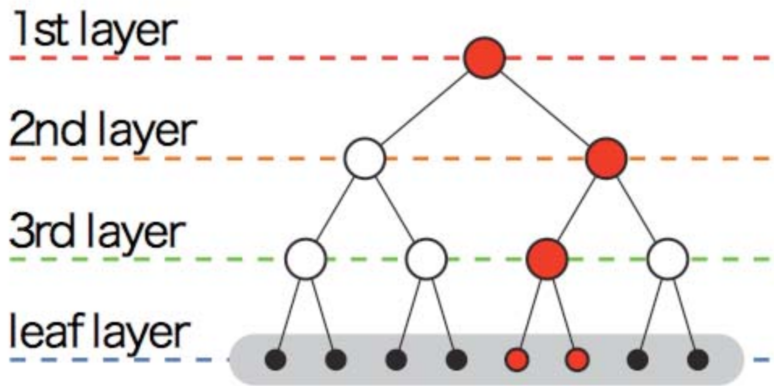
- Tree-structured GMMs calculate c_{ik} quickly!

Tree-structured GMMs

- Calculate responsibilities c_{ik} quickly.



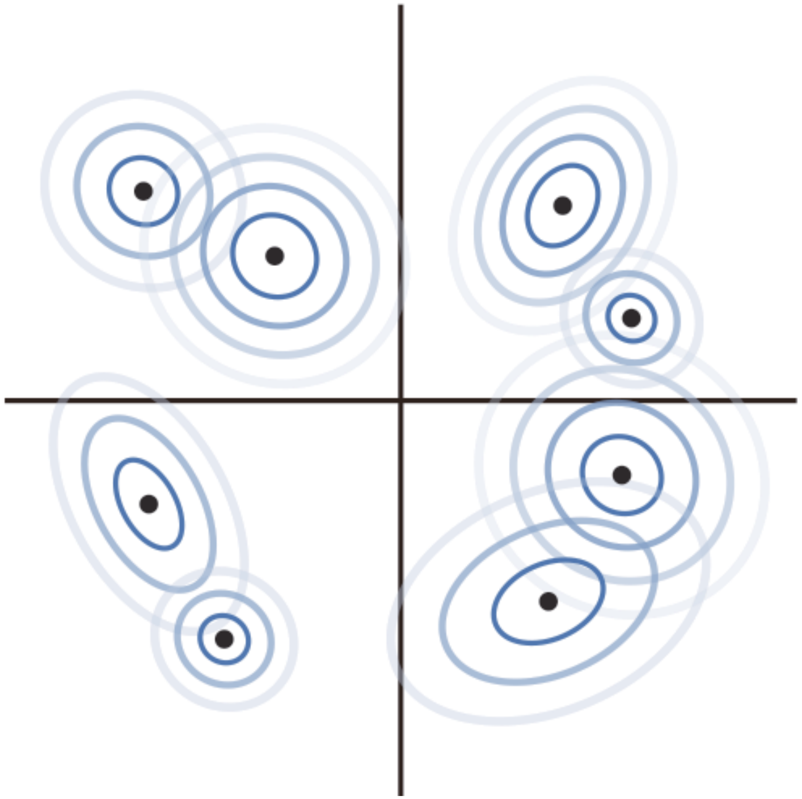
c_{ik} : *responsibility* of component k for x_i



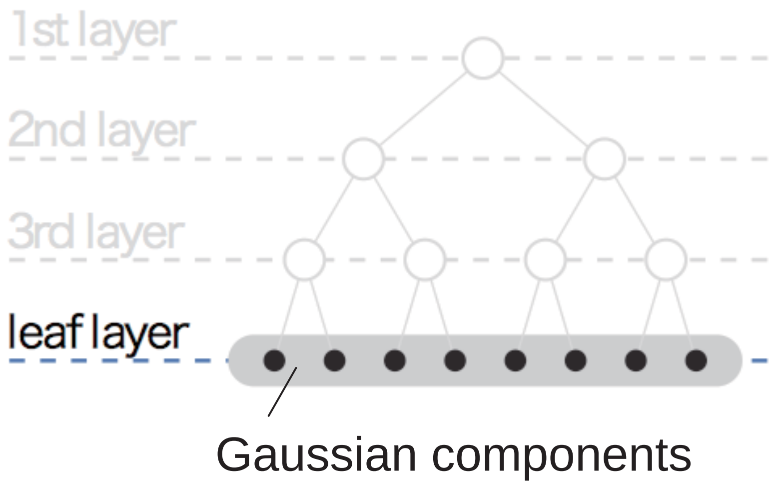
[Nakamasa Inoue, Koichi Shinoda, "A Fast MAP Adaptation Technique for GMM-supervector-based Video Semantic Indexing Systems," In Proc. of ACM Multimedia (short paper), 2011]

Tree-structured GMMs

- Leaf layer



Leaf node has a Gaussian of the UBM (prior GMM).

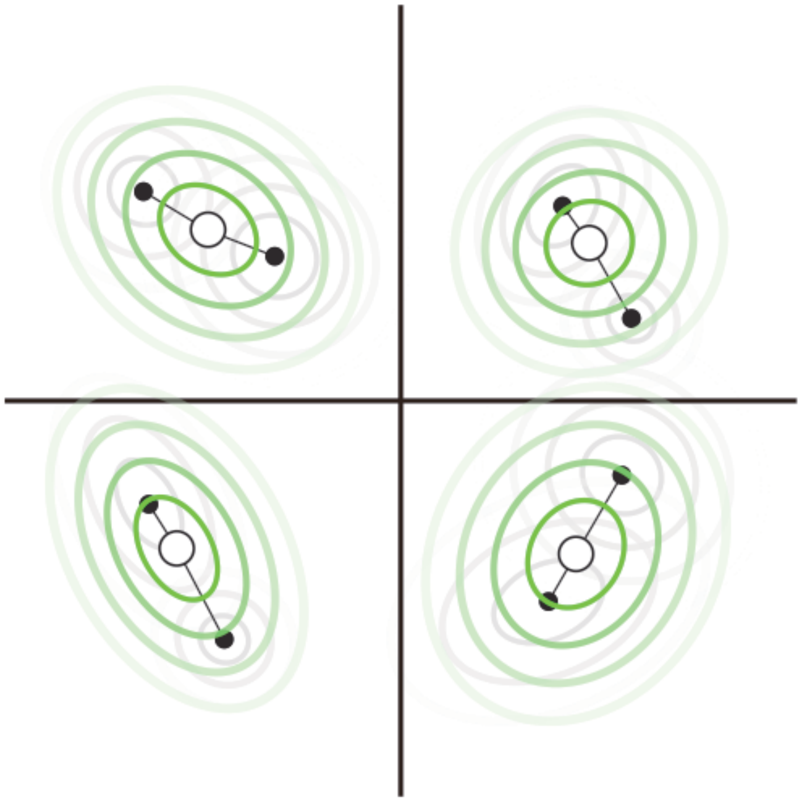


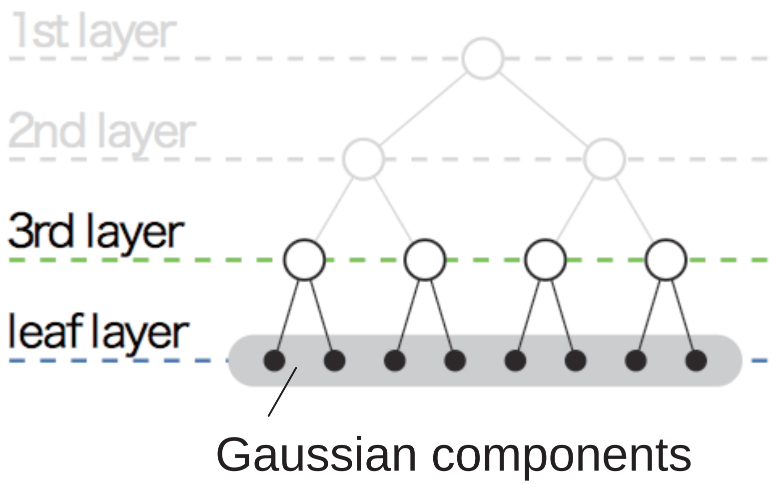
[Nakamasa Inoue, Koichi Shinoda, "A Fast MAP Adaptation Technique for GMM-supervector-based Video Semantic Indexing Systems," In Proc. of ACM Multimedia (short paper), 2011]

