



浙江大学计算机学院
数字媒体与网络技术

Digital Asset Management

数字媒体资源管理

6. Introduction to digital media retrieval

任课老师：张宏鑫
2008-10-08

Origin of digital media retrieval

- IR (Information retrieval)
 - To retrieve information that users want based on some keys or hints



Main methods of digital media retrieval

- **Text-based** digital media retrieval

- Boolean model
- Clustering model
- Vector model
- Probability model

Google™
谷歌

YAHOO!®

Bai 百度

- **Content-based** digital media retrieval

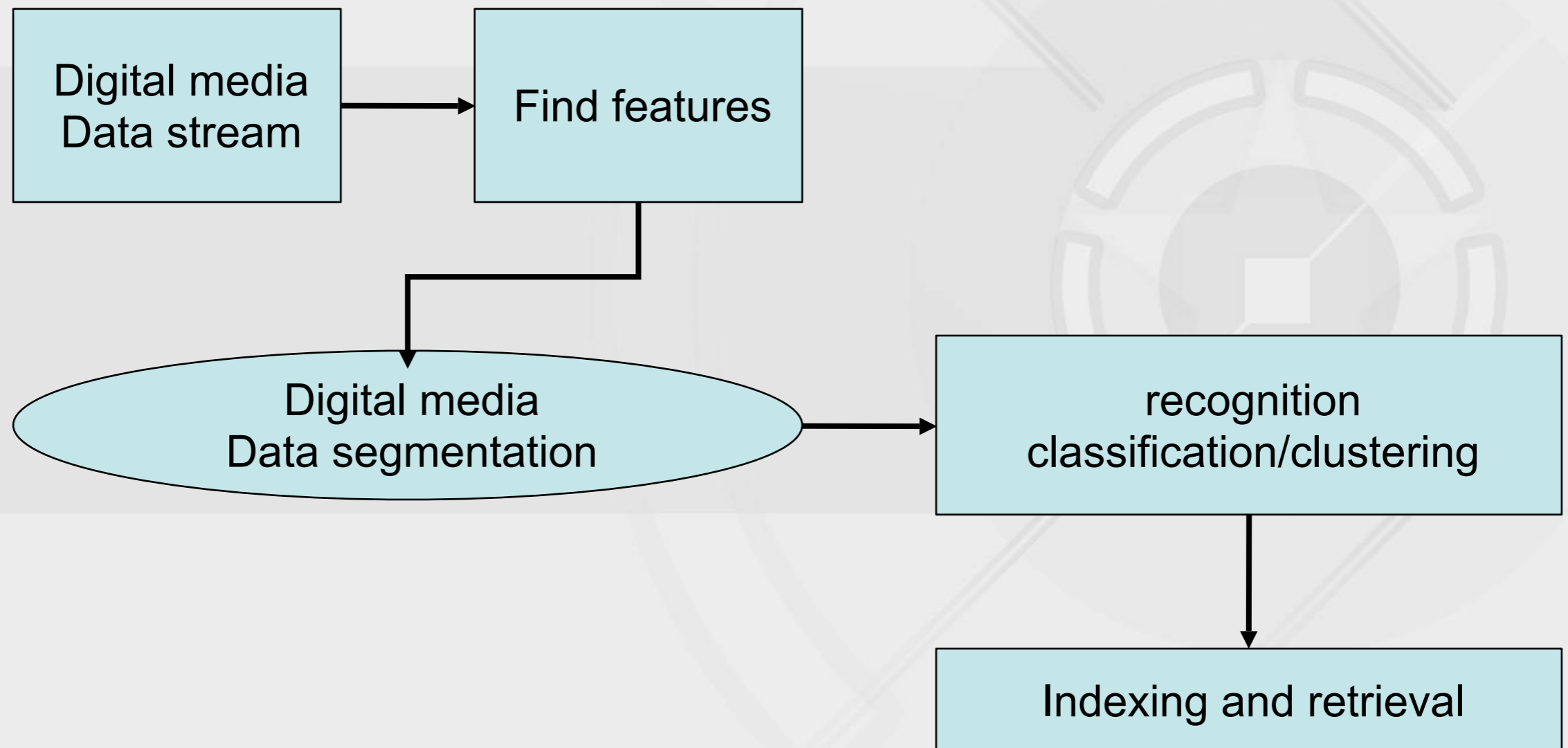


Content based digital media retrieval

- Query by example on multimedia-data
- Demo:
 - The **GNU Image-Finding Tool**
 - <http://www.gnu.org/software/gif/>



The workflow of digital media analysis and retrieval



Content-based digital media retrieval

- In this lesson, we will know ...
 - Content-based image retrieval
 - Content-based video retrieval
 - Content-based audio retrieval
 - Content-based graphics retrieval

- Merging and analysis of multiple media
- Development and challenging





1. Content-based image retrieval





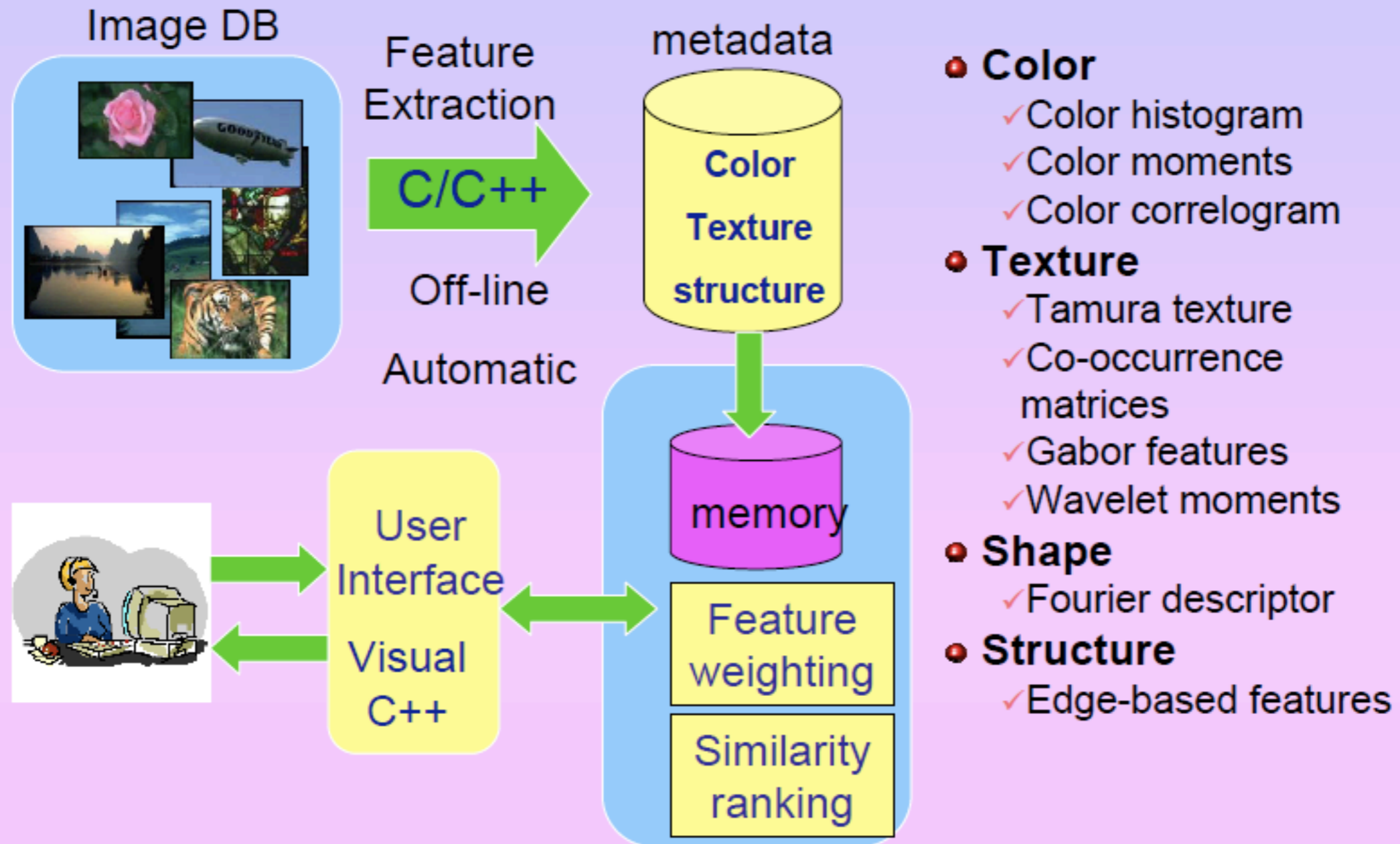
1. Content-based image retrieval



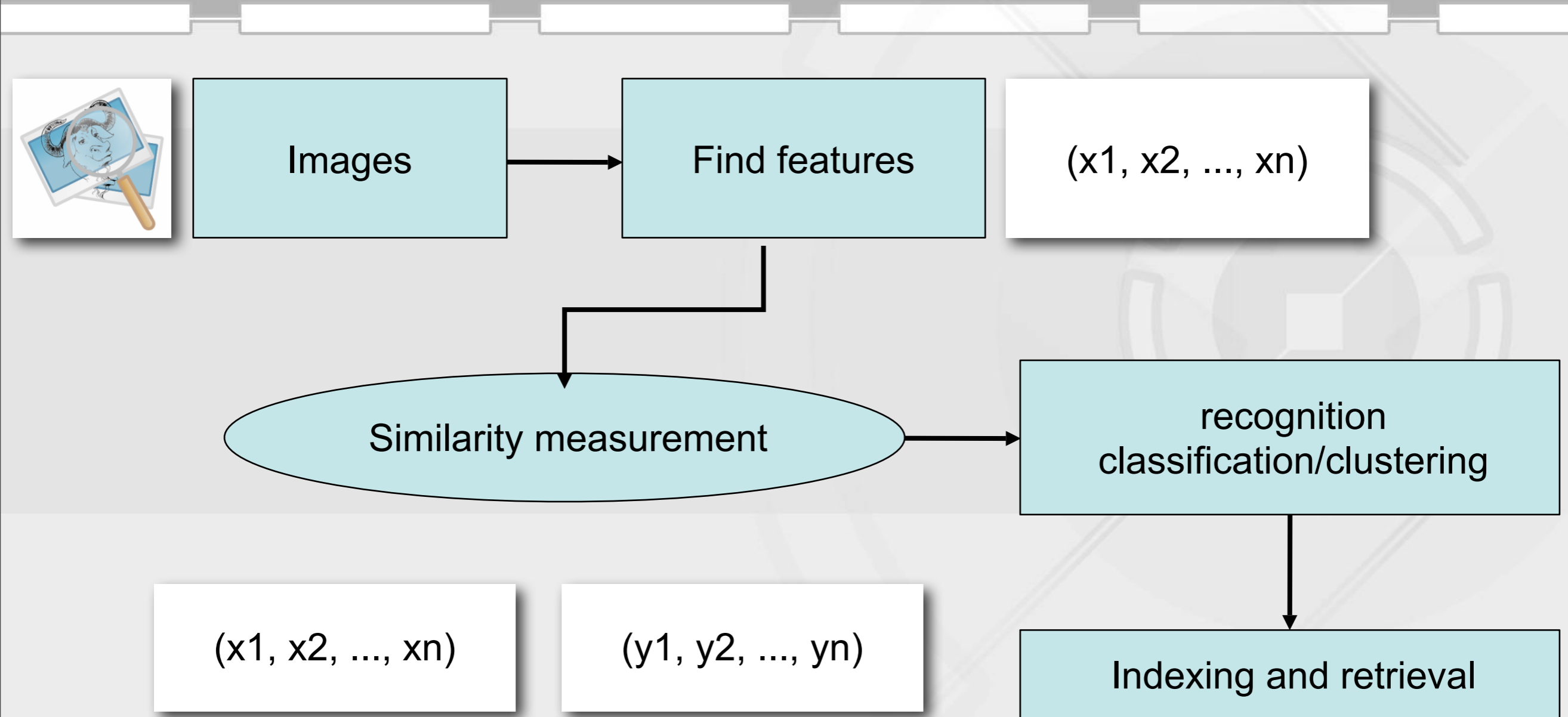
CBIR

Multimedia Information Retrieval

• Content-based Image Retrieval



Workflow of CBIR



Features of image

- Finding out features of image is a key step of image retrieval
 - Image-based retrieval usually need to pre-construct feature database of images for retrieval
- Major image features:
 - Color features
 - Texture features
 - Shape features
 - Space relation features



Color features of image

- Color feature is a most widely used vision feature. It is mainly used to analyze color distributions in an image, including:
 - Color histogram
 - Color moments
 - Color set
 - Color clustering vectors
 - Color relation graph

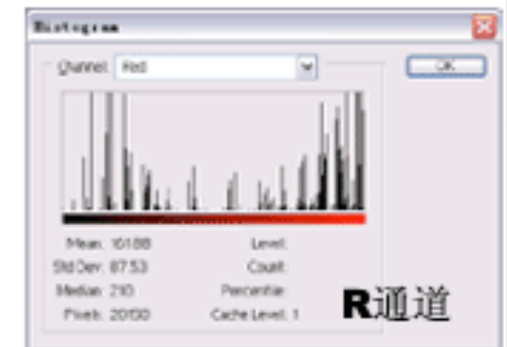
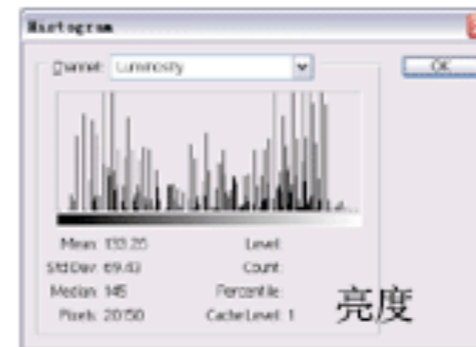
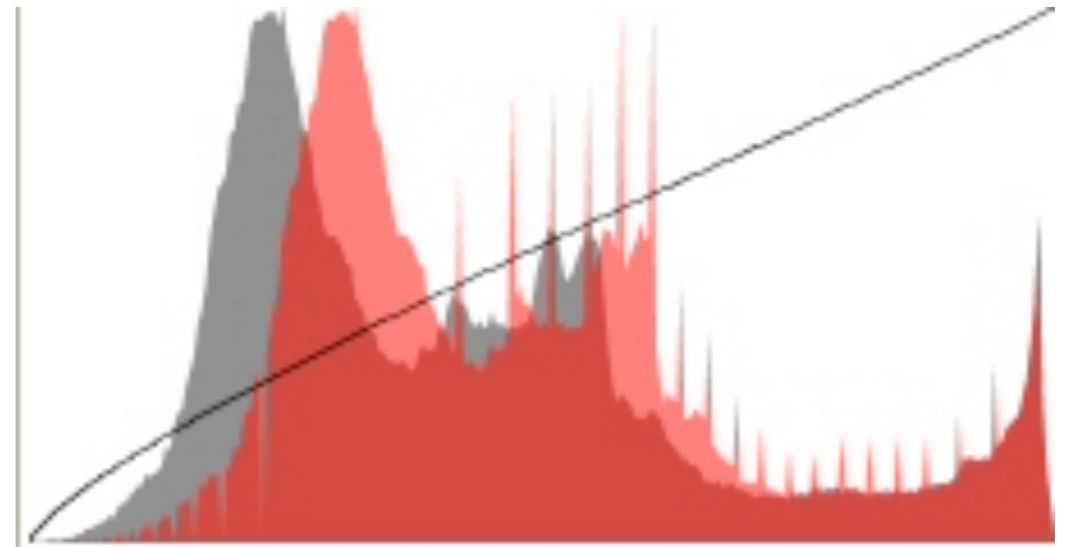
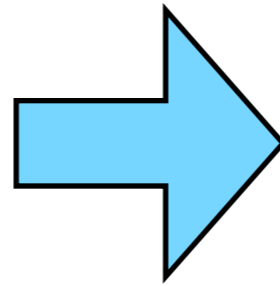


Image histogram



图像的颜色矩 (color moments)

- Color moments are global statistical features of an image, which are proposed by Stricker and Orengo.