Tree-structured GMMs

- Non-leaf layers

Non-leaf node has a Gaussian that approximates its descendant Gauusians



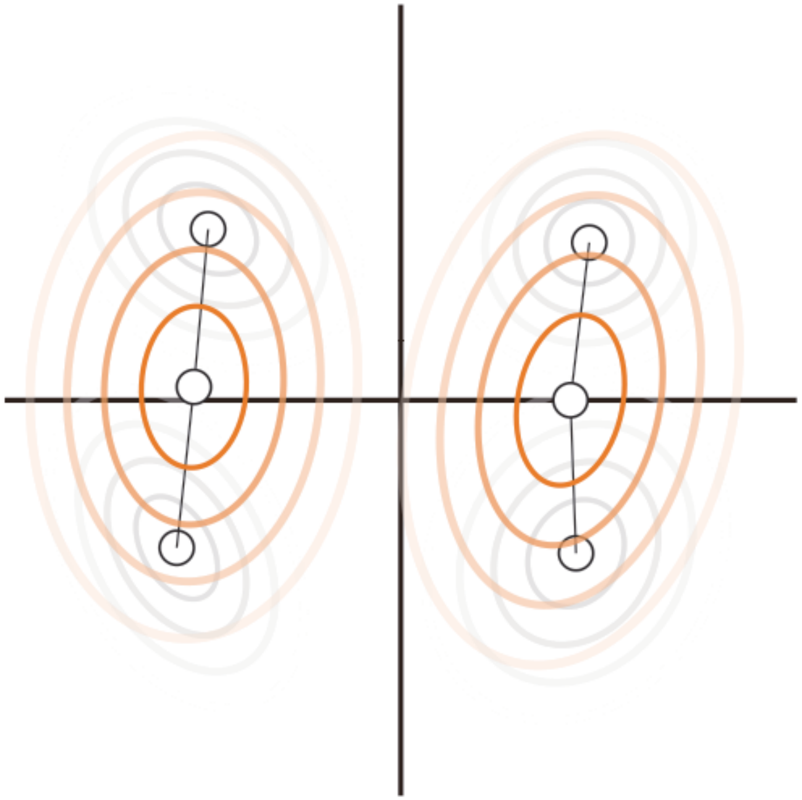


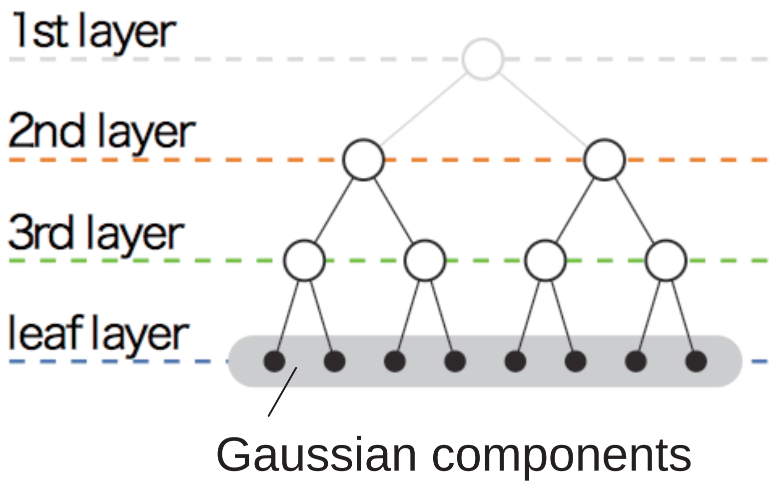
[Nakamasa Inoue, Koichi Shinoda, "A Fast MAP Adaptation Technique for GMM-supervector-based Video Semantic Indexing Systems," In Proc. of ACM Multimedia (short paper), 2011]

Tree-structured GMMs

- Non-leaf layers

Non-leaf node has a Gaussian that approximates its descendant Gauusians

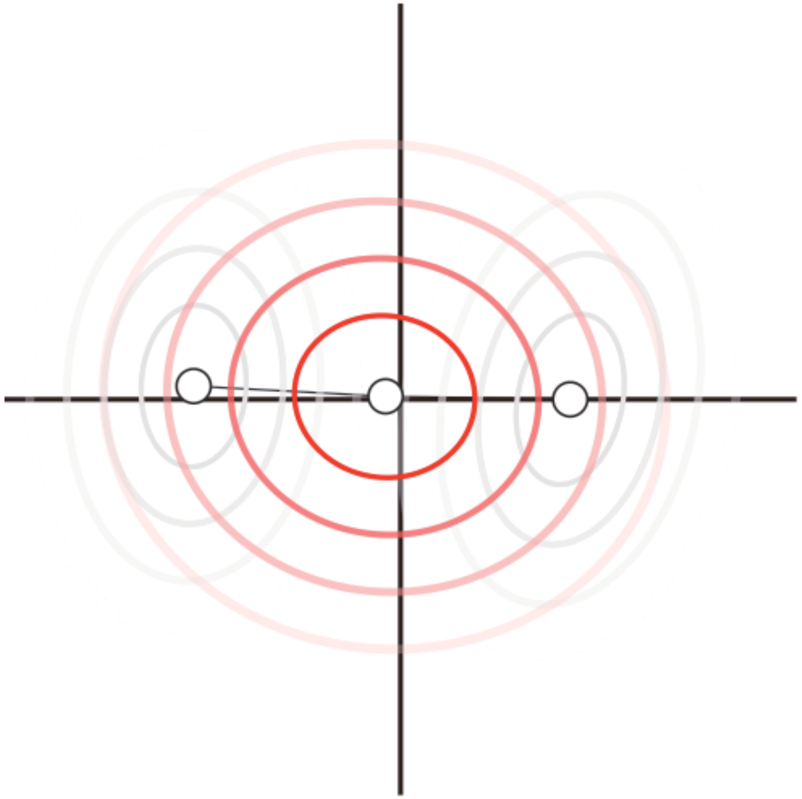




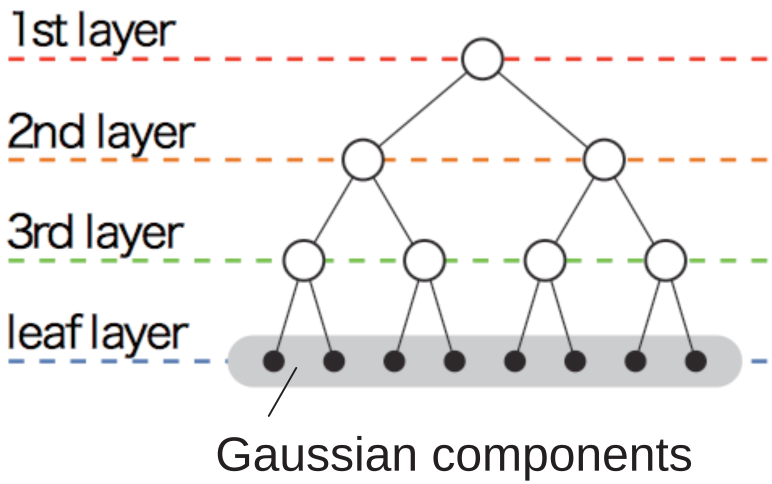
[Nakamasa Inoue, Koichi Shinoda, "A Fast MAP Adaptation Technique for GMM-supervector-based Video Semantic Indexing Systems," In Proc. of ACM Multimedia (short paper), 2011]

Tree-structured GMMs

- Non-leaf layers



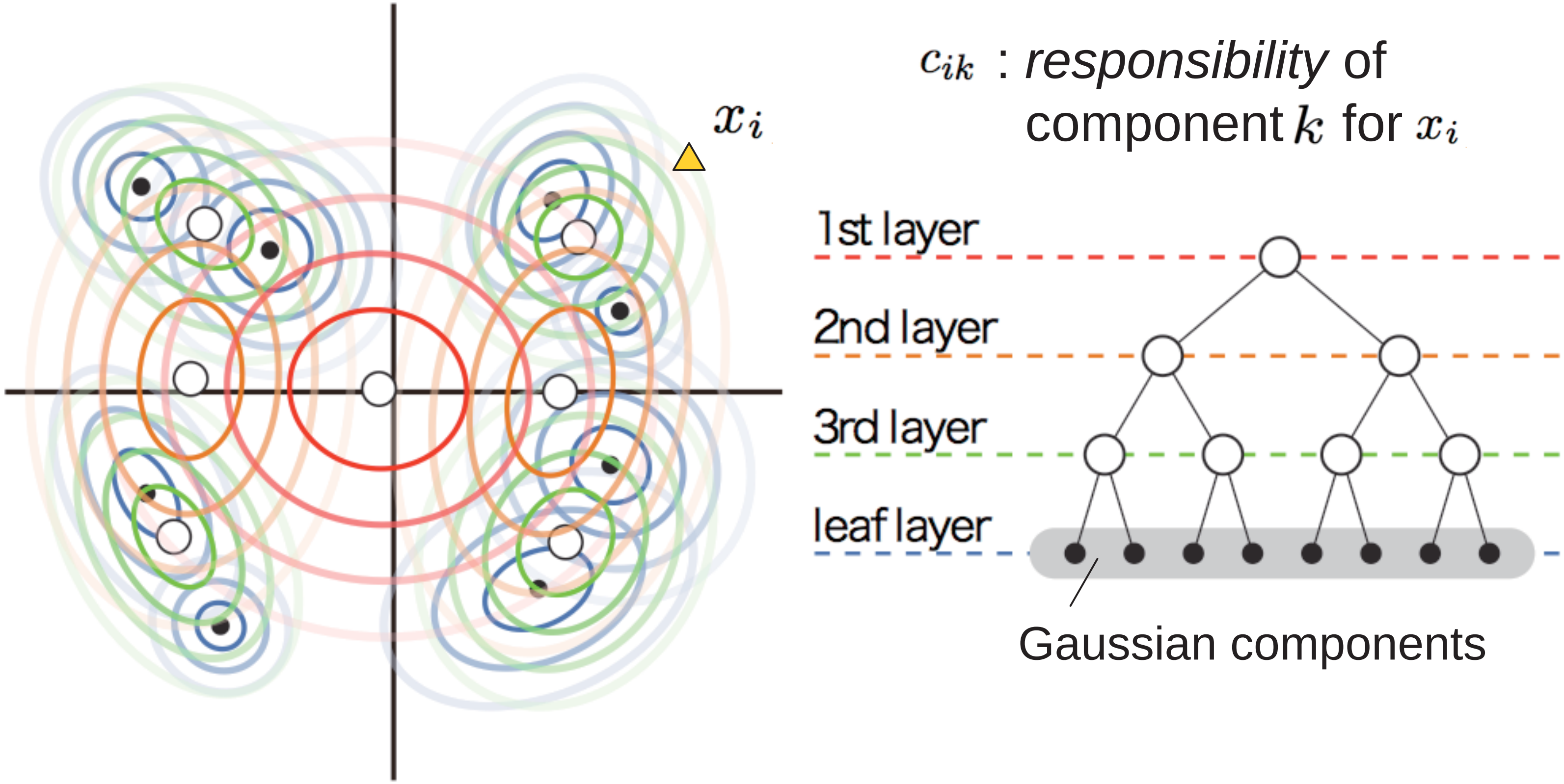
Non-leaf node has a Gaussian that approximates its descendant Gauusians



[Nakamasa Inoue, Koichi Shinoda, “A Fast MAP Adaptation Technique for GMM-supervector-based Video Semantic Indexing Systems,” In Proc. of ACM Multimedia (short paper), 2011]

Tree-structured GMMs

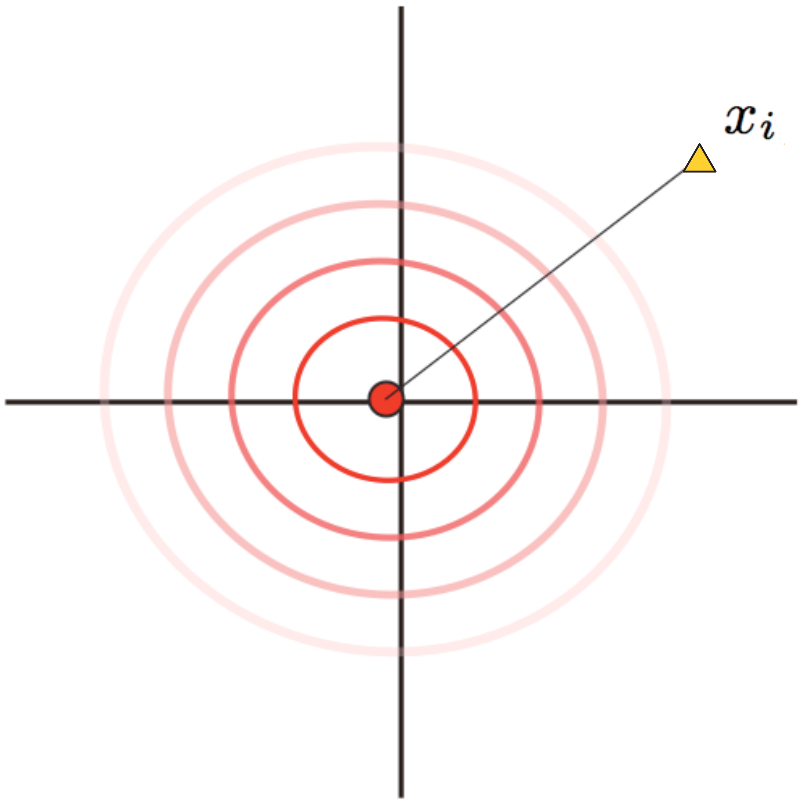
- Calculate responsibilities c_{ik} quickly.



[Nakamasa Inoue, Koichi Shinoda, "A Fast MAP Adaptation Technique for GMM-supervector-based Video Semantic Indexing Systems," In Proc. of ACM Multimedia (short paper), 2011]

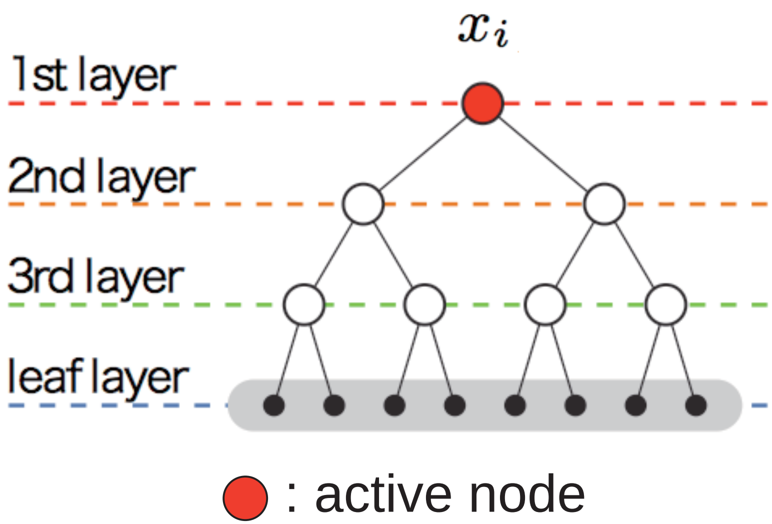
Fast MAP Adaptation

- Calculate responsibilities c_{ik} quickly.