- First order moment (mean)

$$\mu_i = \frac{1}{n} \sum_{j=1}^n I_{ij}$$

- Second order moment (variance)

$$\sigma_i^2 = \frac{1}{n} \sum_{j=1}^n (I_{ij} - \mu_i)^2$$

- Third order moment (skewness)

$$s_i^3 = \frac{1}{n} \sum_{j=1}^n (I_{ij} - \mu_i)^3$$

- Color moments are always applied with other image features for efficiently shrinking seeking ranges.



color moments: example

1	3	6	3	1
3	6	8	6	3
6	8	10	8	6
3	6	8	6	3
1	3	6	3	1

mean =4.72

variance =6.52

skewness =2.34

Image texture features

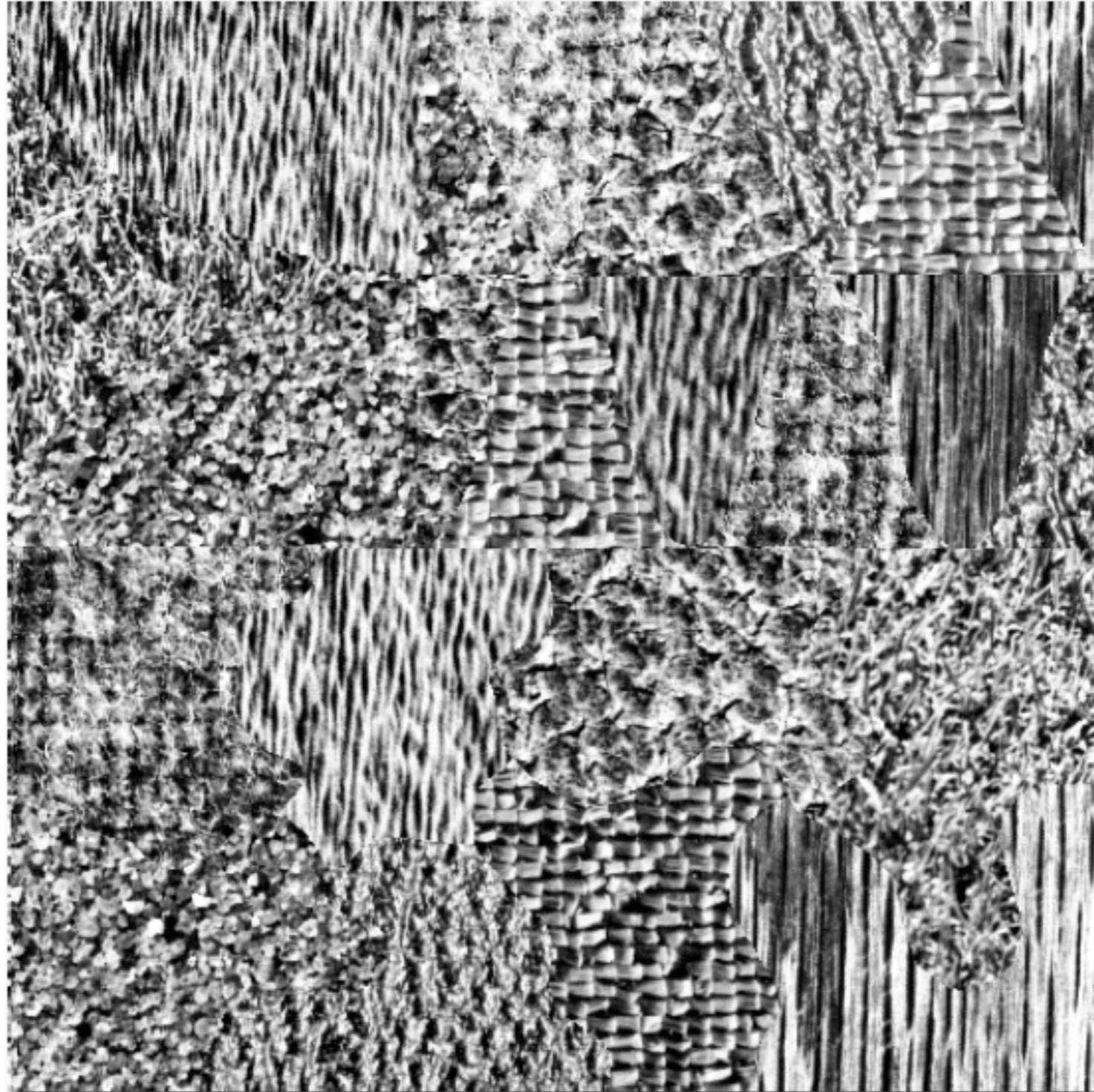


Image texture features

- Texture features are such vision features employed to measure homogeneous phenomenon in images. They are
 - independent to color or illuminance,
 - and are intrinsic features of object surfaces.
- Major texture features
 - Tamura texture features
 - Self-regression texture model
 - Transform based texture features
 - DWT, DFT, Garbor filter bank
 - others

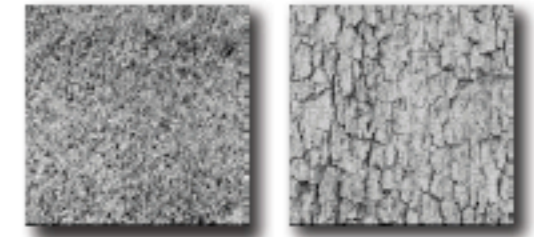


Tamura texture features

- a set of texture feature representation based on the psychology research results on human vision cognition of textures:
 - coarseness (粗糙度)
 - contrast (对比度)
 - directionality (方向度)
 - linelikeness (线像度)
 - regularity (规整度)
 - roughness (粗略度)



Tamura – Coarseness

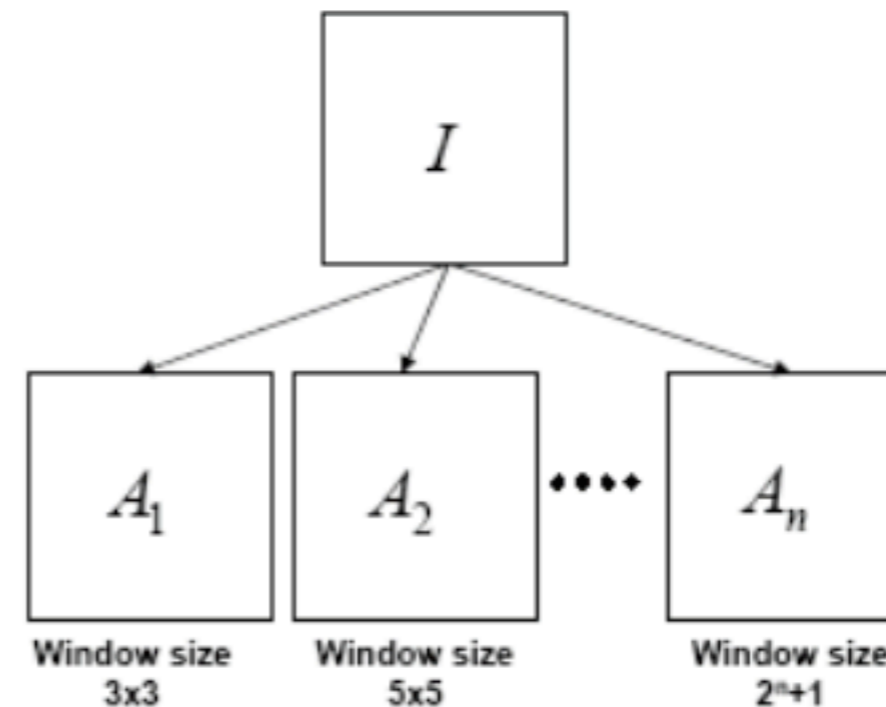
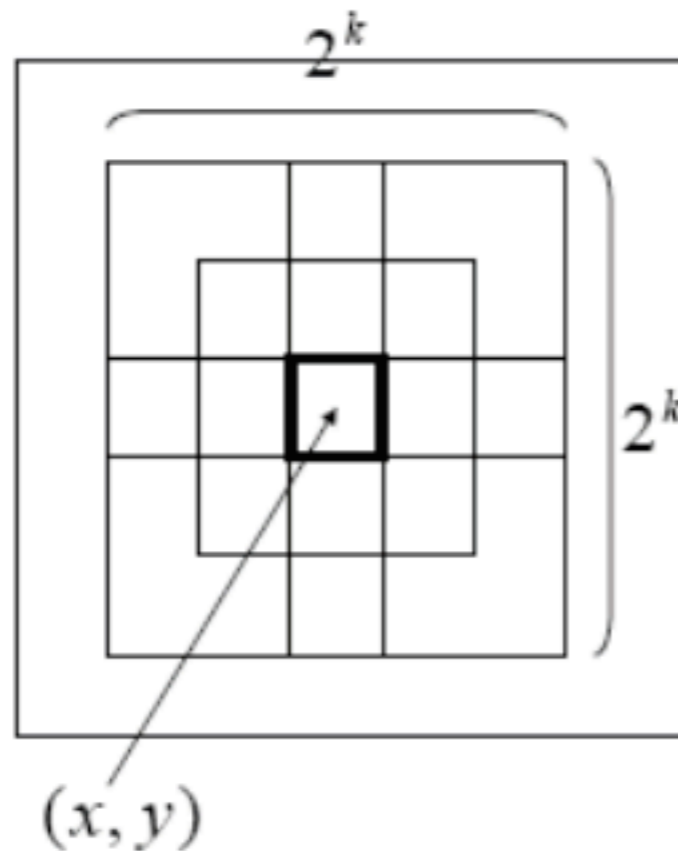


- Goal

- Pick a large size as best when coarse texture is present, or a small size when only fine texture

- Step 1: Compute averages at different scales at every points

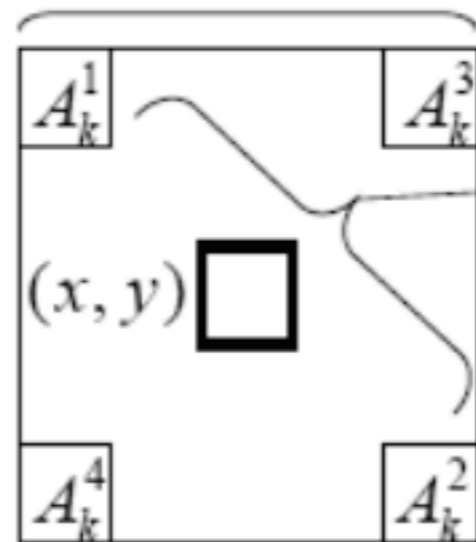
$$A_k(x, y) = \frac{1}{2^{2k}} \sum_{i=x-2^{k-1}}^{x+2^{k-1}-1} \sum_{j=y-2^{k-1}}^{y+2^{k-1}-1} I(i, j)$$



Tamura – Coarseness (cont.)

- Step 2: compute neighborhood difference at each scale on opposite sides of different directions

$$E_{k,h}(x,y) = \frac{|A_k(x - 2^{k-1}, y) - A_k(x + 2^{k-1}, y)|}{2^k}$$



$$E_{k,a}(x,y) = |A_k^1 - A_k^2|$$

$$E_{k,b}(x,y) = |A_k^3 - A_k^4|$$

$$(x,y) \rightarrow \{E_{1,a}, E_{1,b}, E_{2,a}, E_{2,b}, \dots, E_{n,a}, E_{n,b}\}$$

Tamura – Coarseness (cont.)

- Step 3: select the scale with the largest variation

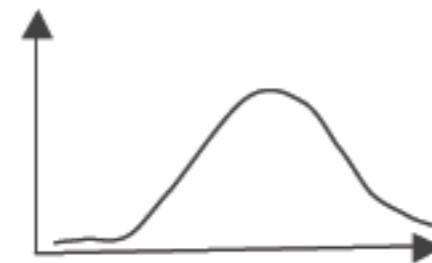
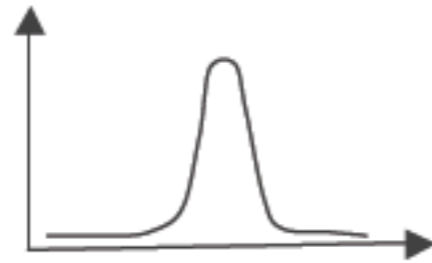
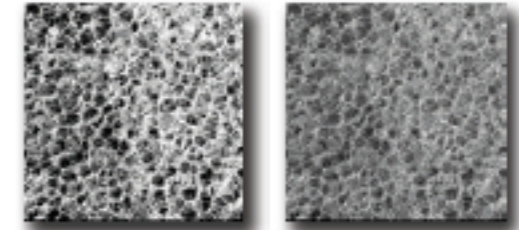
$$S_{max}(x, y) = 2^k \quad / \quad E_k = \max\{E_1, E_2, \dots, E_L\}$$

- Step 4: compute the coarseness

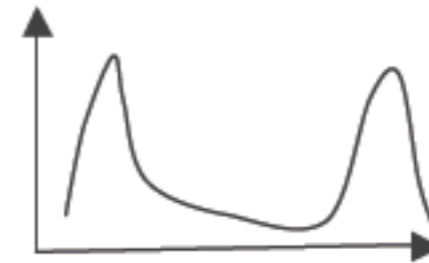
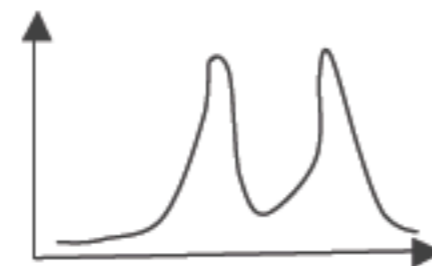
$$M_{\text{crs}} = \frac{1}{n \times m} \sum_i^n \sum_j^m S_{max}(i, j)$$

Tamura – Contrast

- Gaussian-like histogram distribution → low contrast



- Histogram polarization. Is it Gaussian? How many peaks it has? Where they are?



- Polarization can be estimated by the kurtosis (曲率度)

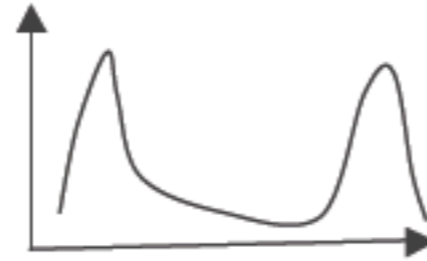
$$\alpha_4 = \frac{\mu_4}{\sigma^4}$$

$$\mu_4 = E[I^4(x, y)]$$

$$\sigma^4 = E[(I(x, y) - \mu)^4]$$

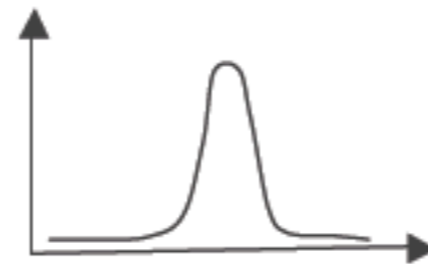
Tamura – Contrast (cont.)

$$\alpha_4 = \frac{\mu_4}{\sigma^4}$$



**distribution with
two separate peaks**

$$\alpha_4 = \frac{\mu_4}{\sigma^4}$$

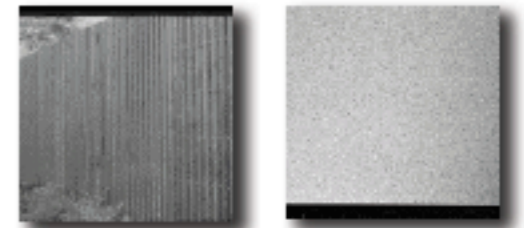


unimodal distribution

- Contrast estimate is given by:

$$M_{contrast} = \frac{\sigma}{(\alpha_4)^{\frac{1}{4}}}$$

Tamura – Orientation



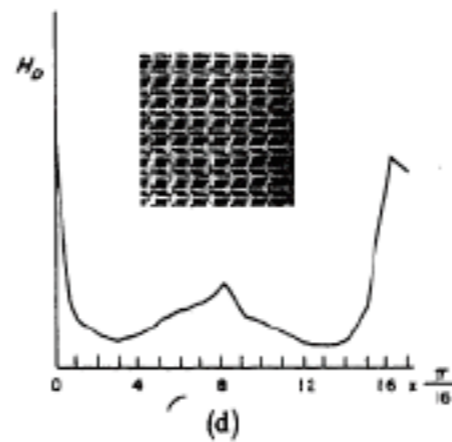
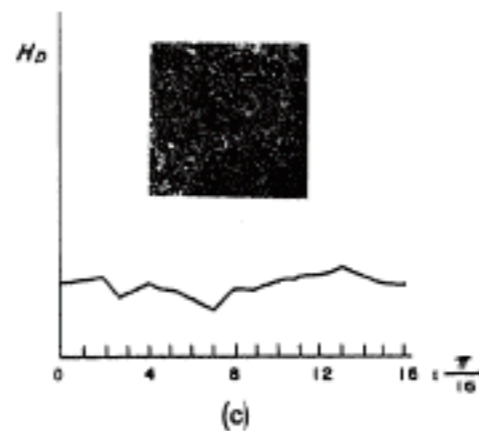
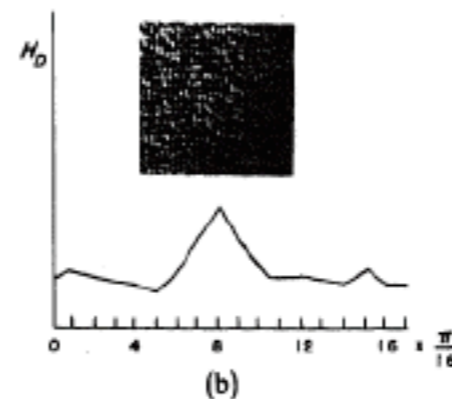
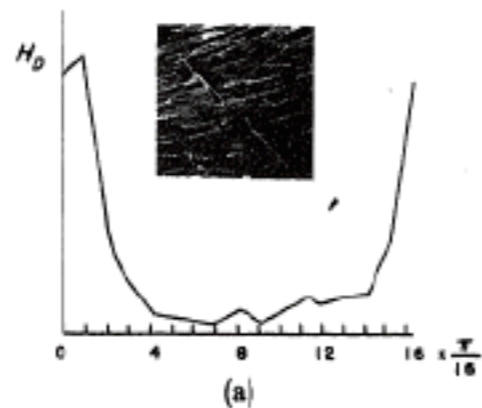
- Building the histogram of local edges at different orientations $H_D(k)$

- By deriving the edge magnitude at X and Y directions

$$\theta = \text{tg}^{-1}(\nabla_V / \nabla_H) + \frac{\pi}{2}$$

$$|\nabla G| = (|\nabla_V| + |\nabla_H|) / 2$$

$$\begin{matrix} \nabla_V & \nabla_H \\ \begin{pmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{pmatrix} & \begin{pmatrix} 1 & 1 & 1 \\ 0 & 0 & 0 \\ -1 & -1 & -1 \end{pmatrix} \end{matrix}$$



Tamura – Orientation (cont.)