1. Initialize $V_A \leftarrow \{r\}$

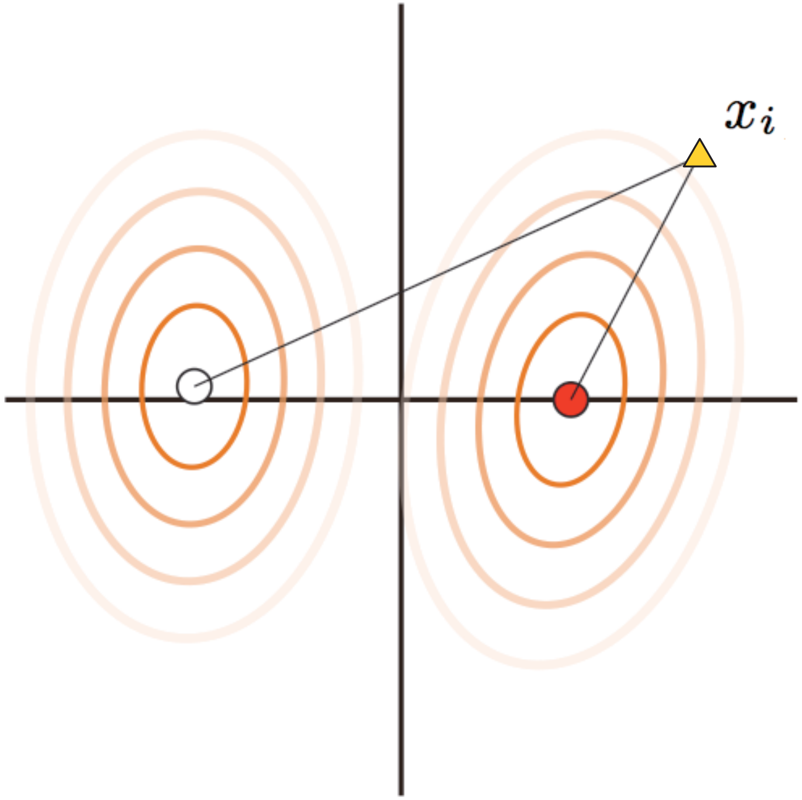
V_A : a set of active nodes



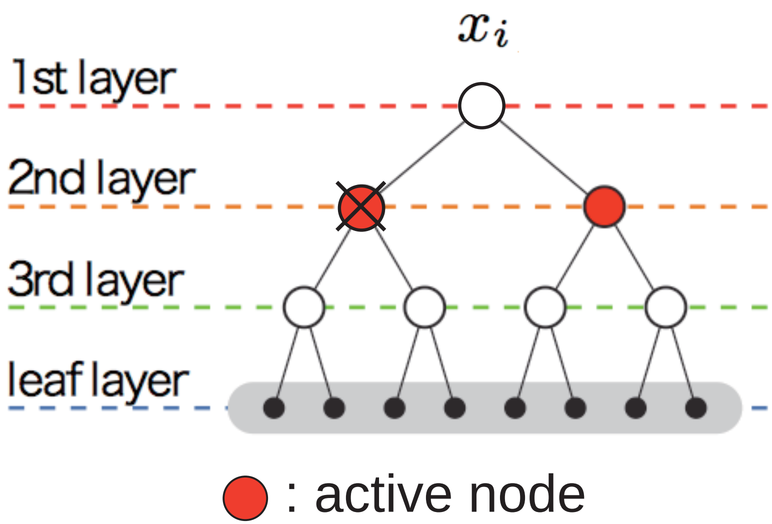
[Nakamasa Inoue, Koichi Shinoda, "A Fast MAP Adaptation Technique for GMM-supervector-based Video Semantic Indexing Systems," In Proc. of ACM Multimedia (short paper), 2011]

Fast MAP Adaptation

- Calculate responsibilities c_{ik} quickly.



2. Make children of V_A active
3. Keep nodes if $c_i^{(v)} > c_{\mathrm{TH}}$

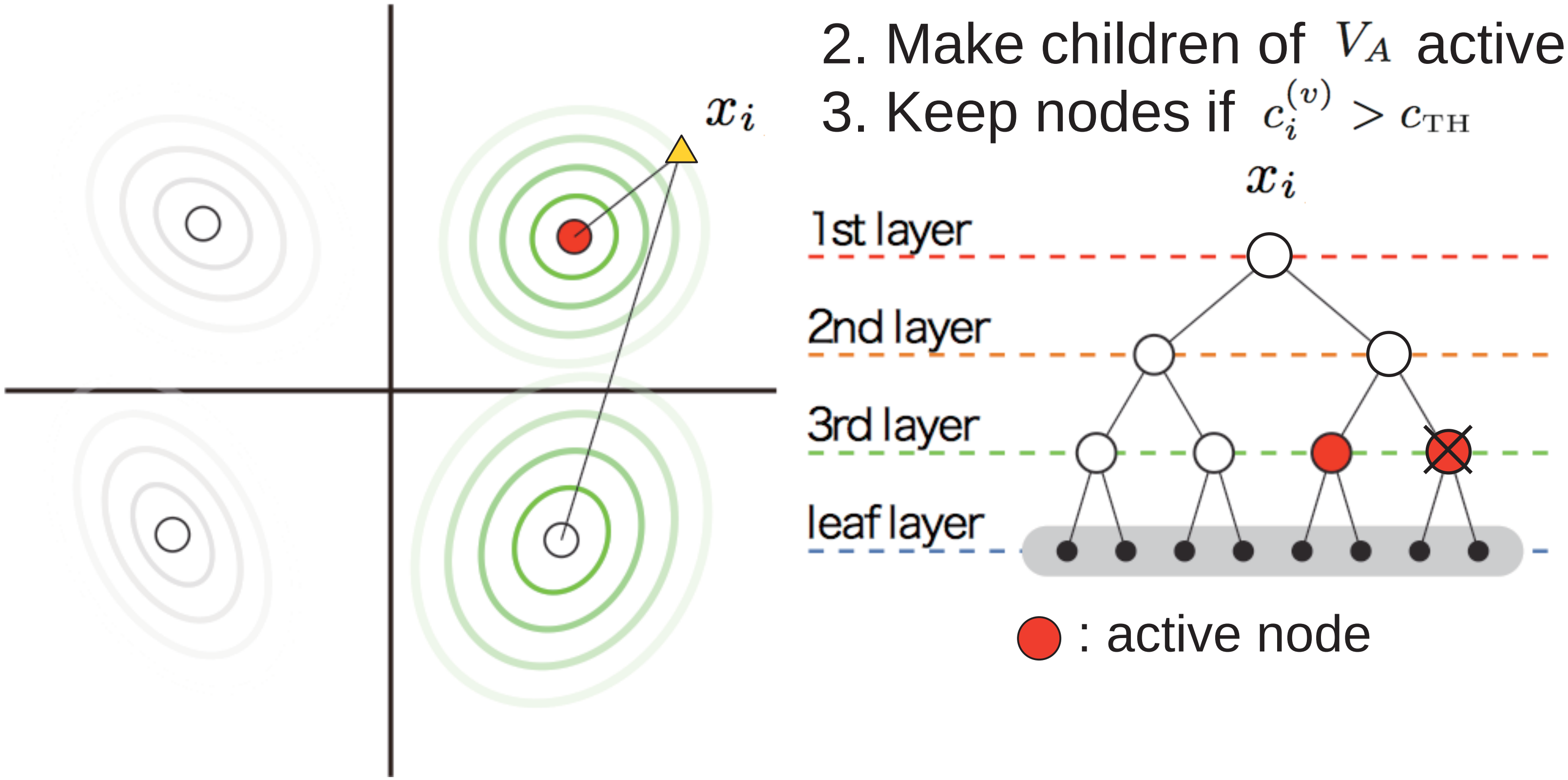


[Nakamasa Inoue, Koichi Shinoda, "A Fast MAP Adaptation Technique for GMM-supervector-based Video Semantic Indexing Systems," In Proc. of ACM Multimedia (short paper), 2011]

Fast MAP Adaptation

- Calculate responsibilities c_{ik} quickly.

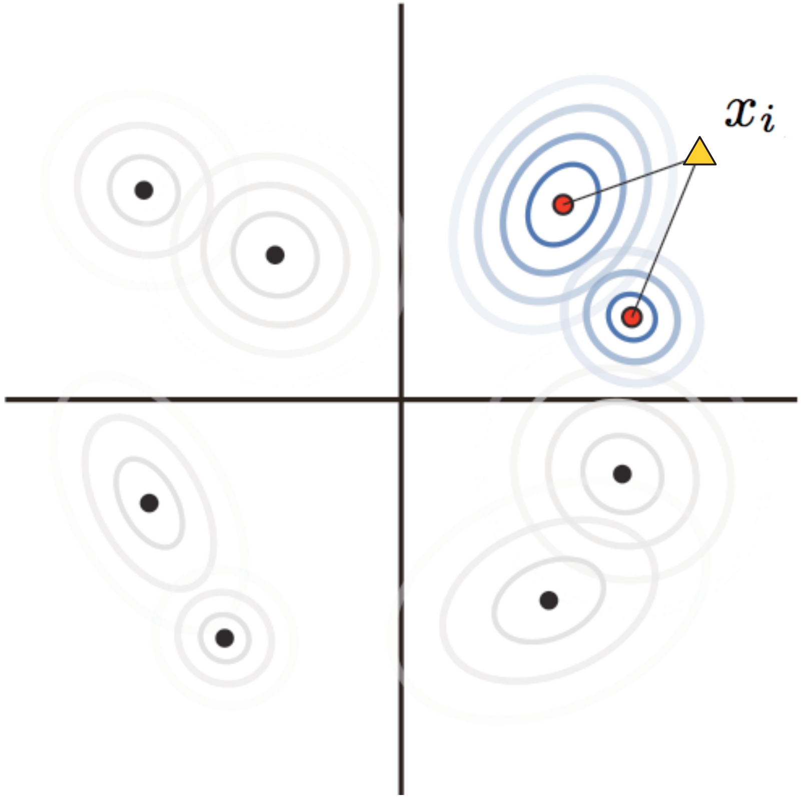
2. Make children of V_A active
3. Keep nodes if $c_i^{(v)} > c_{\mathrm{TH}}$



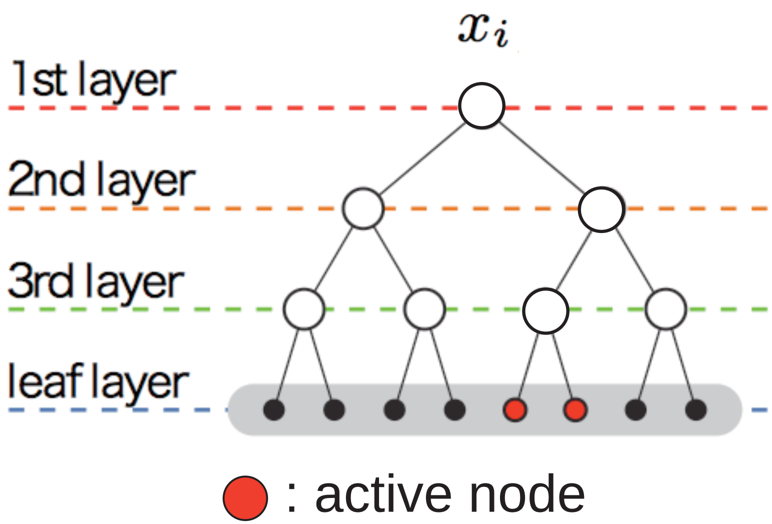
[Nakamasa Inoue, Koichi Shinoda, "A Fast MAP Adaptation Technique for GMM-supervector-based Video Semantic Indexing Systems," In Proc. of ACM Multimedia (short paper), 2011]

Fast MAP Adaptation

- Calculate responsibilities c_{ik} quickly.



2. Make children of V_A active
3. Keep nodes if $c_i^{(v)} > c_{\mathrm{TH}}$



[Nakamasa Inoue, Koichi Shinoda, “A Fast MAP Adaptation Technique for GMM-supervector-based Video Semantic Indexing Systems,” In Proc. of ACM Multimedia (short paper), 2011]

Fast MAP Adaptation

- Summary of the algorithm

V_A : a set of active nodes

1. Initialize $V_A \leftarrow \{r\}$

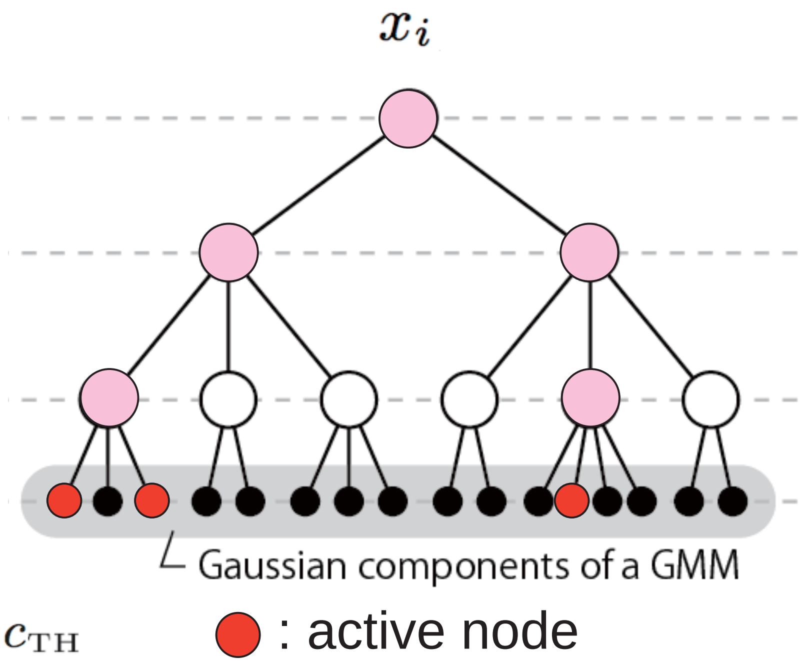
   r : root node

2. Make children of V_A active
3. Calculate

$$c_i^{(v)} = \frac{\tilde{w}^{(v)} g^{(v)}(x_i)}{\sum_{v \in V_A} \tilde{w}^{(v)} g^{(v)}(x_i)}$$

and keep nodes active if $c_i^{(v)} > c_{\mathrm{TH}}$

4. Go to 5 if all nodes in V_A are leafs, otherwise return to 2
5. Output GMM parameters



[Nakamasa Inoue, Koichi Shinoda, "A Fast MAP Adaptation Technique for GMM-supervector-based Video Semantic Indexing Systems," In Proc. of ACM Multimedia (short paper), 2011]

Fast MAP Adaptation

- Summary of the algorithm

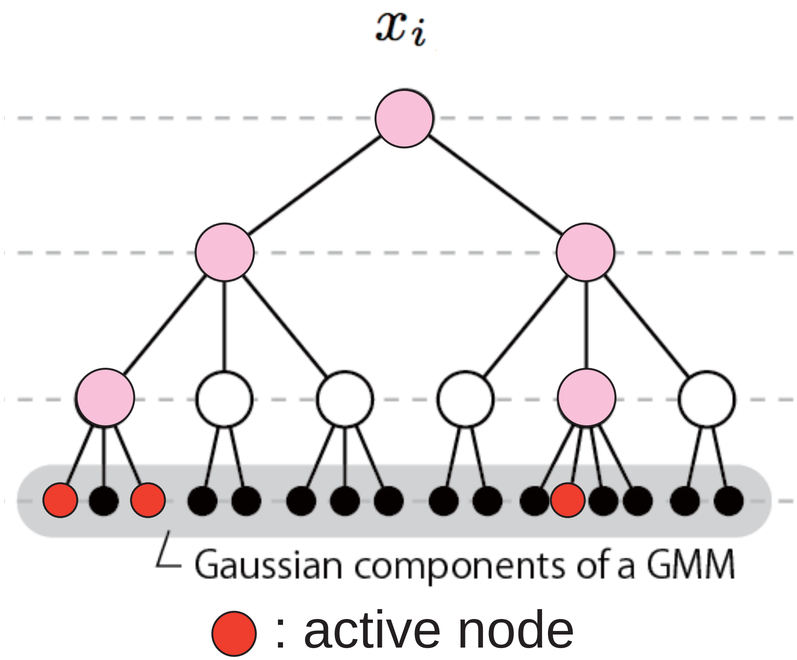
5. Output GMM parameters

$$\hat{\mu}_k = \frac{\tau \hat{\mu}_k^{(\mathrm{U})} + \sum_{\hat{c}_{ik} \neq 0} \hat{c}_{ik} x_i}{\tau + \hat{C}_k}$$

where

$$\hat{c}_{ik} = \begin{cases} c_i^{(\ell)} & (\ell \in V_A,\ g^{(\ell)} = g_k) \\ 0 & (\text{otherwise}) \end{cases}$$