- Compute the estimate from the sharpness of the peaks
 - By summing the second moments around each peak
 - e.g., flat histogram
 - large 2nd moment (variance)
 - small orientation

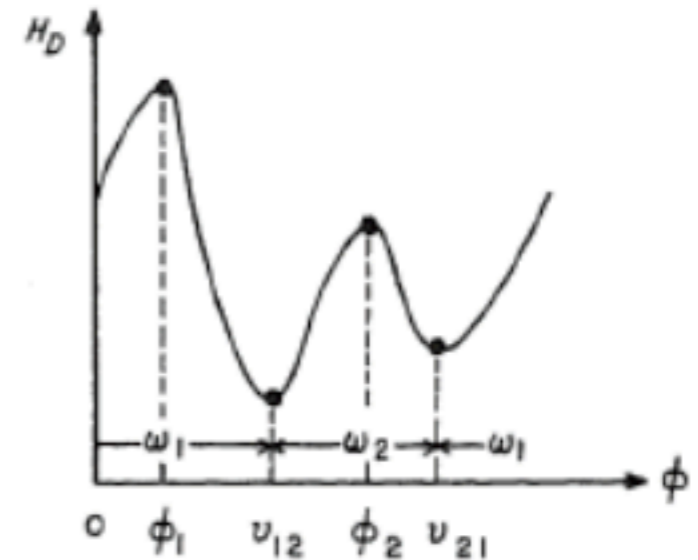
$$M_{orient} = 1 - r \cdot n_p \cdot \sum_p \sum_{\phi \in w_p} (\phi - \phi_p)^2 \cdot H_D(\phi)$$

n_p = Number of peaks

ϕ_p = Position of peak, p , in H_D

w_p = Points in peak p

r = Normalisation factor



(MR)SAR

[Mao'92]

- Each pixel is a random variable whose value is estimated from its neighboring pixels + noise

- A kind of Markov Random Field model



- **SAR Model (Simultaneous Autoregressive)**

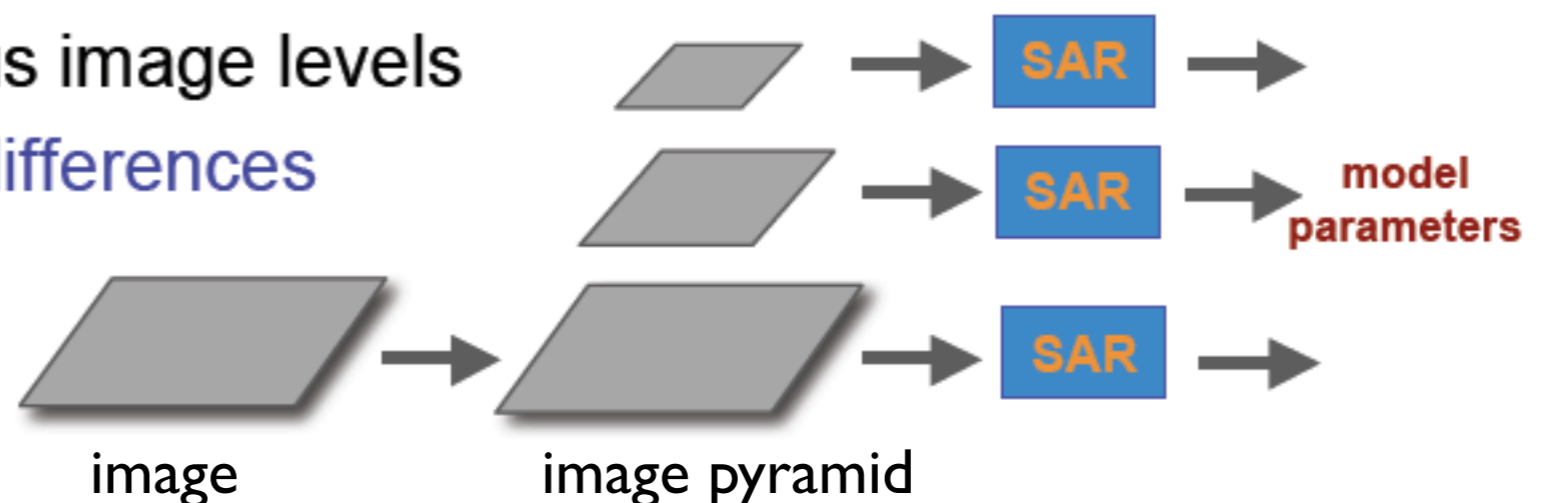
- Describes each pixel in terms of its neighboring pixels.

- **MRSAR Model (MultiResolution SAR)**

- Describing granularities by representing textures at variety of resolutions

- SAR applied at various image levels

- Metric → parameter differences



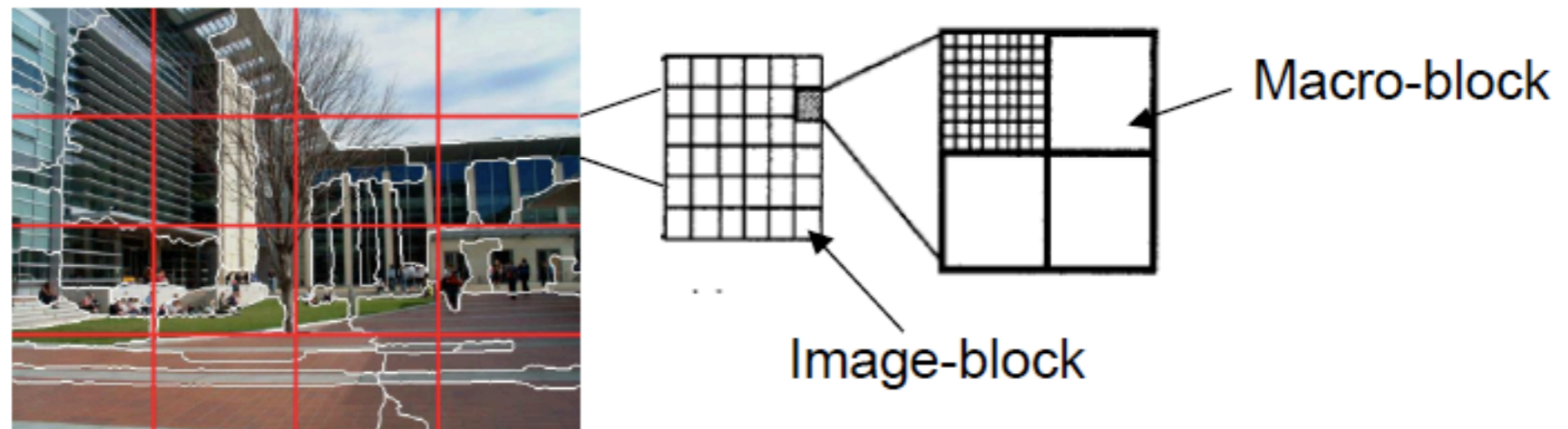
Edge Histogram

- Edge histogram (EHD)
- Captures the spatial distribution of the edge in six statues: 0° , 45° , 90° , 135° , non direction and no edge.

- Utilizing the filters



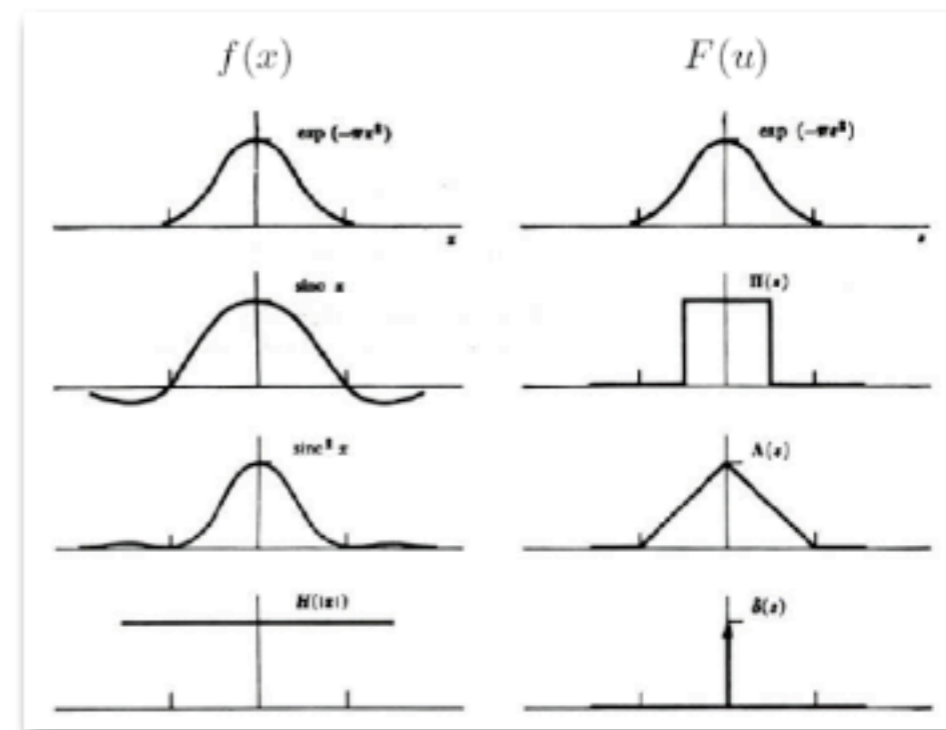
- Global EHD of an image: Concatenating 16 sub EHDs into a 96 bins
- Local EHD of a segment
 - Grouping the edge histogram of the image-blocks fallen into the segment



The Fourier Transform

- Represent function on a new basis
 - Think of functions as vectors, with many components
 - We now apply a linear transformation to transform the basis
 - dot product with each basis element
- In the expression, u and v select the basis element, so a function of x and y becomes a function of u and v
- basis elements have the form $e^{-i2\pi(ux+vy)}$