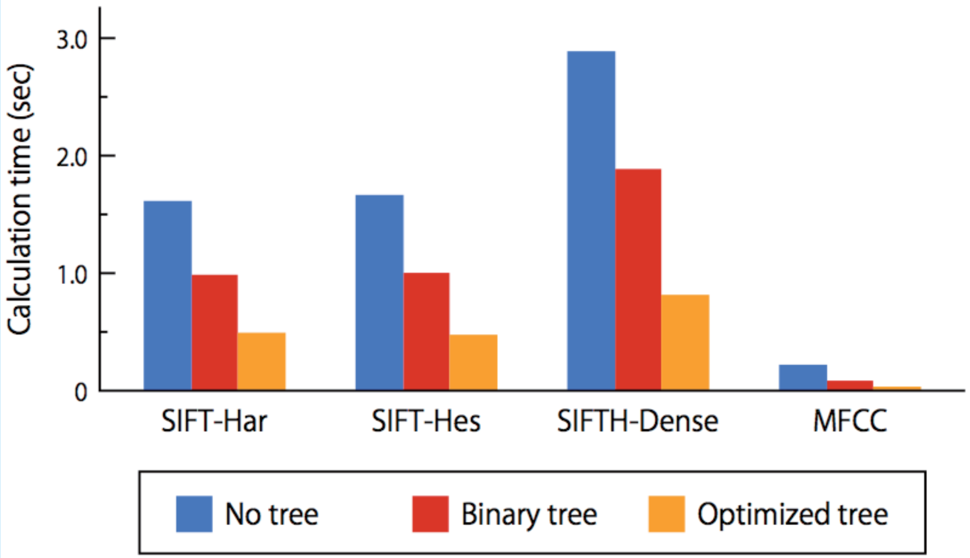
$$\hat{C}_k = \sum_{\hat{c}_{ik} \neq 0} \hat{c}_{ik}$$



[Nakamasa Inoue, Koichi Shinoda, "A Fast MAP Adaptation Technique for GMM-supervector-based Video Semantic Indexing Systems," In Proc. of ACM Multimedia (short paper), 2011]

Fast MAP Adaptation

- Calculation time for MAP adaptation
  - 4.2 times faster than without tree-structured GMMs
  - No decrease in accuracy



Mean InfAP(%) on TRECVID 2010 dataset

Feature	No tree	$\mathcal{T}_{\mathrm{opt}}$
SIFT-Har	6.30	6.32
SIFT-Hes	5.96	6.08
SIFTH-Dense	7.10	6.95
MFCC	1.99	2.00
Fusion	10.15	10.16

Optimized tree: the best tree in terms of calculation time on training data.
Trees of depth at most 5 that have at most 5 children per node are tested.

GMM Supervector

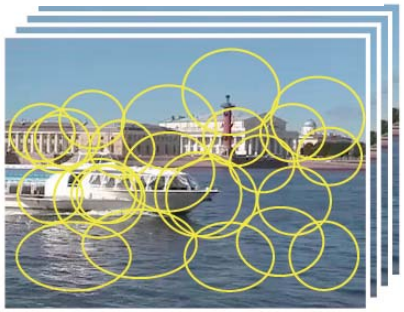
- Combine normalized mean vectors.

$$\phi(X_{\mathrm{F}}) = \begin{pmatrix} \tilde{\mu}_1 \\ \tilde{\mu}_2 \\ \vdots \\ \tilde{\mu}_K \end{pmatrix}$$

where

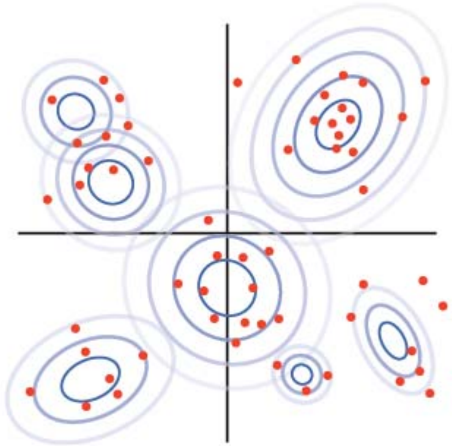
$$\tilde{\mu}_k = \underbrace{\sqrt{w_k^{(U)}}(\Sigma_k^{(U)})^{-\frac{1}{2}}}_{\text{normalized}} \underbrace{\hat{\mu}_k}_{\text{mean}}$$