$$F(g(x, y))(u, v) = \iint_{\mathbb{R}^2} g(x, y) e^{-i2\pi(ux+vy)} dx dy$$

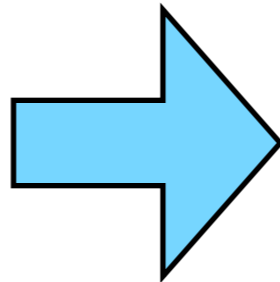




Zebra



Zebra

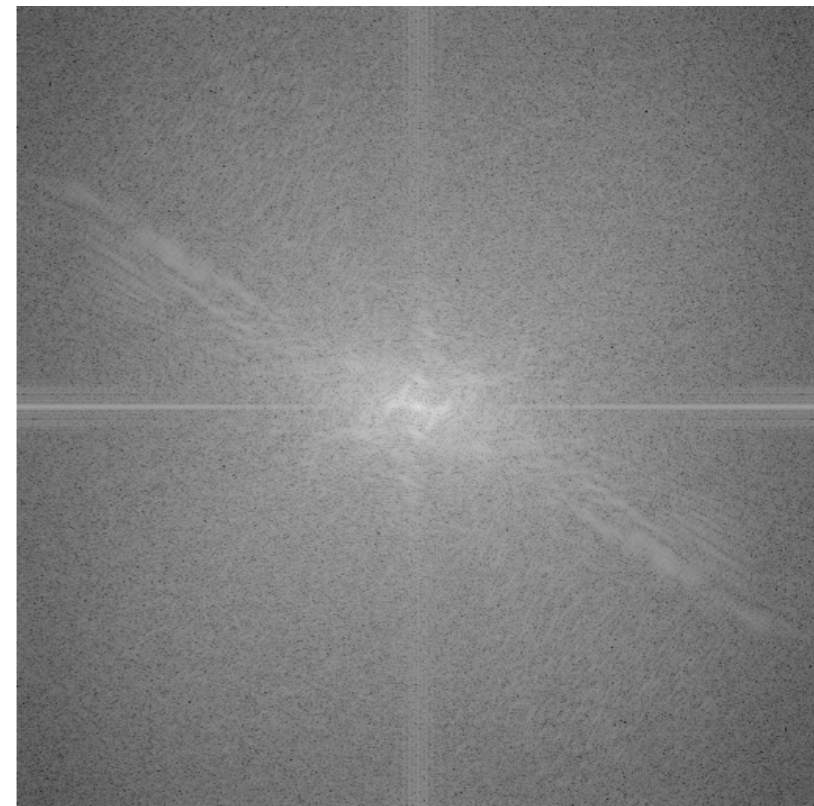


Fourier
Transform

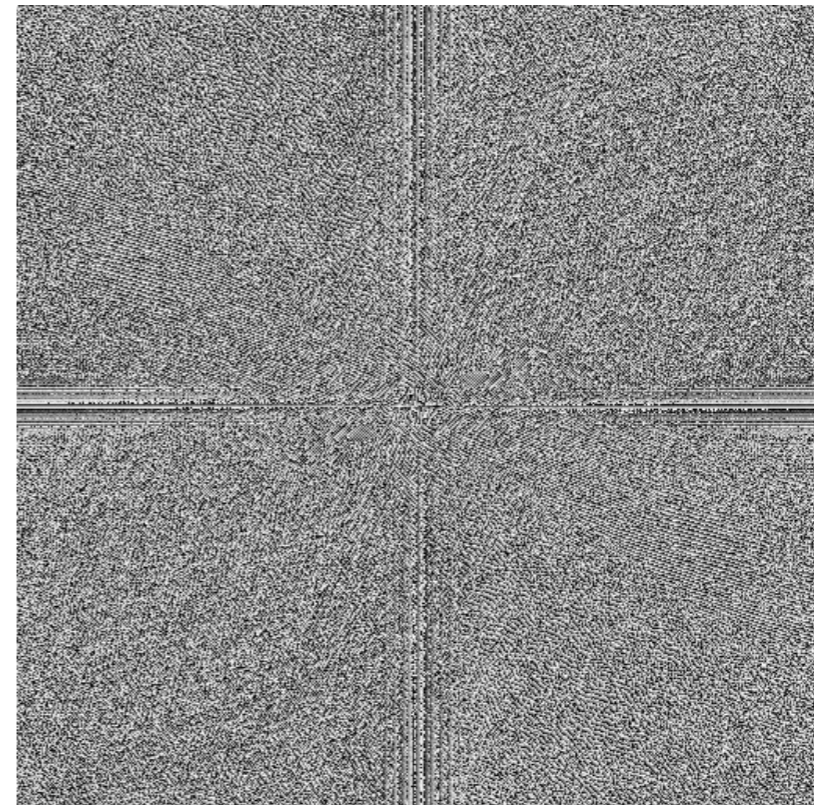


Zebra

Fourier
Transform



magnitude transform



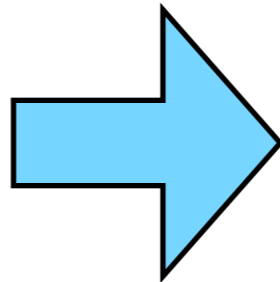
phase transform



Leopard



Leopard

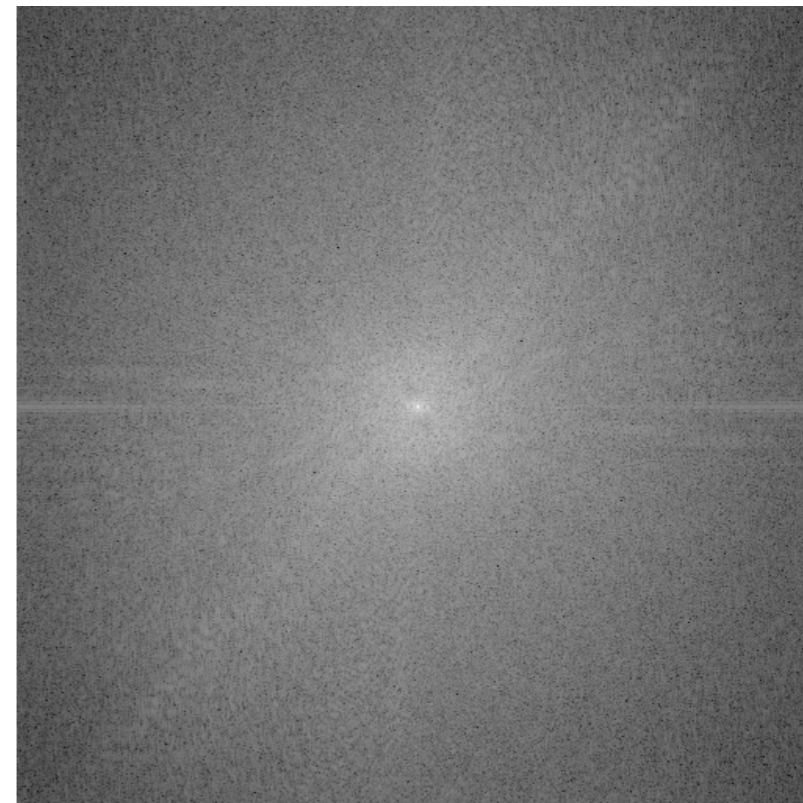


**Fourier
Transform**

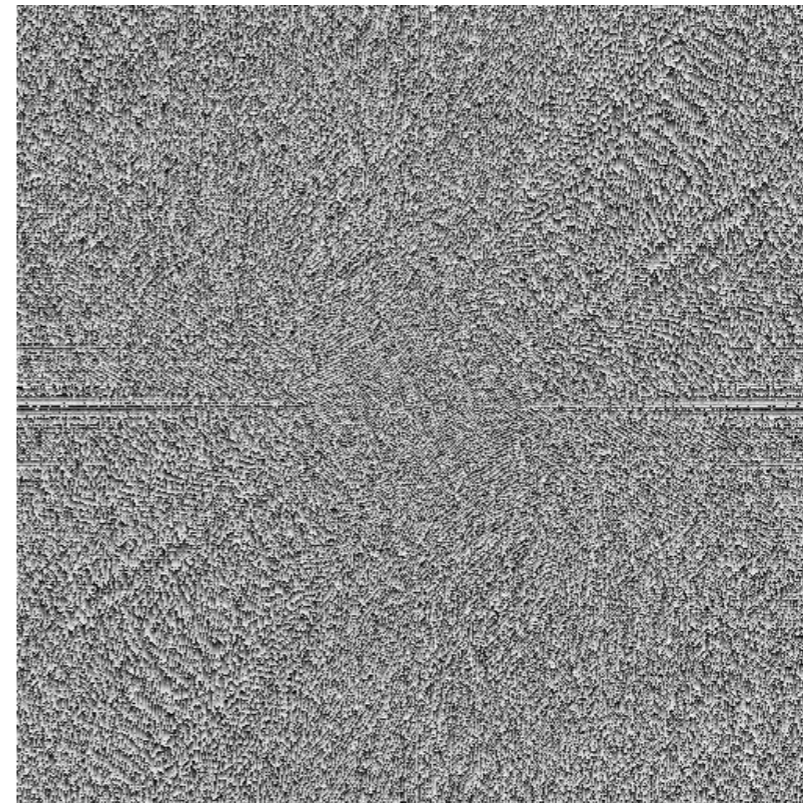


Leopard

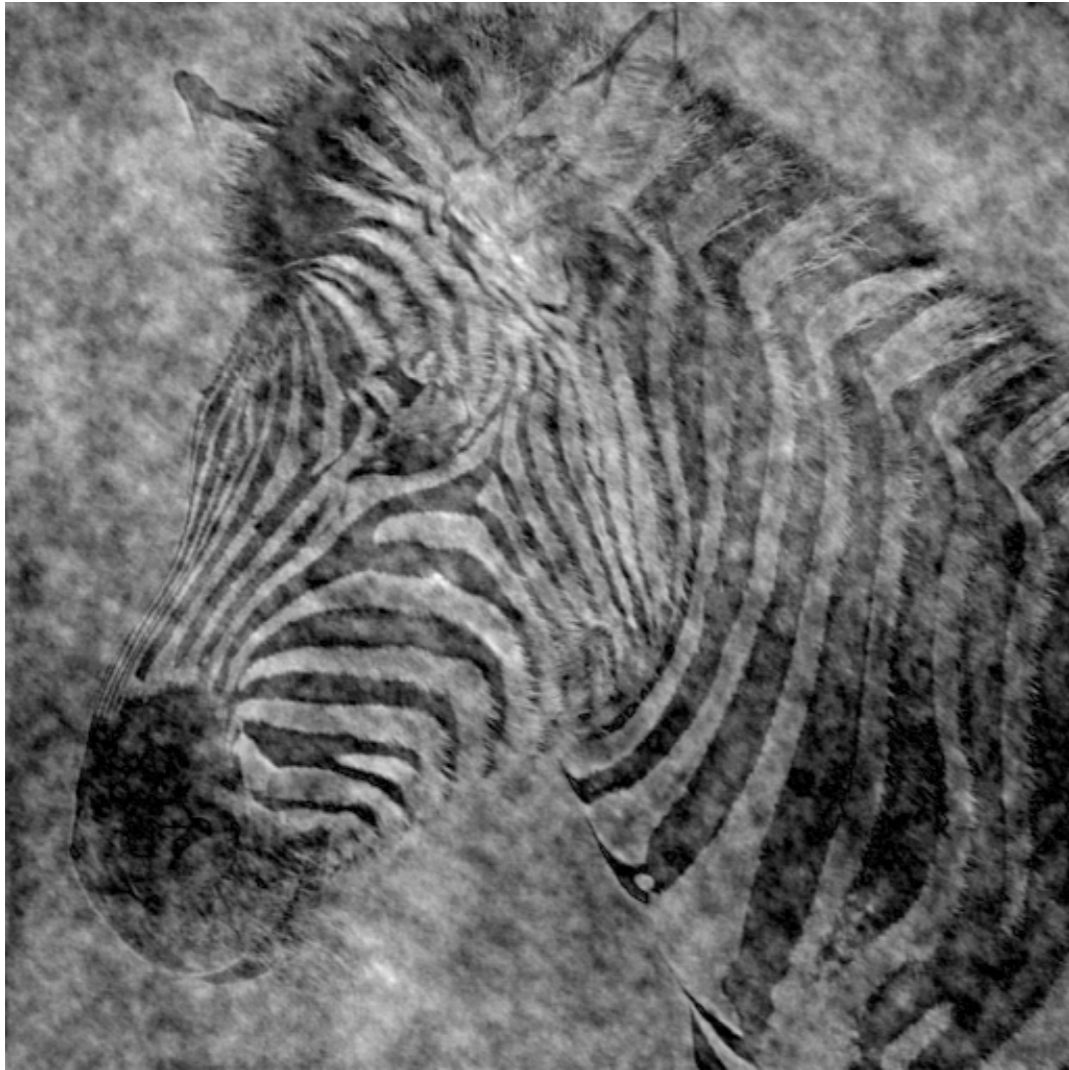
Fourier
Transform



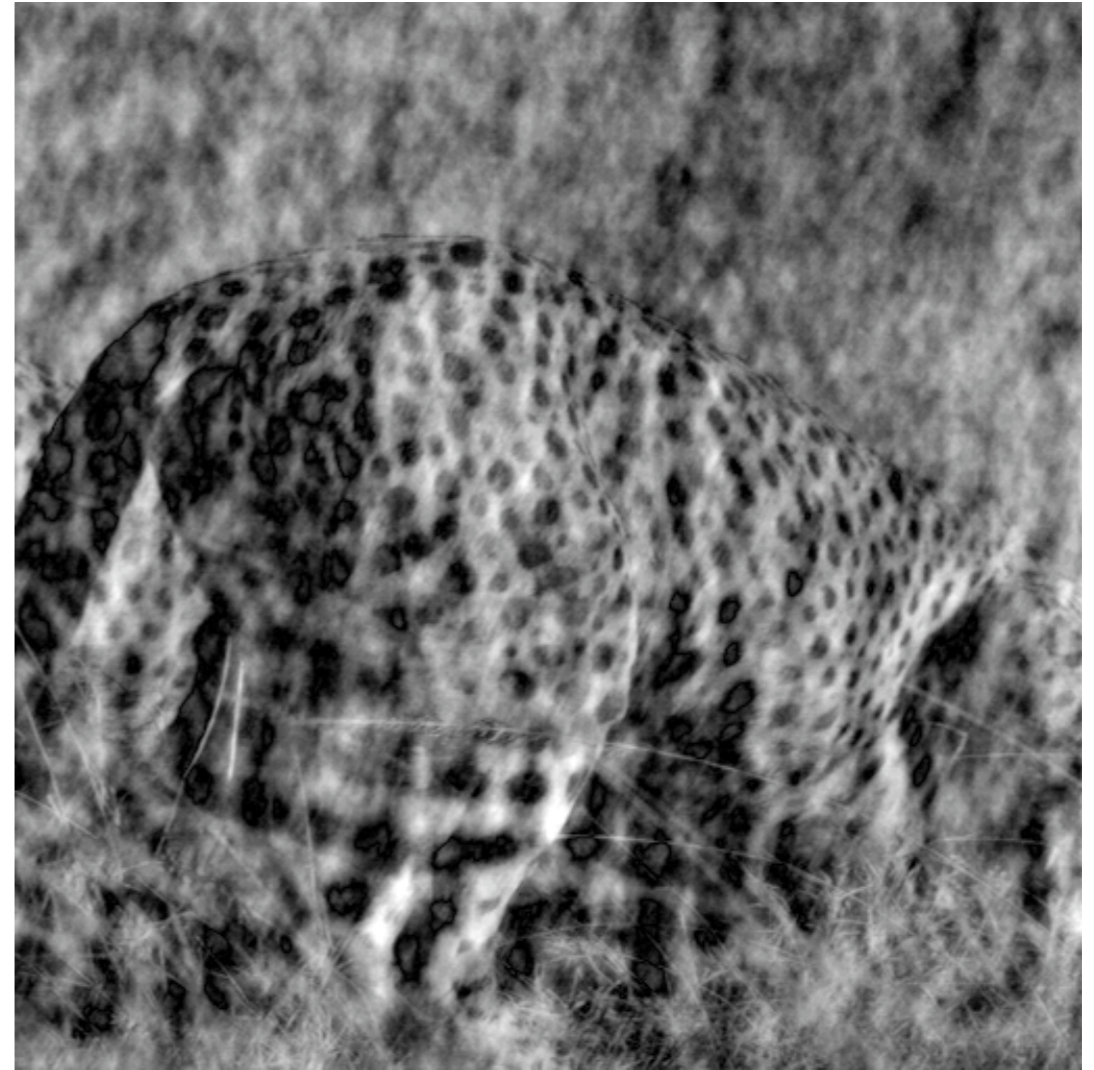
magnitude transform



phase transform

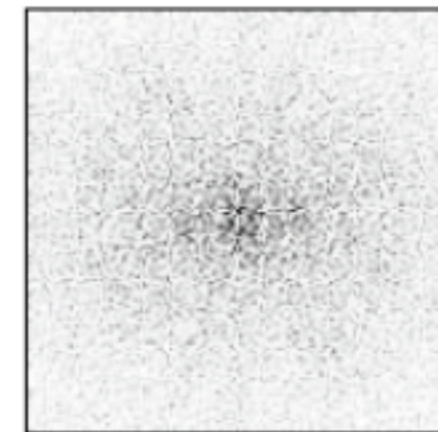
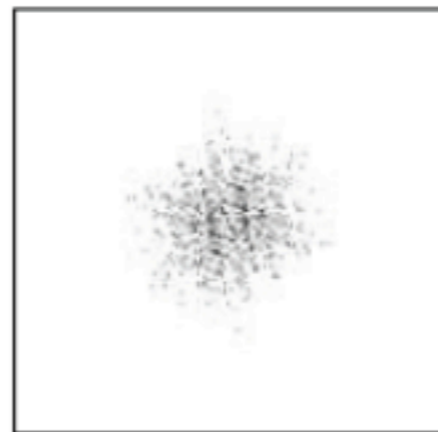
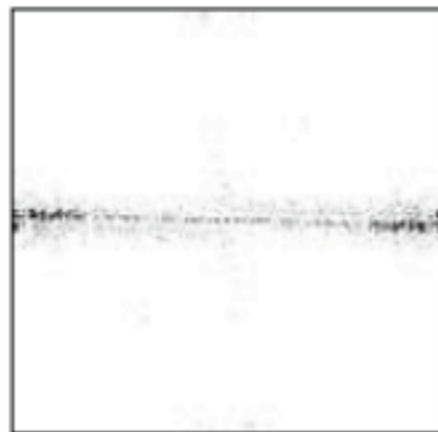
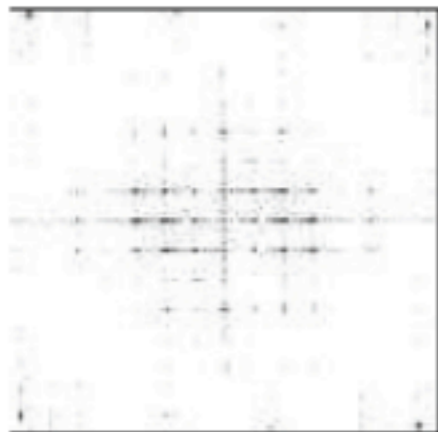
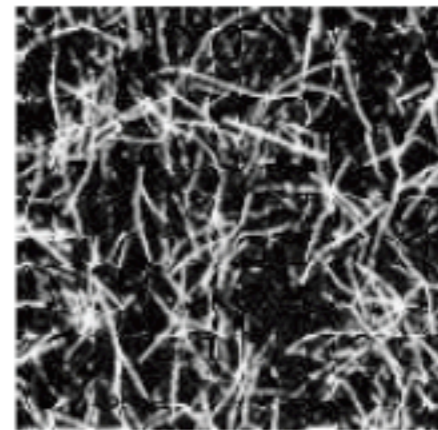
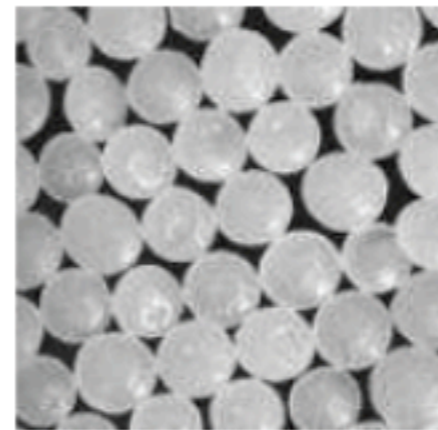
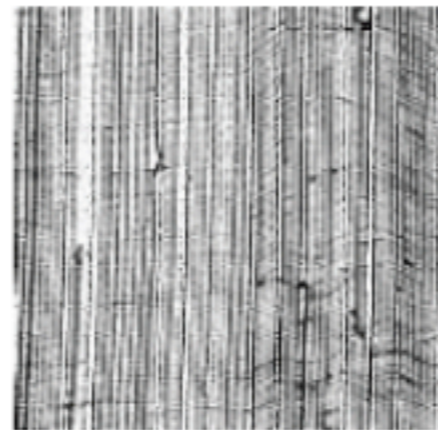


Zebra's phase
+ Leo's mag



Leo's phase
+ Zebra's mag

Natural Images and Their FT



(a) structured

(b) oriented

(c) granular

(d) random

- What happened to the FT patterns when the texture scale and orientation are changed?

Frequency Domain Features

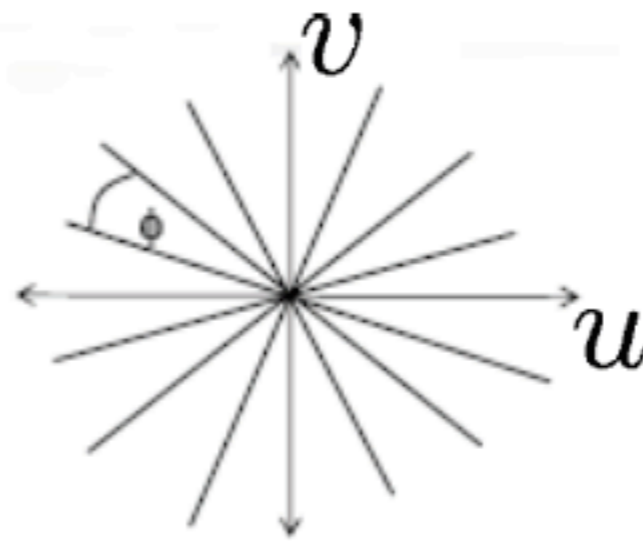
Fourier domain energy distribution

- Angular features (directionality)

$$V_{\theta_1\theta_2}^{(a)} = \int \int |F(u, v)|^2 du dv$$

where,

$$\theta_1 \leq \tan^{-1} \left[\frac{v}{u} \right] \leq \theta_2$$

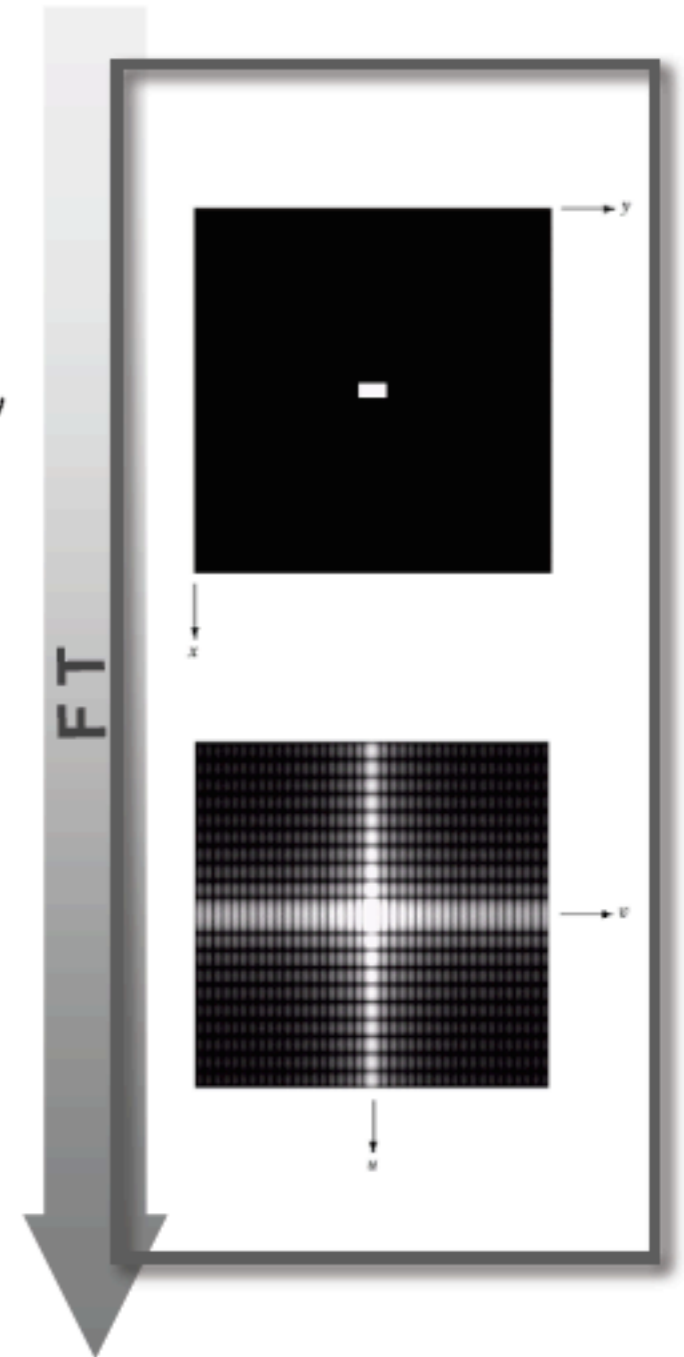
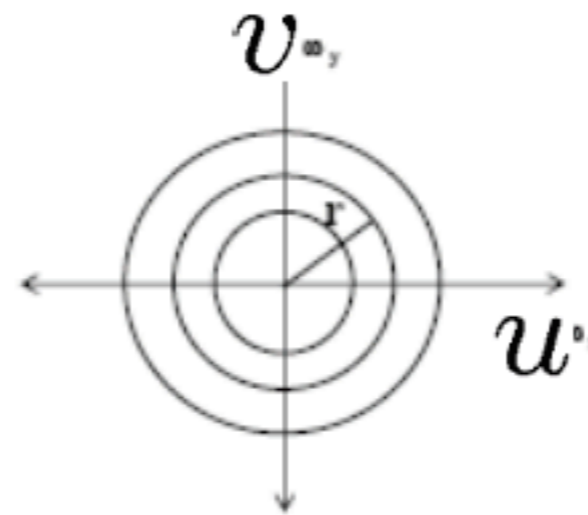


- Radial features (coarseness)

$$V_{r_1r_2}^{(r)} = \int \int |F(u, v)|^2 du dv$$

where,

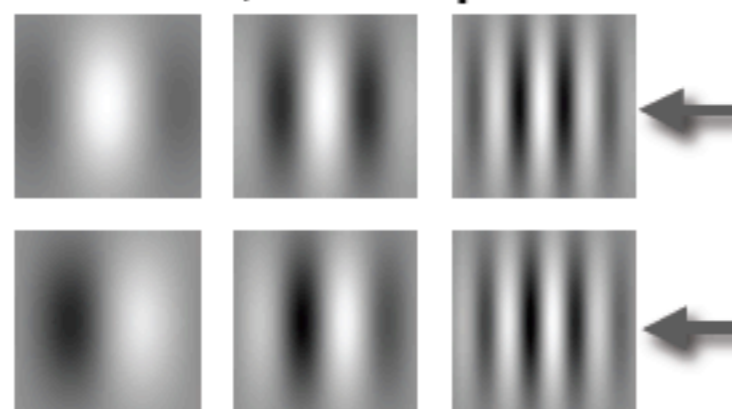
$$r_1 \leq u^2 + v^2 < r_2$$



Uniform division may not be the best!!

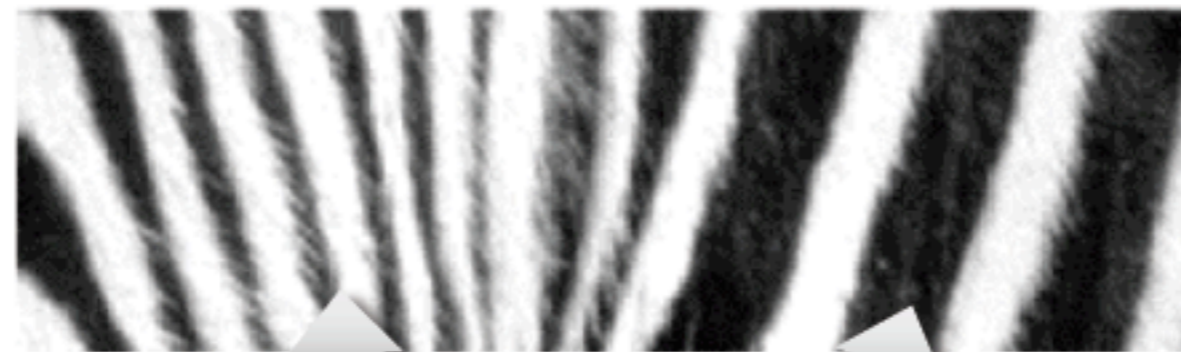
Gabor Texture

- Fourier coefficients depend on the entire image (Global) → we lose spatial information
- **Objective**: local spatial frequency analysis
- Gabor kernels: looks like Fourier basis multiplied by a Gaussian
 - The product of a symmetric (even) Gaussian with an oriented sinusoid
 - Gabor filters come in pairs: symmetric and anti-symmetric (odd)
 - Each pair recover symmetric and anti-symmetric components in a particular direction
 - (k_x, k_y) : the spatial frequency to which the filter responds strongly
 - σ : the scale of the filter. When $\sigma = \text{infinity}$, similar to FT
- We need to apply a number of Gabor filters are different scales, orientations, and spatial frequencies

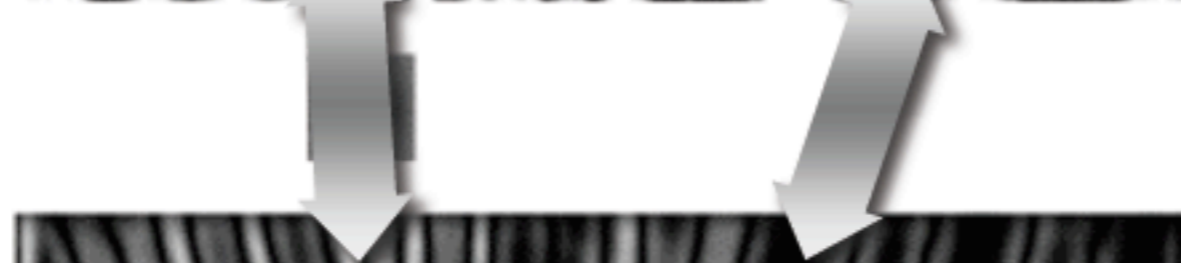

$$G_{\text{symmetric}}(x, y) = \cos(k_x x + k_y y) \exp -\frac{x^2 + y^2}{2\sigma^2}$$
$$G_{\text{anti-symmetric}}(x, y) = \sin(k_x x + k_y y) \exp -\frac{x^2 + y^2}{2\sigma^2}$$

Example – Gabor Kernel

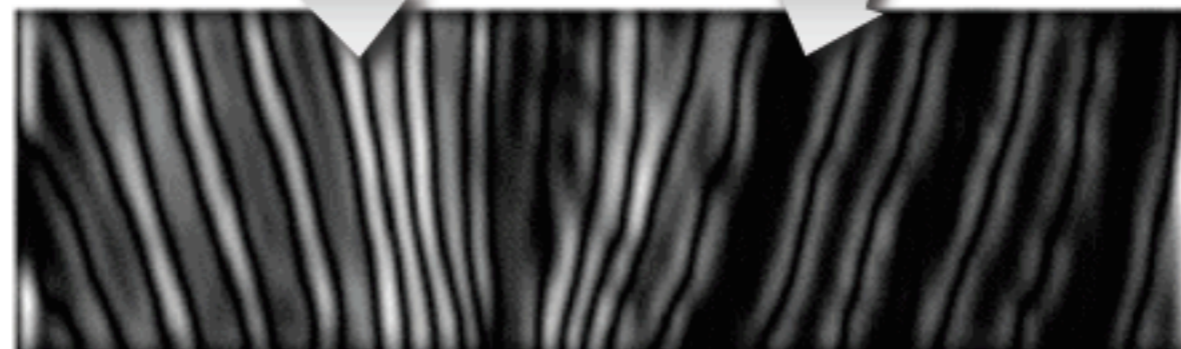
- Zebra stripes at different scales and orientations and convolved with the Gabor kernel
- The response falls off when the stripes are larger or smaller
- The response is large when the spatial frequency of the bars roughly matches the windowed by the Gaussian in the Gabor kernel
- Local spatial frequency analysis



zebra image

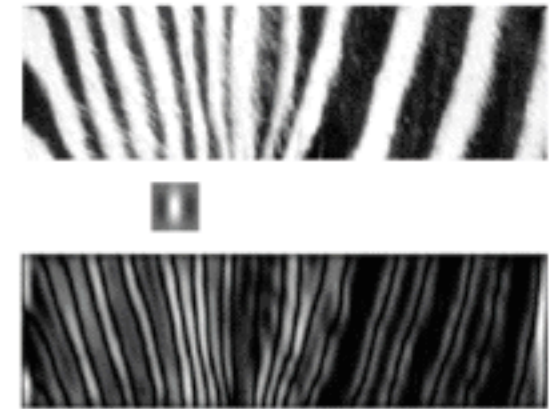


Gabor kernel



magnitude of
the filtered image

Gabor Texture (cont.)



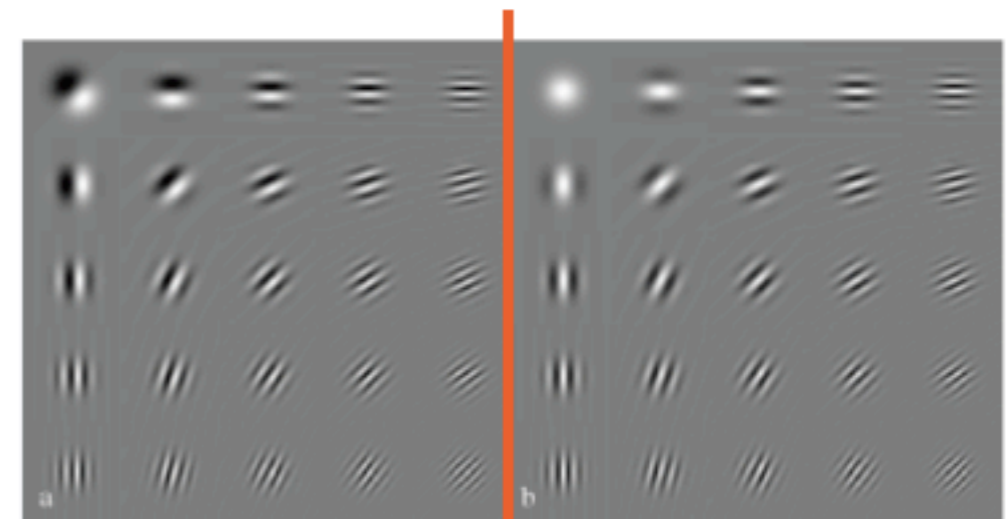
- Image $I(x,y)$ convoluted with Gabor filters h_{mn} (totally $M \times N$)

$$W_{mn}(x, y) = \int I(x_1, y_1) h_{mn}(x-x_1, y-y_1) dx_1 dy_1$$

- Using first and 2nd moments for each scale and orientations

$$\mu_{mn} = \int \int |W_{mn}(x, y)| dx dy$$

$$\sigma_{mn} = \sqrt{\int \int (|W_{mn}(x, y)| - \mu_{mn})^2 dx dy}$$

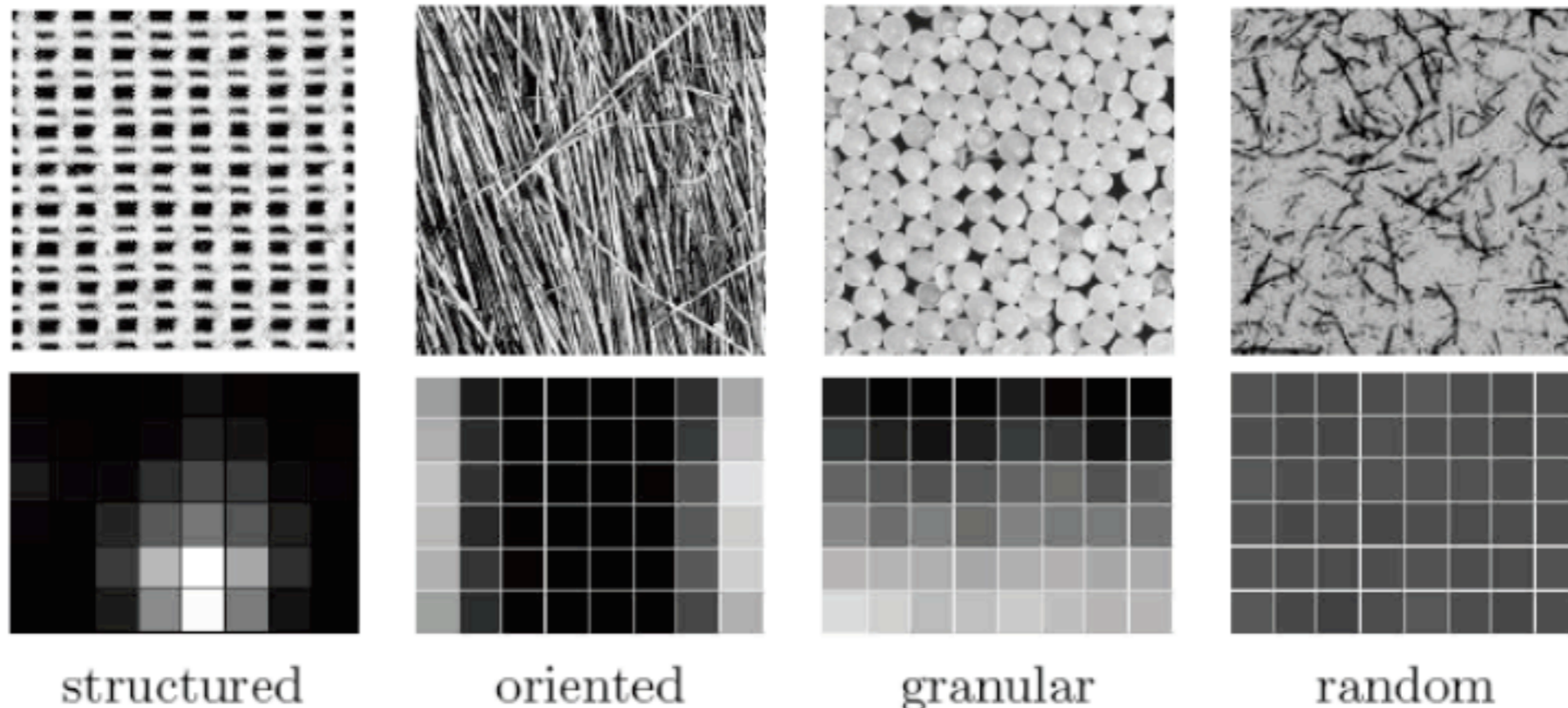


- Features:** e.g., 4 scales, 6 orientations
→ 48 dimensions

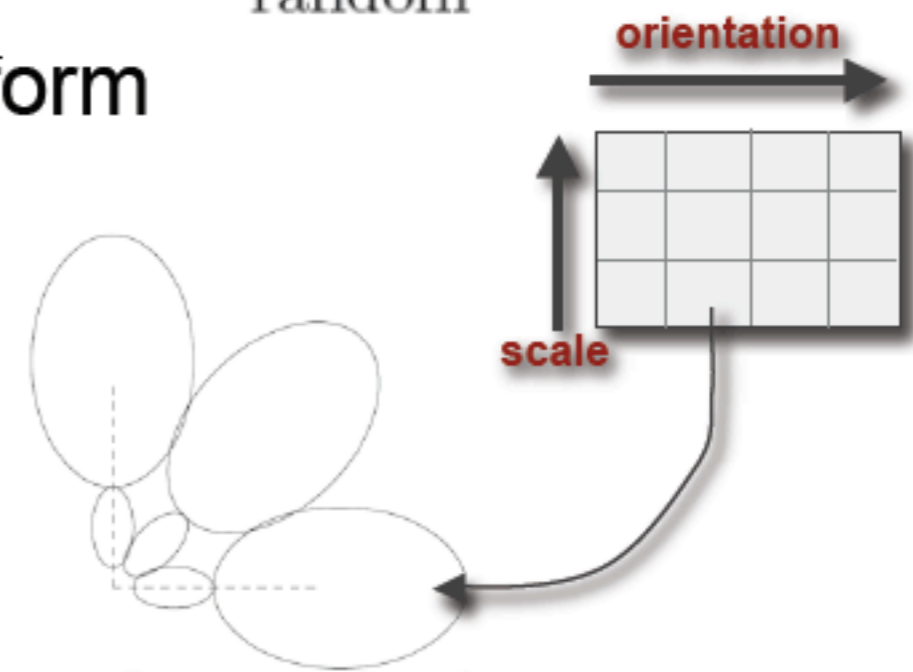
$$\bar{v} = [\mu_{00}, \sigma_{00}, \mu_{01}, \dots, \mu_{35}, \sigma_{35}]$$

Gabor kernels

Gabor Texture (cont.)