$X_s = \{x_i\}_{i=1}^{n_s}$



UBM

Fast MAP adaptation

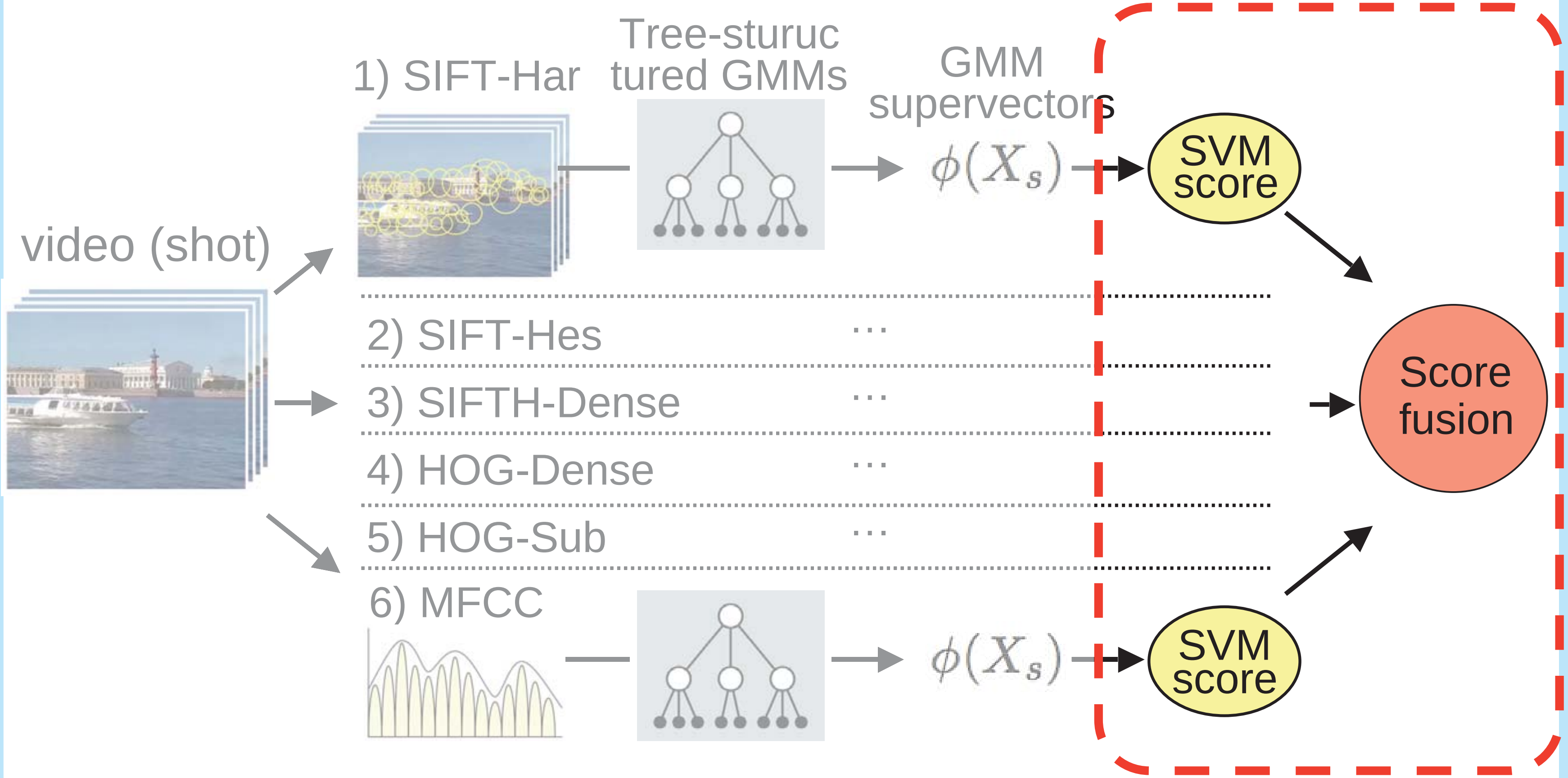


$\phi(X_{\mathrm{F}})$

GMM supervector

System Overview

- Fast and high-performance semantic indexing system



Score Fusion

- SVMs are trained with RBF-kernels

$$k(X_{\mathrm{F}}, X'_{\mathrm{F}}) = \exp\left(-\gamma \|\phi(X_{\mathrm{F}}) - \phi(X'_{\mathrm{F}})\|_2^2\right),$$

- Score fusion

Linear combination of SVM scores:

$$f(X) = \sum_{\mathrm{F}\in\mathcal{F}} \alpha_{\mathrm{F}} f_{\mathrm{F}}(X_{\mathrm{F}}), \quad 0 \le \alpha_{\mathrm{F}} \le 1, \quad \sum_{\mathrm{F}} \alpha_{\mathrm{F}} = 1$$

where $\mathcal{F} = \{\text{SIFT-Har}, \text{SIFT-Hes}, \text{SIFTH-Dense}, \text{HOG-Dense}, \text{HOG-Sub}, \text{MFCC}\}$

Combination coefficients α_{F} are optimized on a validation set (IACC_1_tv10_training for training, and IACC_1_A for validation).

Experimental Condition

- TokyoTech_Canon_1

6 features, 3 parameters for RBF-kernel
(18 SVMs for one semantic concept)

$$f(X) = \sum_{\substack{h \in \{0.5, 1.0, 2.0\} \\ \mathrm{F} \in \mathcal{F}}} \alpha_{\mathrm{F}}^{(h)} f_{\mathrm{F}}^{(h)}(X_{\mathrm{F}})$$

$$\gamma = h\tilde{d}^{-1} \ (h = 0.5, 1.0, 2.0)$$

$$\mathcal{F} = \{\text{SIFT-Har, SIFT-Hes, SIFTH-Dense, HOG-Dense, HOG-Sub, MFCC}\}$$

- TokyoTech_Canon_2

6 features, the parameter *h* is fixed to 1.0
(6 SVMs for one semantic concept)

$$f(X) = \sum_{\mathrm{F} \in \mathcal{F}} \alpha_{\mathrm{F}} f_{\mathrm{F}}(X_{\mathrm{F}})$$

$$\mathcal{F} = \{\text{SIFT-Har, SIFT-Hes, SIFTH-Dense, HOG-Dense, HOG-Sub, MFCC}\}$$

Experimental Condition

- TokyoTech_Canon_3

Scores for all semantic concepts are combined:
(i.e. 6 * 346 SVMs for one semantic concept)

$$g_{\mathrm{S}}(X) = \sum_{\mathrm{S'}} \beta_{\mathrm{S'}} f_{\mathrm{S'}}(X), \quad \sum_{\mathrm{S'}} \beta_{\mathrm{S'}} = 1$$

$f_{\mathrm{S'}}$: score for concept *S'*

- TokyoTech_Canon_4

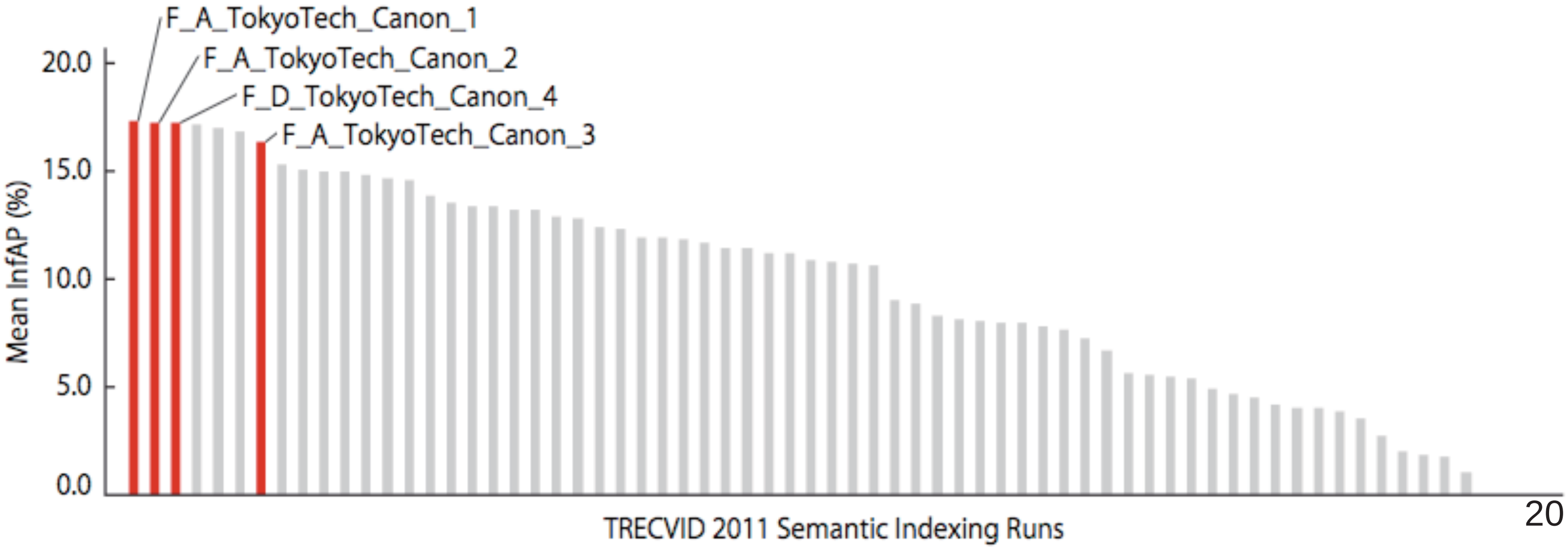
Additional training data from ImageNET
(i.e. 6+1 SVMs for one semantic concept)

$$f(X) = \sum_{\mathrm{F} \in \mathcal{F} \cup \{\text{HOG-Image}\}} \alpha_{\mathrm{F}} f_{\mathrm{F}}(X_{\mathrm{F}})$$