- Arranging the mean energy in a 2D form
 - structured: localized pattern
 - oriented (or directional): column pattern
 - granular: row pattern
 - random: random pattern



Wavelet Features (PWT, TWT)

■ Wavelet

- Decomposition of signal with a family of basis functions with recursive filtering and sub-sampling
- Each level, decomposes 2D signal into 4 subbands, LL, LH, HL, HH (L=low, H=high)

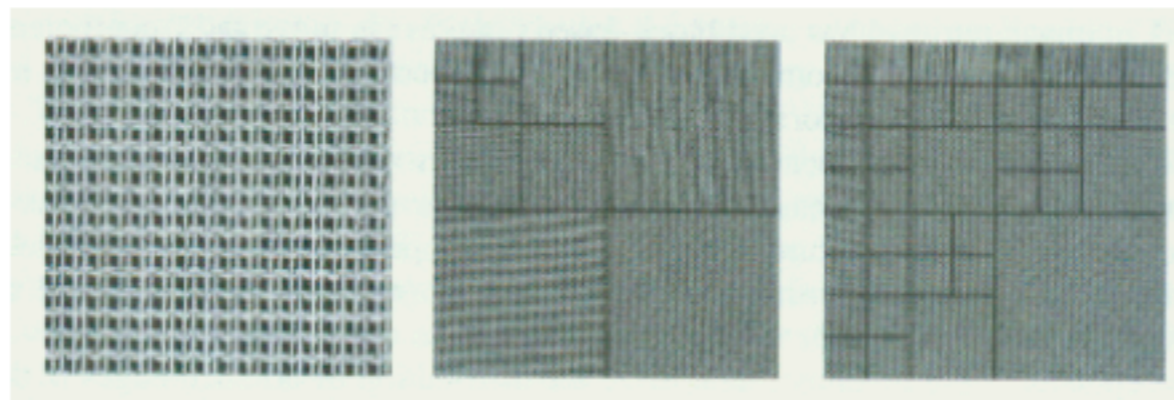
■ PWT: pyramid-structured wavelet transform

- Recursively decomposes the LL band
- Feature dimension $(3 \times 3 \times 1 + 1) \times 2 = 20$

■ TWT: pyramid-structured wavelet transform

- Some information in the middle frequency channels
- Feature dimension $40 \times 2 = 80$

LL3	HL3	HL2	HL1
LH3	HH3		
LL2		HH2	
LH1			HH1



original image

PWT

TWT

Texture Comparisons

[Ma'98]

- Retrieval performance of different texture features according to the number of relevant images retrieved at various scopes using [Corel Photo galleries](#)

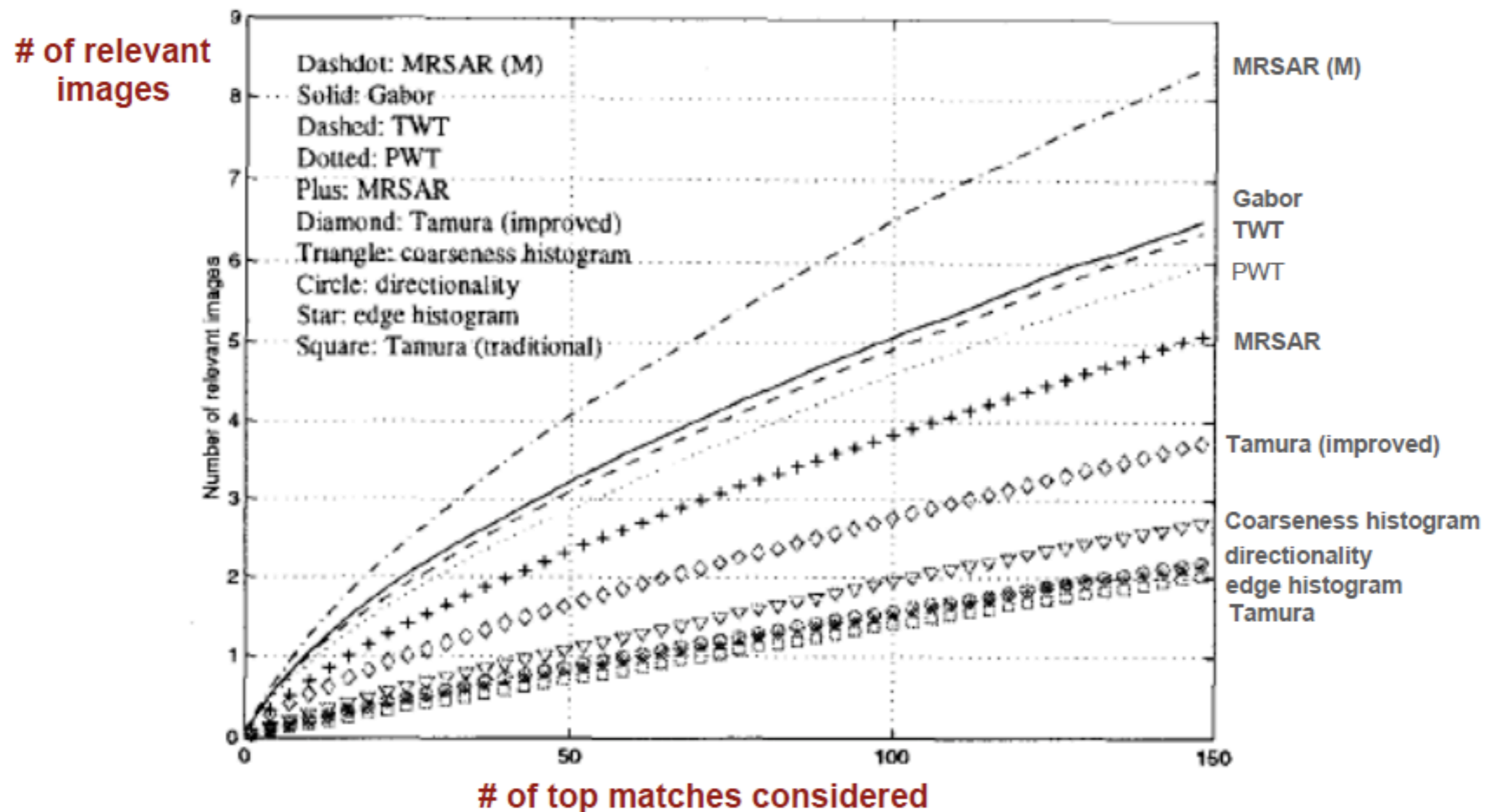
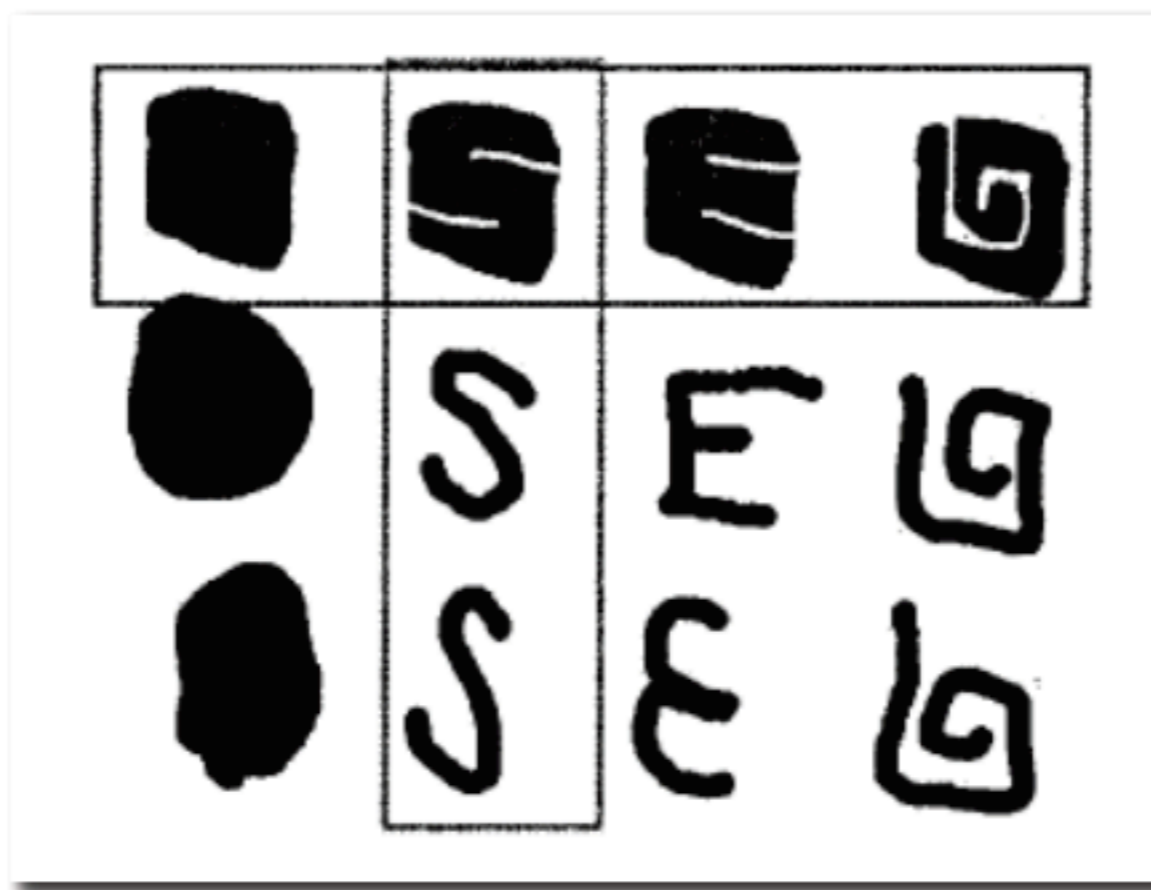


Image shape features

- Shape features are computed out based on object segments or regions, mainly including
 - contour features
 - and regions features.
- Typical approaches include
 - Fourier shape description
 - Moment invariants



Region-based vs. Contour-based Descriptor



- Columns indicate contour similarity
 - Outline of contours
- Rows indicate region similarity
 - Distribution of pixels

Region-based Descriptor

- Express pixel distribution within a 2D object region
- Employs a complex 2D Angular Radial Transformation (ART)
 - 35 fields each of 4 bits
- Rotational and scale invariance
- Robust to some non-rigid transformation
- L_1 metric on transformed coefficients
- Advantages
 - Describing complex shapes with disconnected regions
 - Robust to segmentation noise
 - Small size
 - Fast extraction and matching



Contour-based Descriptor

- It's based on Curvature (曲率) Scale-Space (CSS) representation
- Found to be superior to
 - Zernike moments
 - ART
 - Fourier-based
 - Turning angles
 - Wavelets
- Rotational and scale invariance
- Robust to some non-rigid transformations
- For example
 - Applicable to (a)
 - Discriminating differences in (b)
 - Finding similarities in (c)-(e)



(a)



(b)



(c)



(d)



(e)

Problems in Shape-based Indexing

Many existing approaches assume

- Segmentation is given
- Human operator circle object of interest
- Lack of clutter and shadows
- Objects are rigid
- Planar (2-D) shape models
- Models are known in advance

Dimensional reduction for image features

In image retrieval system, increasing feature dimension can enhance precision of retrieval greatly. However, high feature dimension will lead to high computation cost. Hence it is important to reduce the redundant in feature data.

- Image feature space reduction
 - Linear dimensional reduction techniques: PCA ...
 - Nonlinear dimensional reduction techniques: Isomap, LLE ...
 - Clustering based feature reduction methods
- High-dimensional feature indexing
 - Database oriented high-dimensional data indexing
 - Bucketing grouping searching techniques, K-d tree, R tree ...
 - Clustering methods
 - SOM



Image similarities

- How to measure similarity of different images based on features?
 - Image features always form into a fixed-length feature vector.
 - The similarity therefore can be measured by
 - Euclidean distance
 - Histogram intersection
 - Quadratic distance
 - Mahalanobis distance (马氏距离)
 - Non-geometrical similarity



Similarity and distance

- Similarity:



- distance:



Practical image retrieval systems

- QBIC (Query By Image Content)
 - <http://www.qbic.almaden.ibm.com/>
- Virage
 - <http://www.virage.com/cgi-bin/query-e>
- RetrievalWare
 - <http://vrw.excalib.com/cgi-bin/sdk/cst/cst2.bat>
- Photobook
- MARS
 - <http://jadzia.ifp.uiuc.edu:8000>

Practical image retrieval systems (cont.)

- Most existing image retrieval systems have one or more of following functions features:
 - Random browsing
 - Classified browsing
 - Example based retrieval
 - Sketch based retrieval
 - Texture based retrieval



Future of image retrieval

- Human-computer interaction
- Semantic speech
- Web-oriented
- High dimensional data
- Perspective
- Multiple media channels
- Image feature mapping
- Standards of performance measurements
- Construction of test sets



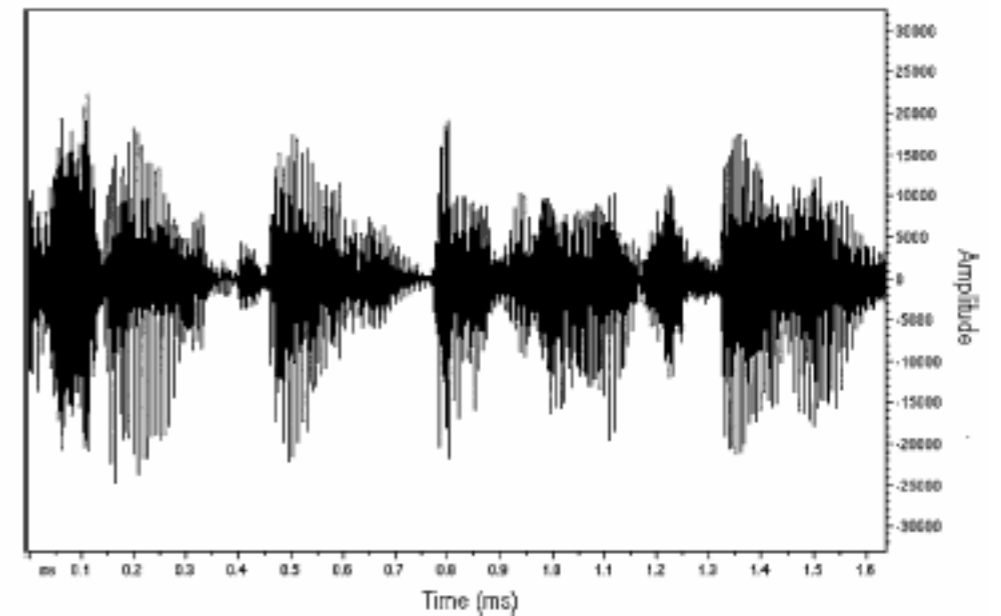
2. music retrieval techniques



reference:

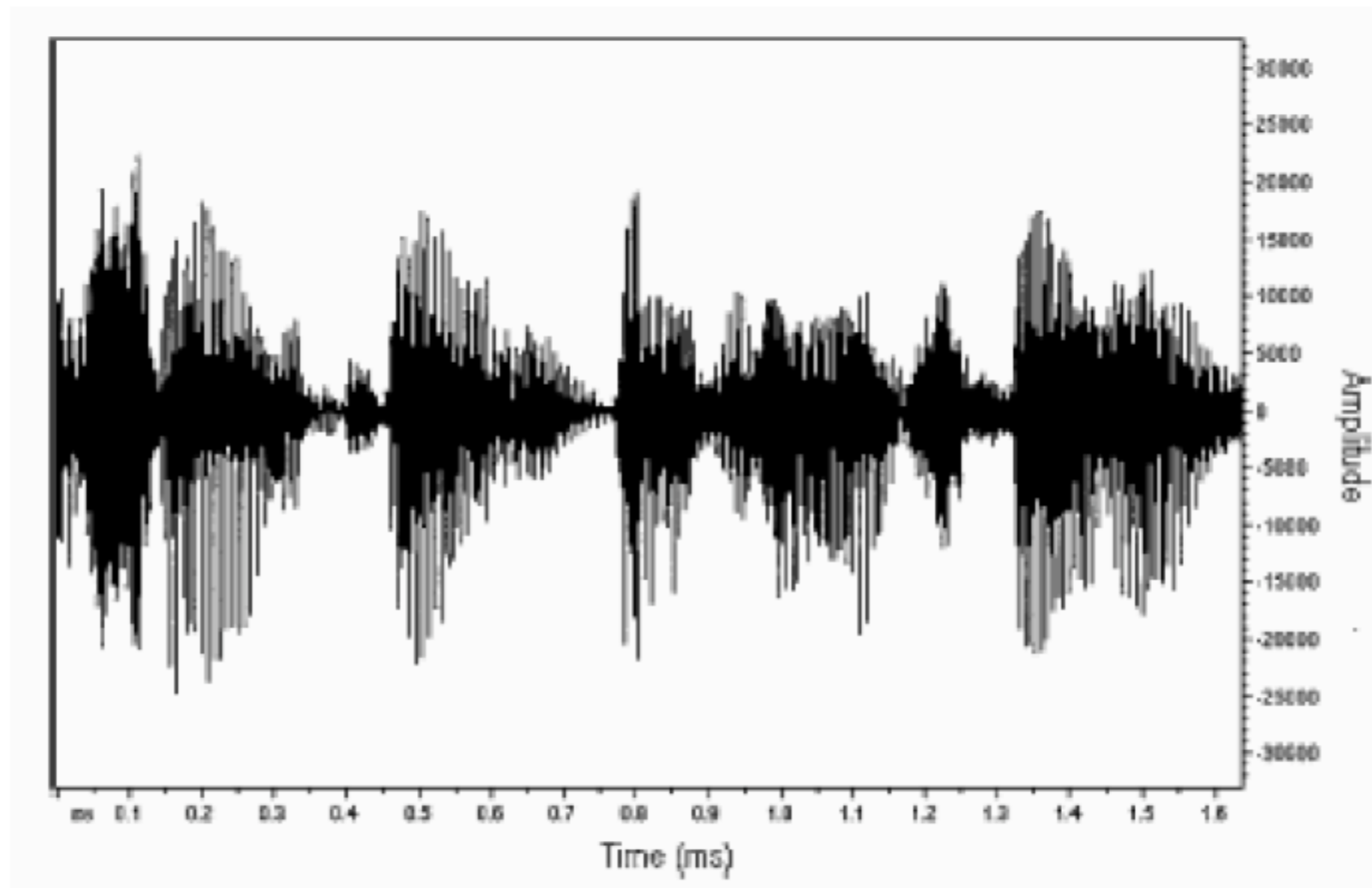
Main Audio Features

- Time-Domain Features
 - Average Energy
 - Zero Crossing Rate
 - Silence Ratio
- Frequency-Domain Features
 - Sound Spectrum
 - Bandwidth
 - Energy Distribution
 - Harmonicity
 - Pitch
- Spectrogram



Time-Domain Features

- Amplitude-time representation of an audio signal



Time-Domain Features (2)

- Average Energy

- Indicates the loudness of the audio signal

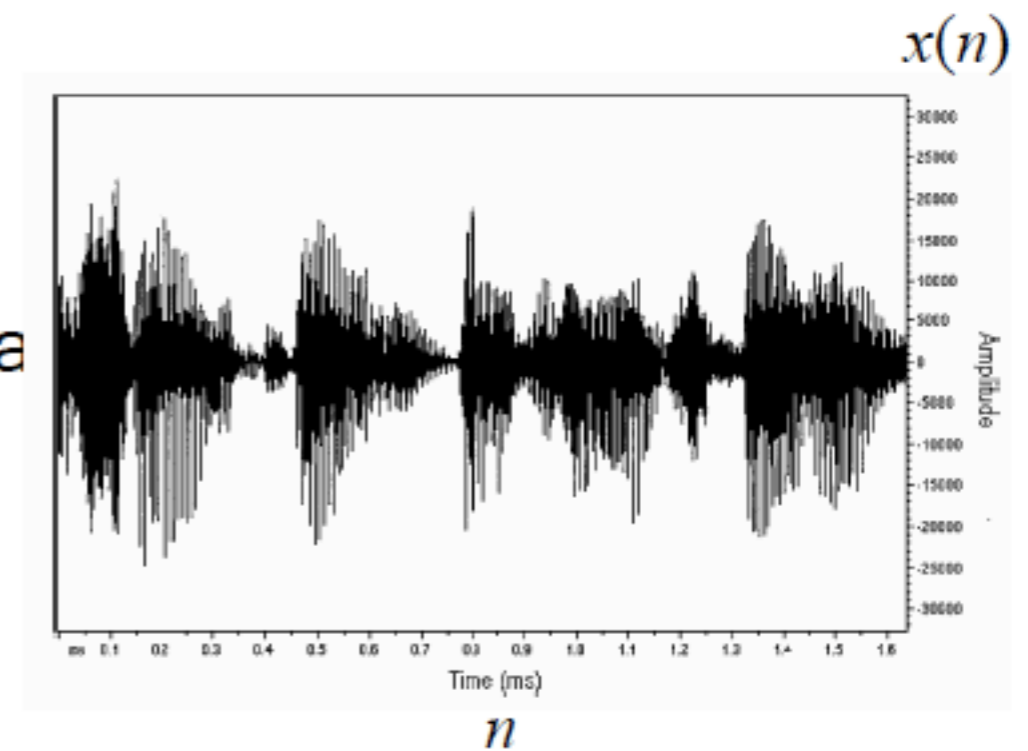
$$E = \frac{\sum_{n=1}^{N-1} x(n)^2}{N}$$

- Zero Crossing Rate

- Indicates the frequency of signal amplitude sign change

$$ZC = \frac{\sum_{n=1}^{N-1} \text{sgn}[x(n)] - \text{sgn}[x(n-1)]}{2N}$$

$$\text{sgn}(a) = \begin{cases} 1 & a > 0 \\ 0 & a = 0 \\ -1 & a < 0 \end{cases}$$

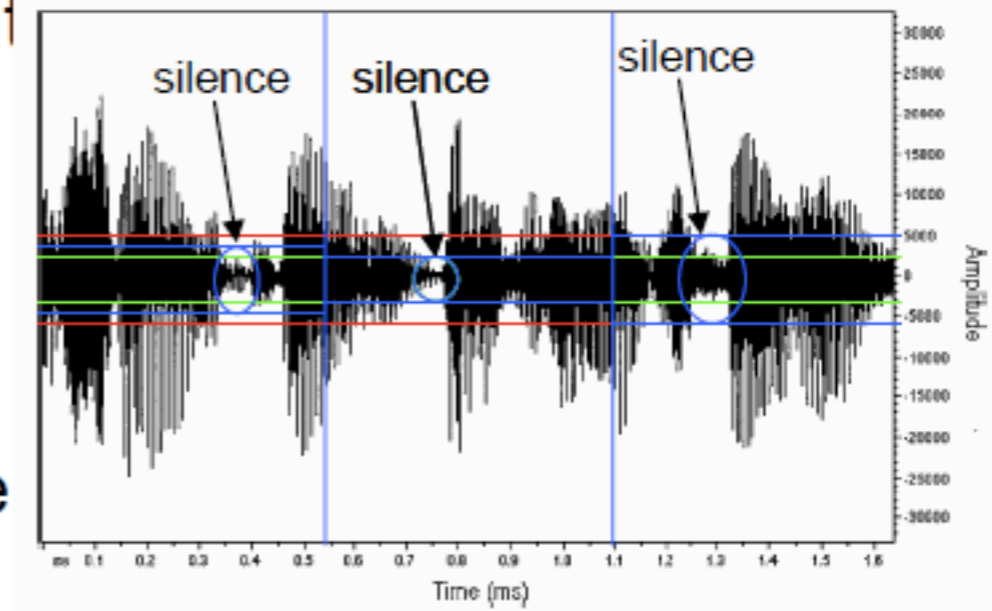


Time-Domain Features (3)

- Silence Ratio
 - Indicates the proportion of the sound piece that is silent
 - Silence is a period within which the absolute amplitude values of a certain number of samples are below a certain threshold
 - Silence ratio is calculated as the ratio between the sum of silent periods and the total length

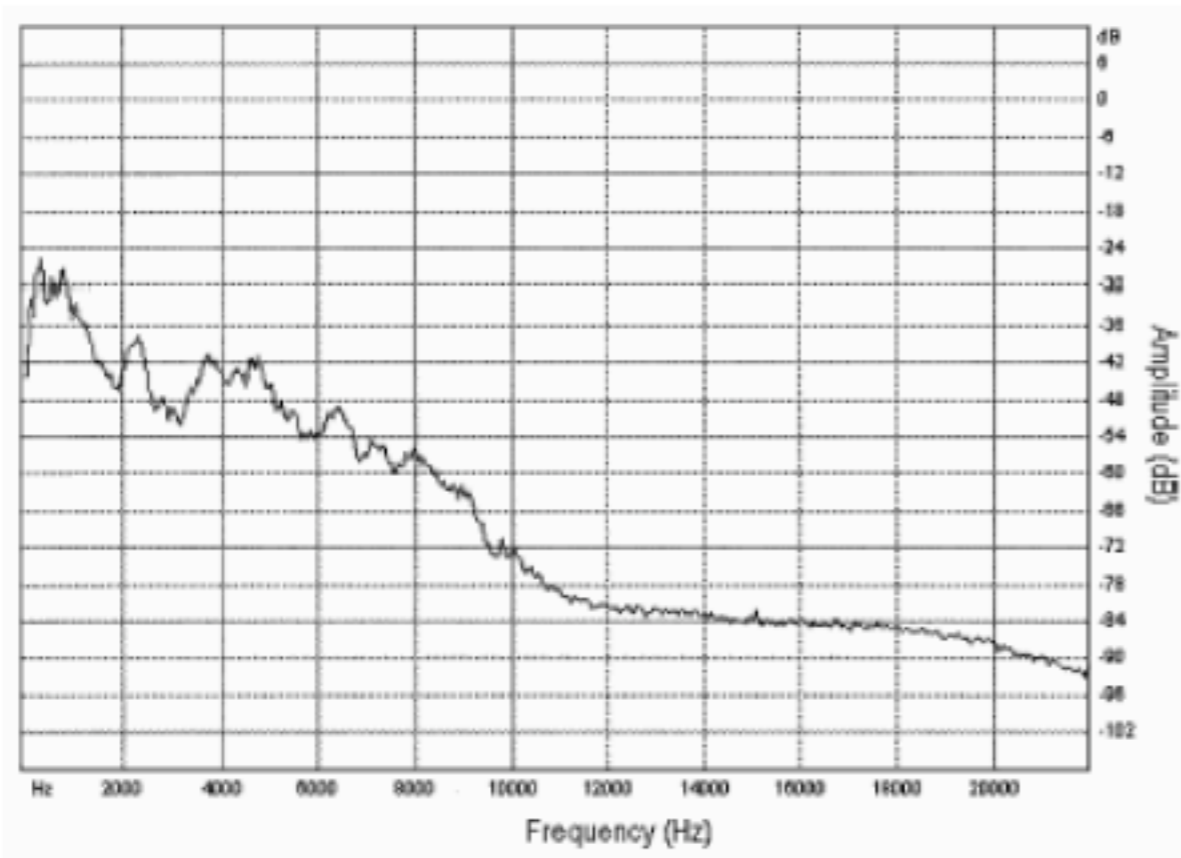
Approaches:

1. Fixed Threshold
2. Select Reference Silence Value
3. Adaptive Silence Thresholds



Frequency-Domain Features

- Sound Spectrum



Discrete Fourier Transform (DFT)

$$X(k) = \sum_{n=0}^{N-1} x(n)e^{-\frac{j2\pi nk}{N}}$$

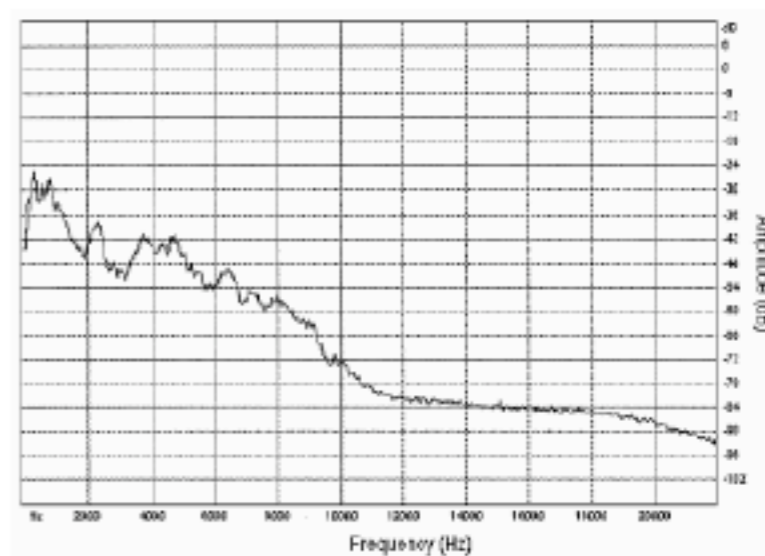
Inverse Discrete Fourier Transform (IDFT)

$$x(n) = \frac{1}{N} \sum_{k=0}^{N-1} X(k)e^{\frac{j2\pi nk}{N}}$$

- For large value of N , the signal is often broken into blocks called frames and DFT is applied to each of the frames.

Frequency-Domain Features (2)

- Bandwidth
 - indicated the frequency range of a sound
 - can be taken as the difference between the highest frequency and lowest frequency of non-zero spectrum components
 - “non-zero” may be defined as at least 3dB above the silence level
- Energy distribution
 - Signal distribution across frequency components
 - One important feature derived from the energy distribution is the *centroid*, which is the mid-point of the spectral energy distribution of a sound. Centroid is also called *brightness*



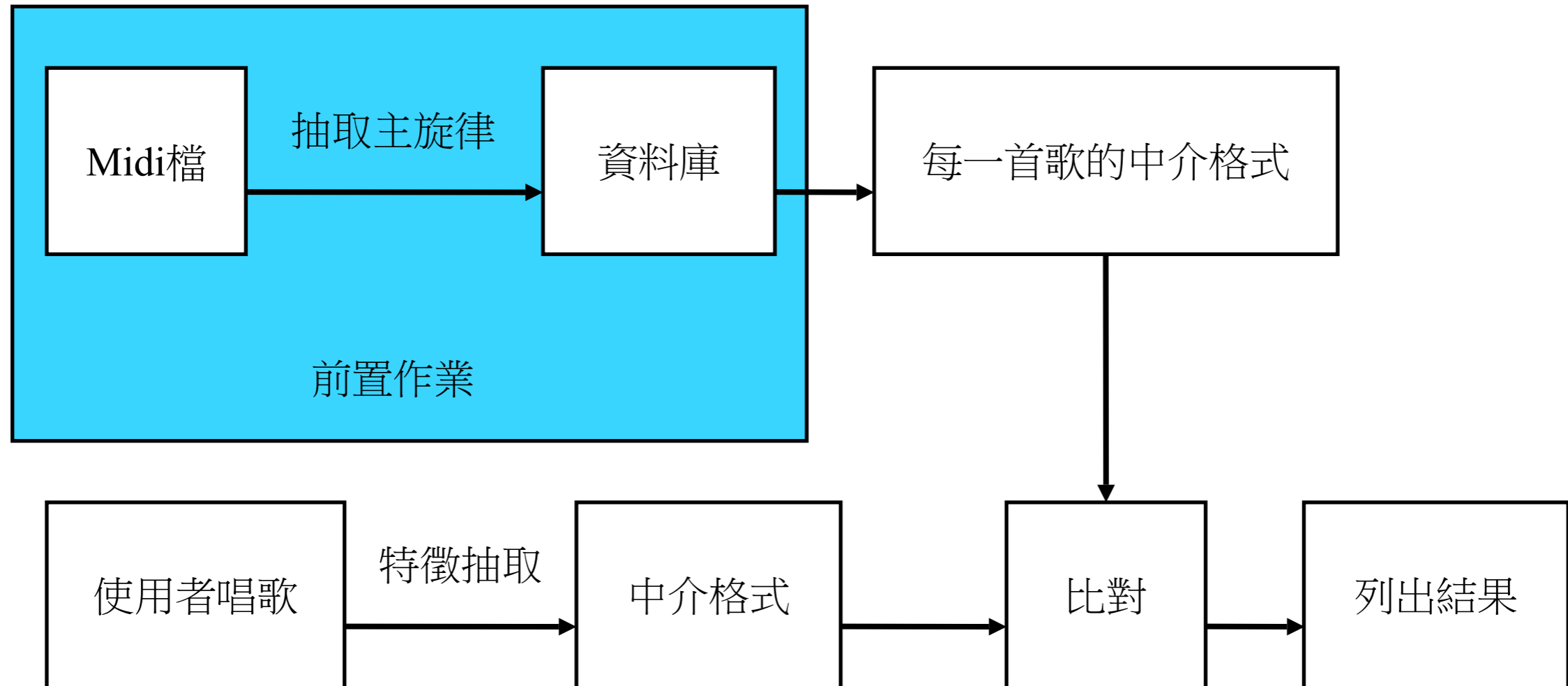
Frequency-Domain Features (3)

- Harmonicity
 - In harmonic sound, the spectral components are mostly whole number multiples of the lowest and most often loudest frequency
 - Lowest frequency is called *fundamental frequency*
 - Music is normally more harmonic than other sounds
- Pitch
 - the distinctive quality of a sound, dependent primarily on the frequency of the sound waves produced by its source
 - only period sounds, such as those produced by musical instruments and the voice, give rise to a sensation of a pitch
 - In practice, we use the fundamental frequency as the approximation of the pitch

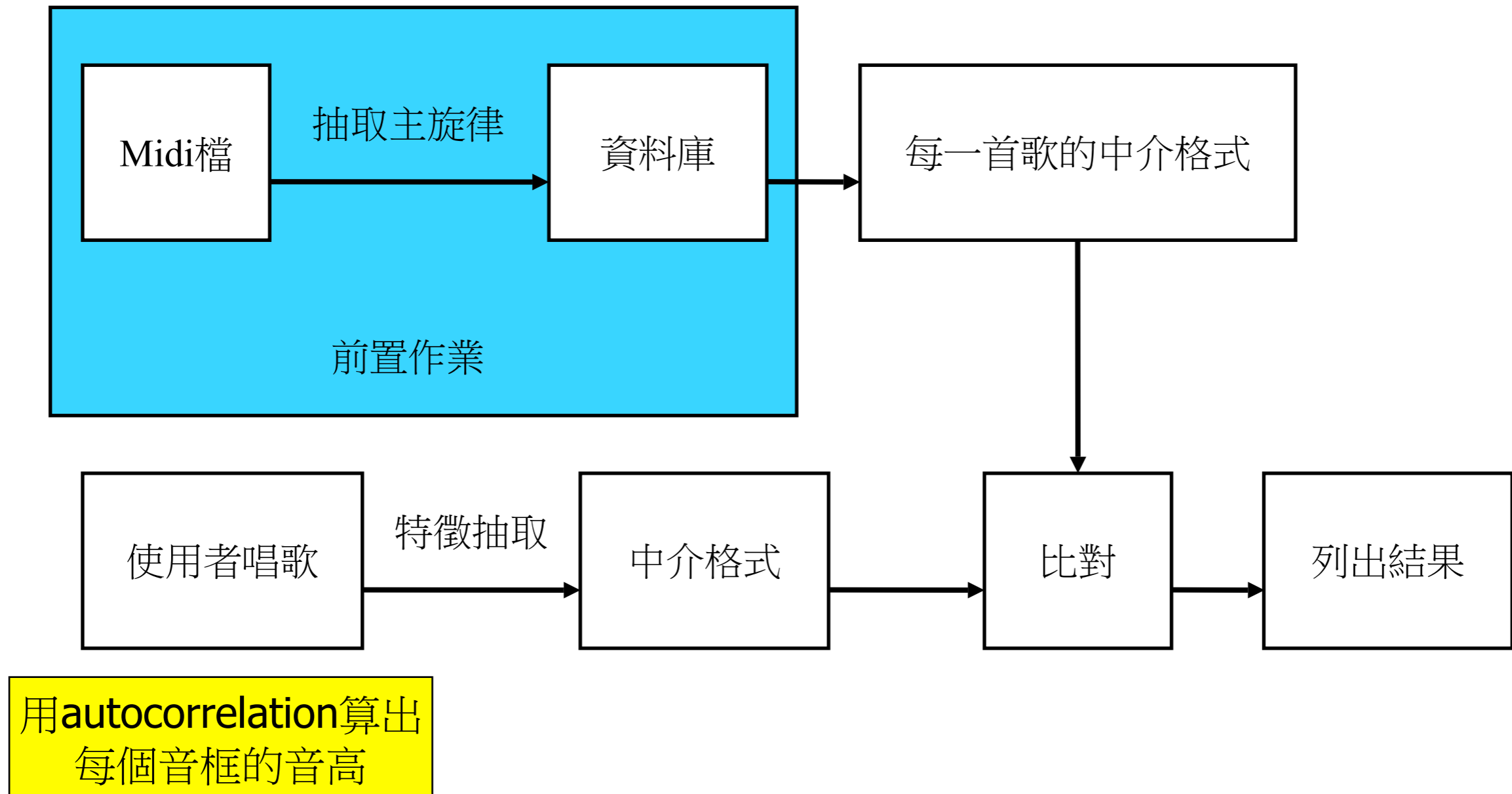
Content based music retrieval

- 說明
 - 用聲音的內容為根據，做音樂的檢索
- 目的
 - 讓使用者可以用自然的方式點選歌曲
- 困難
 - 使用者的節奏（**tempo**）快慢不同、拍子不準、音調（**key**）高低不同
 - 若允許使用者從歌的任意處唱，計算量很大