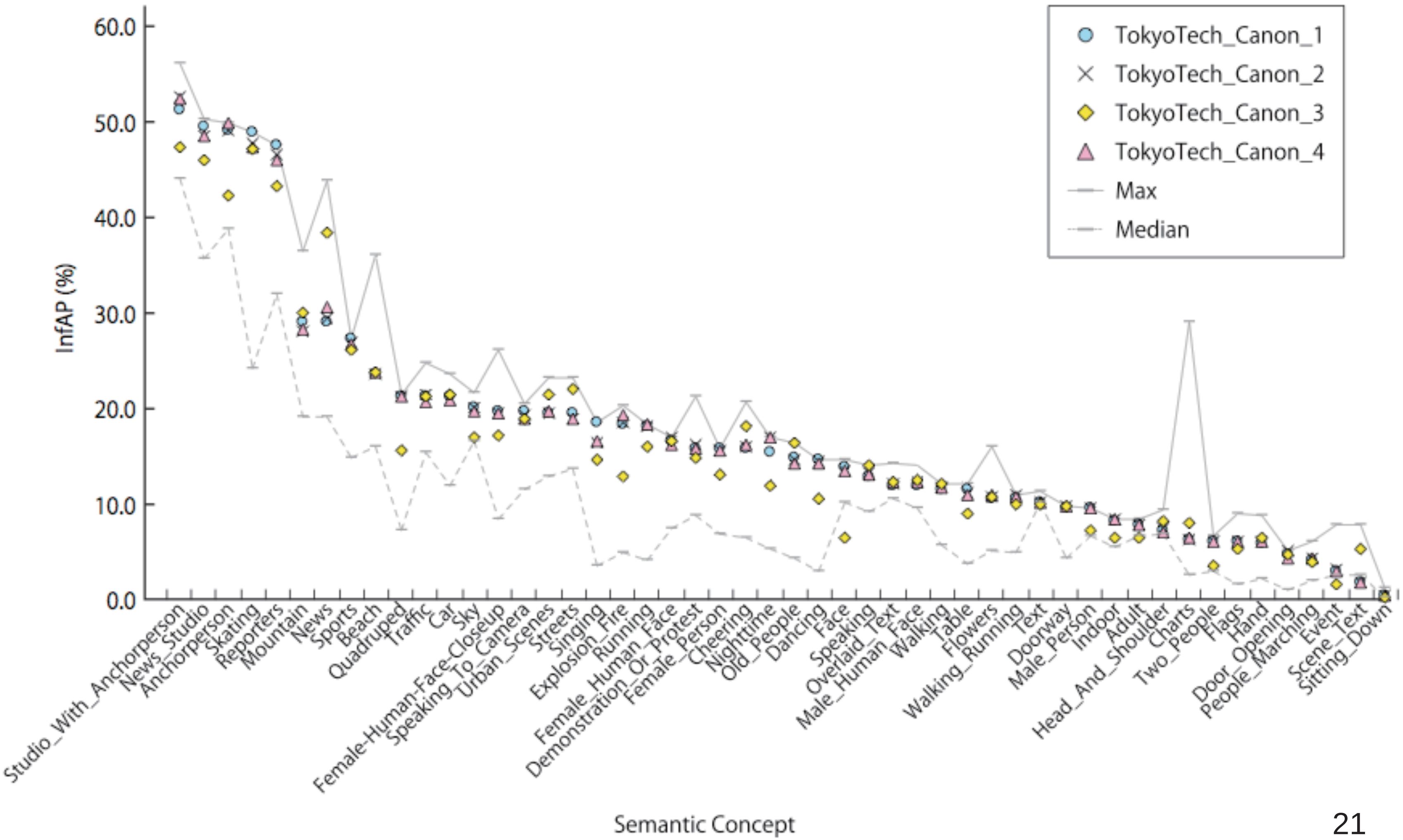
$f_{\text{HOG-Image}}$ is trained on the TRECVID+ImageNET dataset with HOG-Dense features

Results

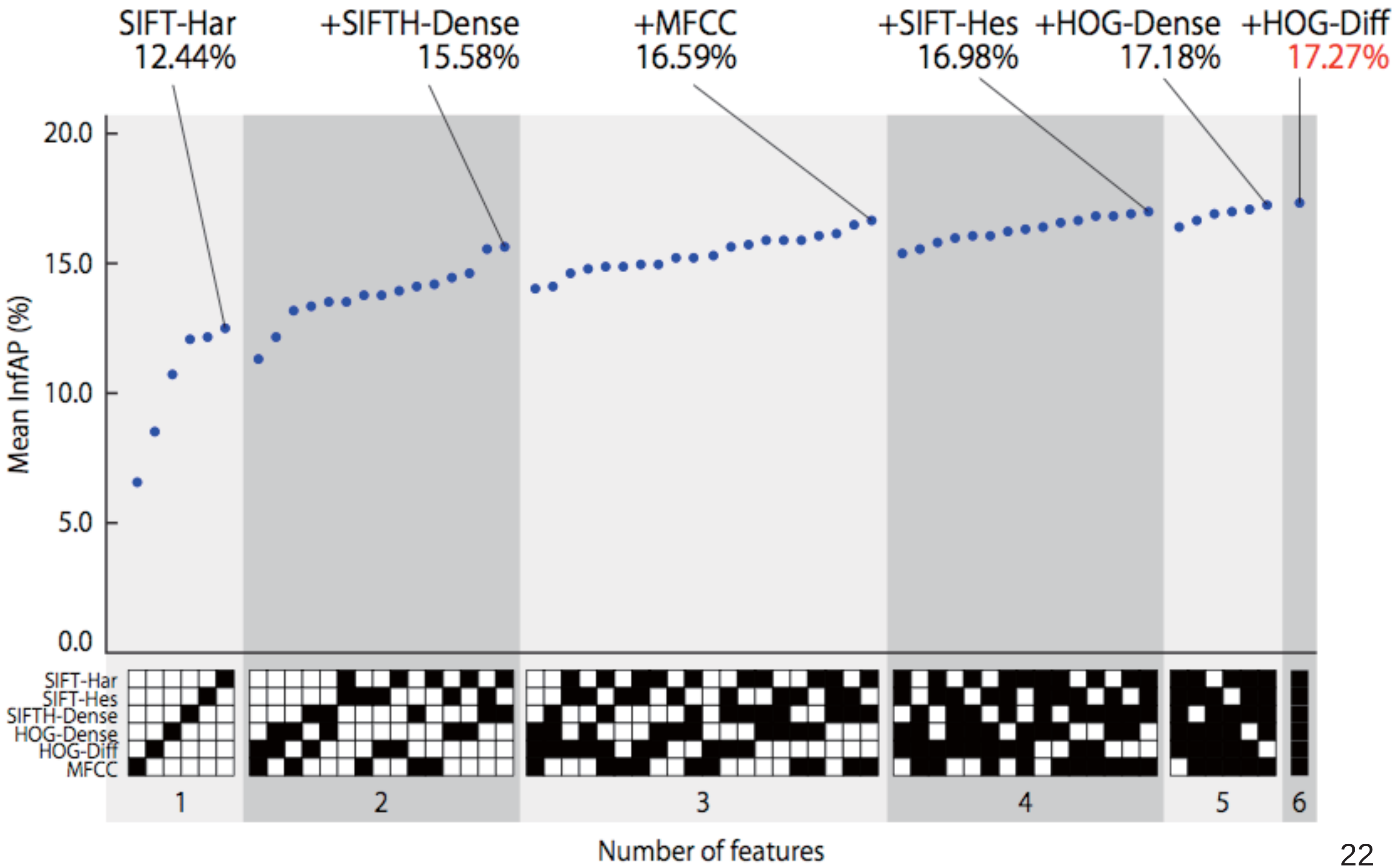
Run ID	Method	Mean InfAP
TokyoTech_Canon_1	6 audio/visual GMM supervectors, 3 parameters for RBF-kernels	17.3%
TokyoTech_Canon_2	fixed parameter for RBF-kernels	17.3%
TokyoTech_Canon_3	2nd run + concept score fusion	16.4%
TokyoTech_Canon_4	2nd run + ImageNET images(Type D)	17.2%



InfAP by Semantic Concepts



Which features are important?



Conclusion

- 6 types of audio and visual GMM supervectors
  Mean InfAP: 17.3%
  - Single feature: 12.4% (SIFT-Har (multi-frame))
  - 3 features: 16.6% (SIFT-Har, SIFTH-Dense, MFCC)
  - No audio:16.4% (5 visual features)

- Fast MAP adaptation
  Tree-structured GMMs cut MAP adaptation costs.
  4.2 times faster than without tree-structured GMMs.

- Future work
  Human actions and event detection.
  Spatial and temporal localization.

Thank you!