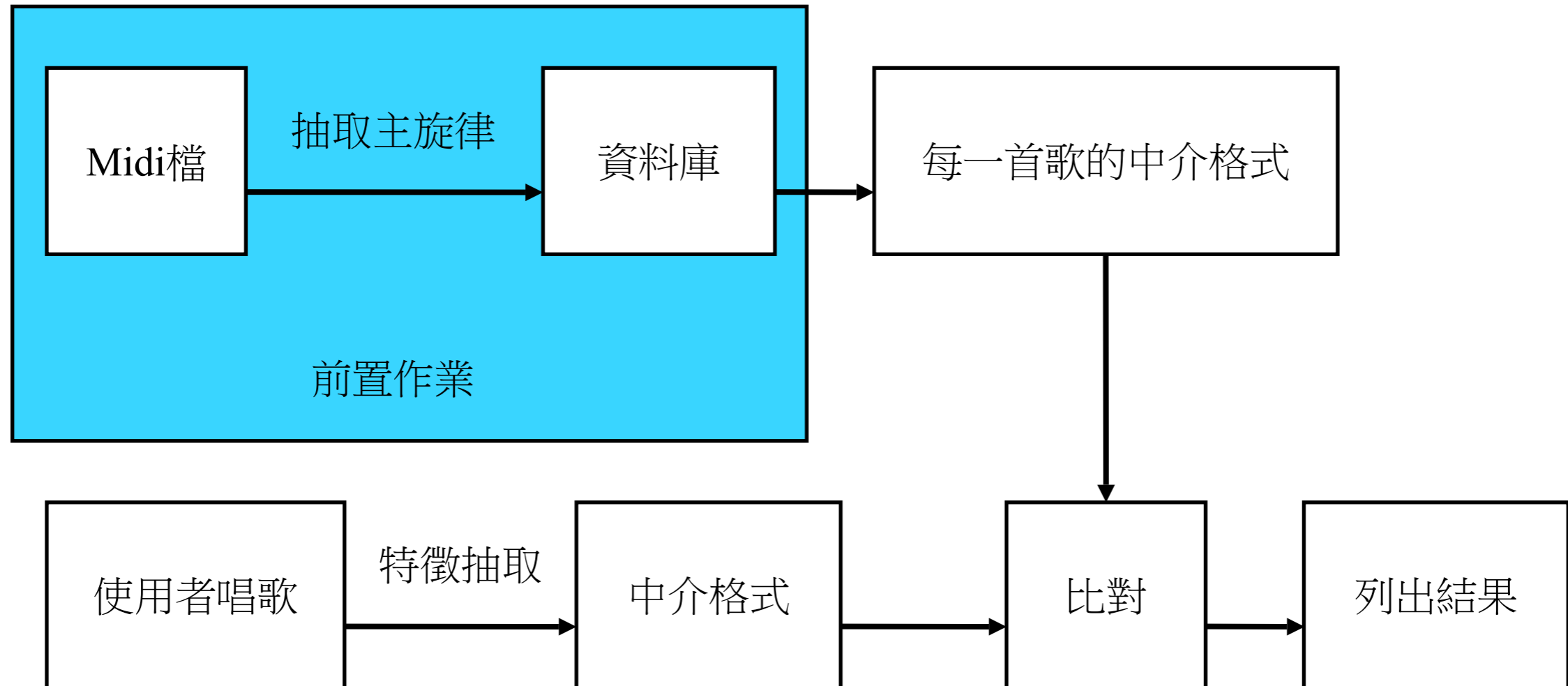
CBMR系統流程圖



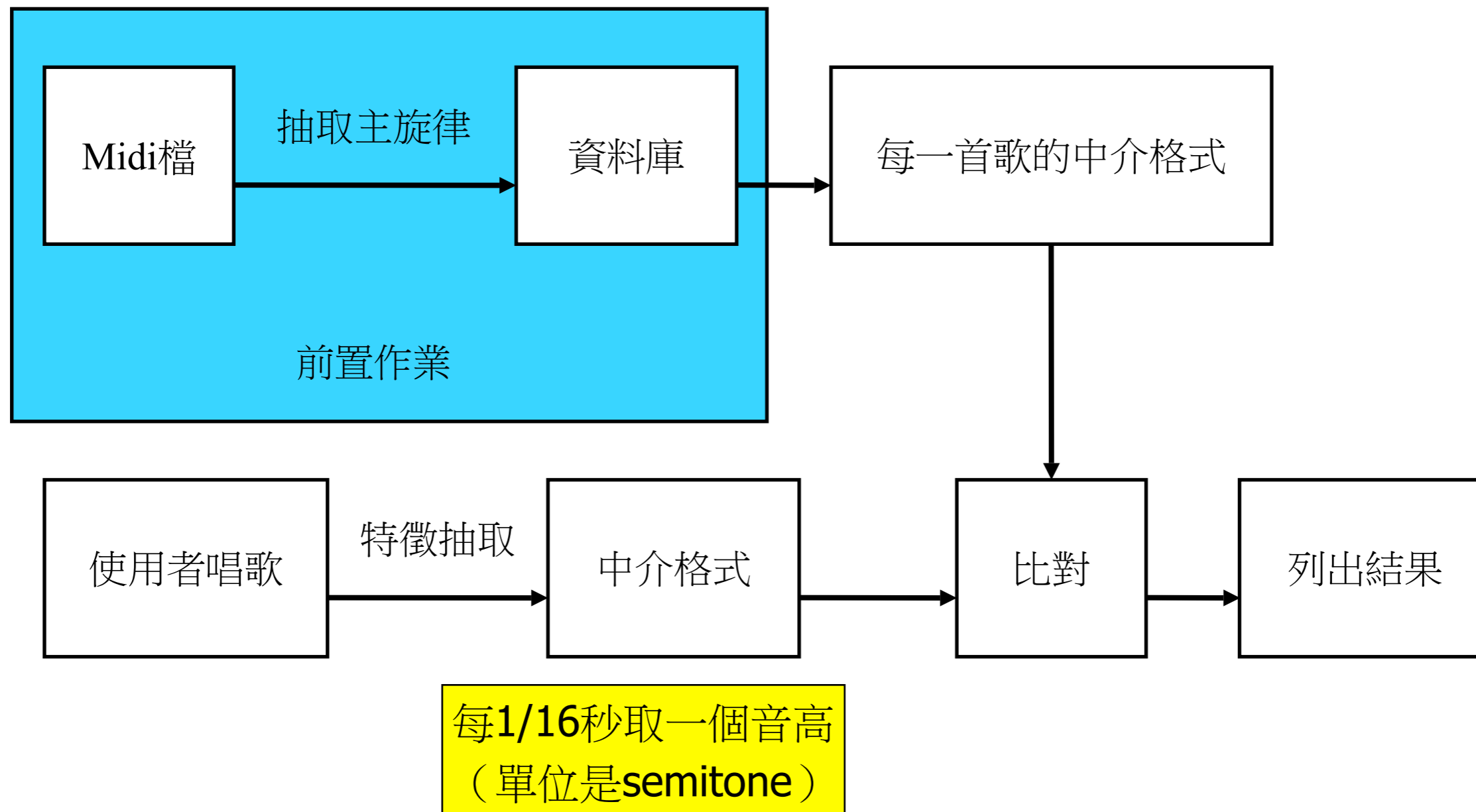
CBMR系統流程圖



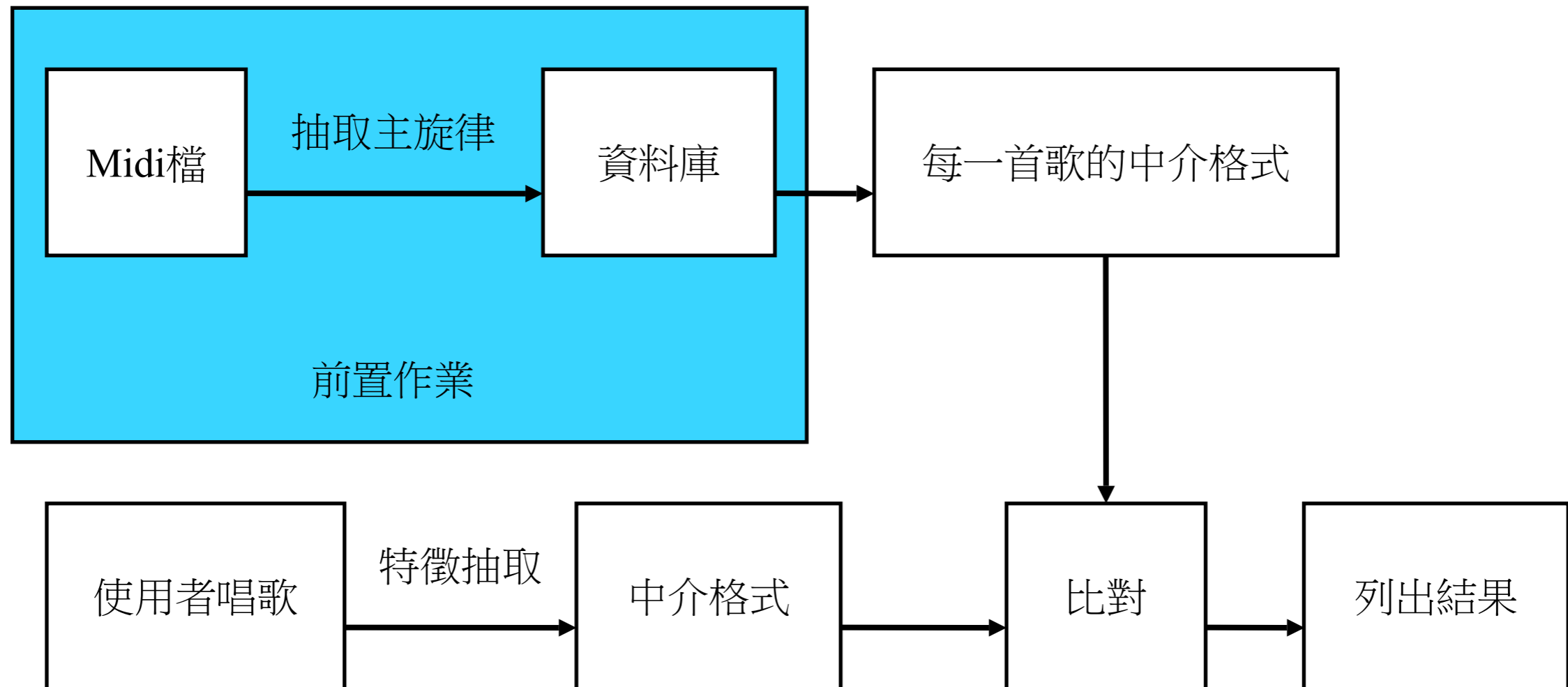
CBMR系統流程圖



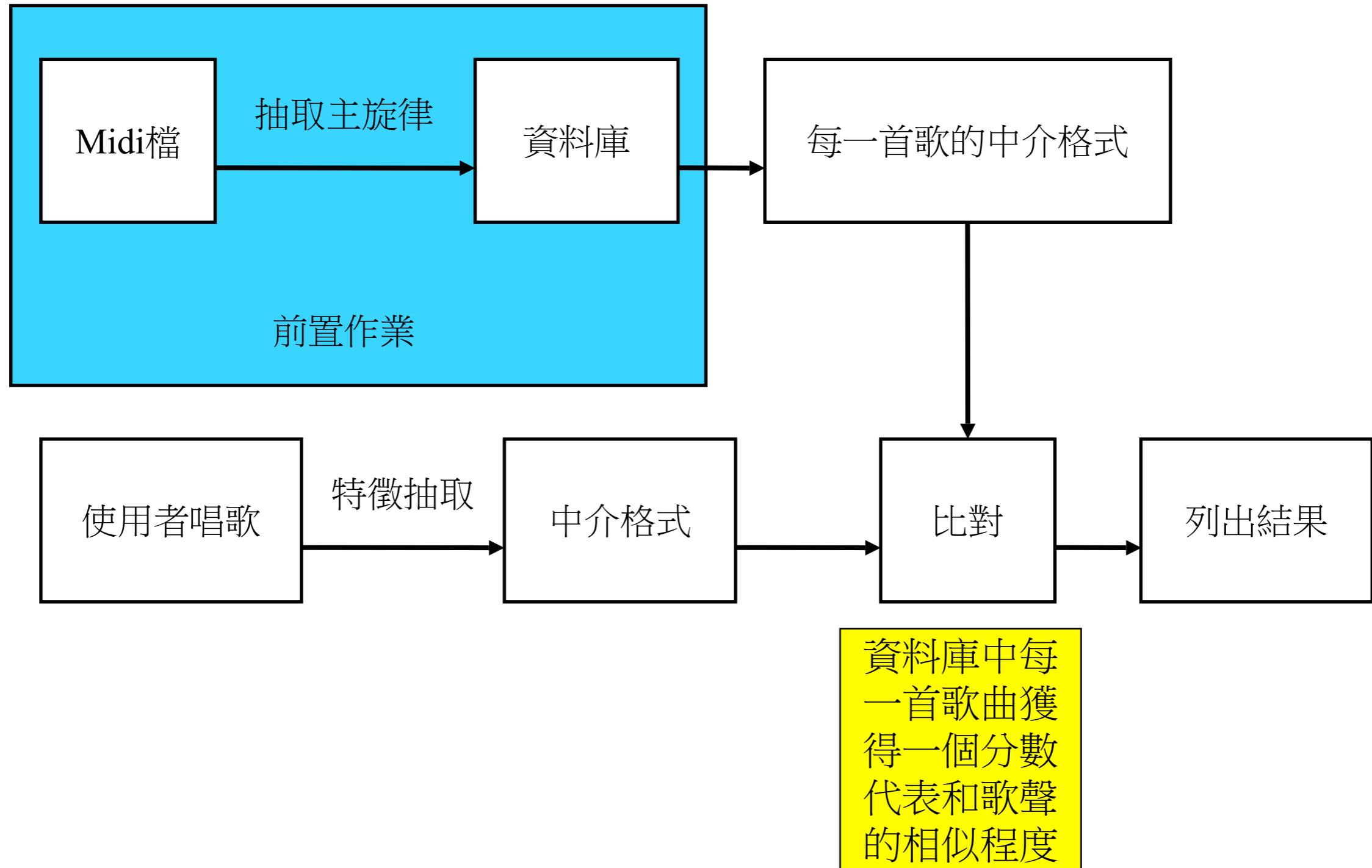
CBMR系統流程圖



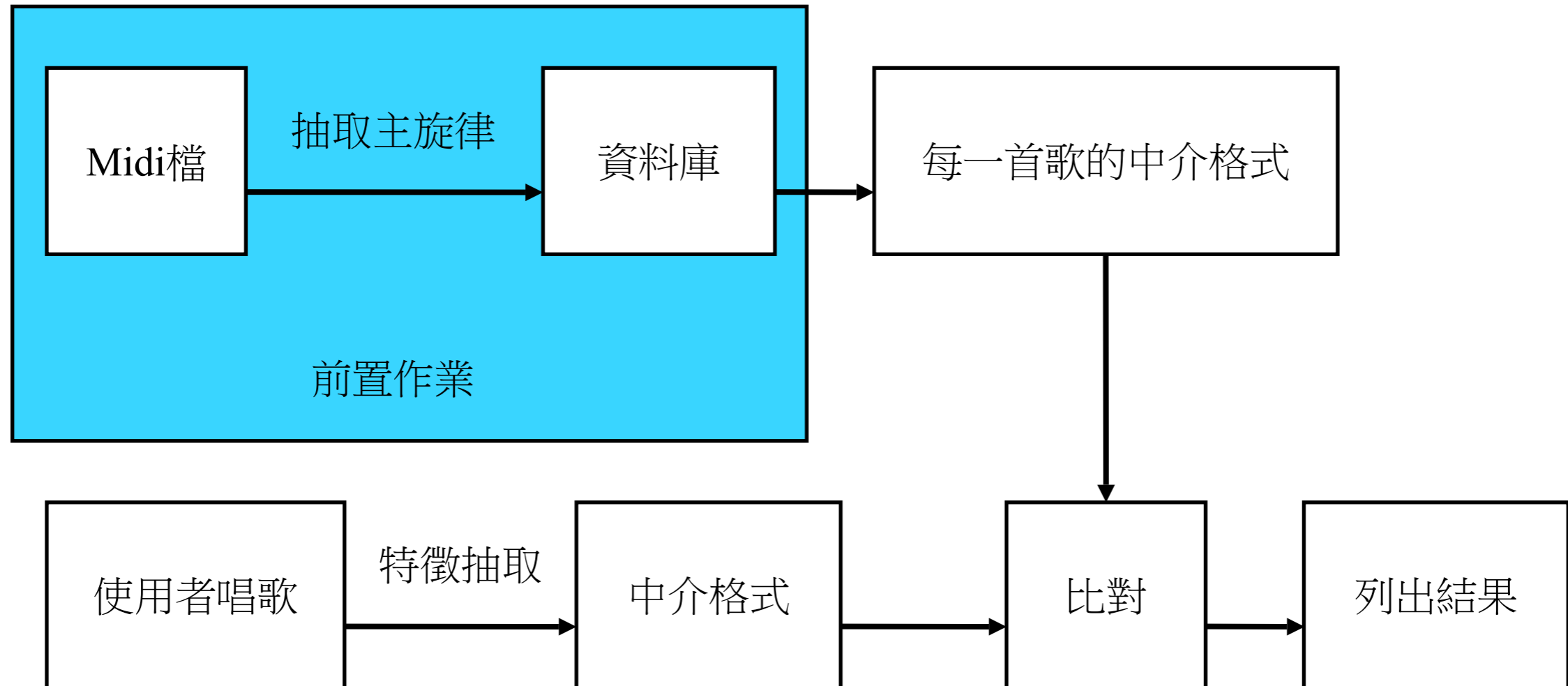
CBMR系統流程圖



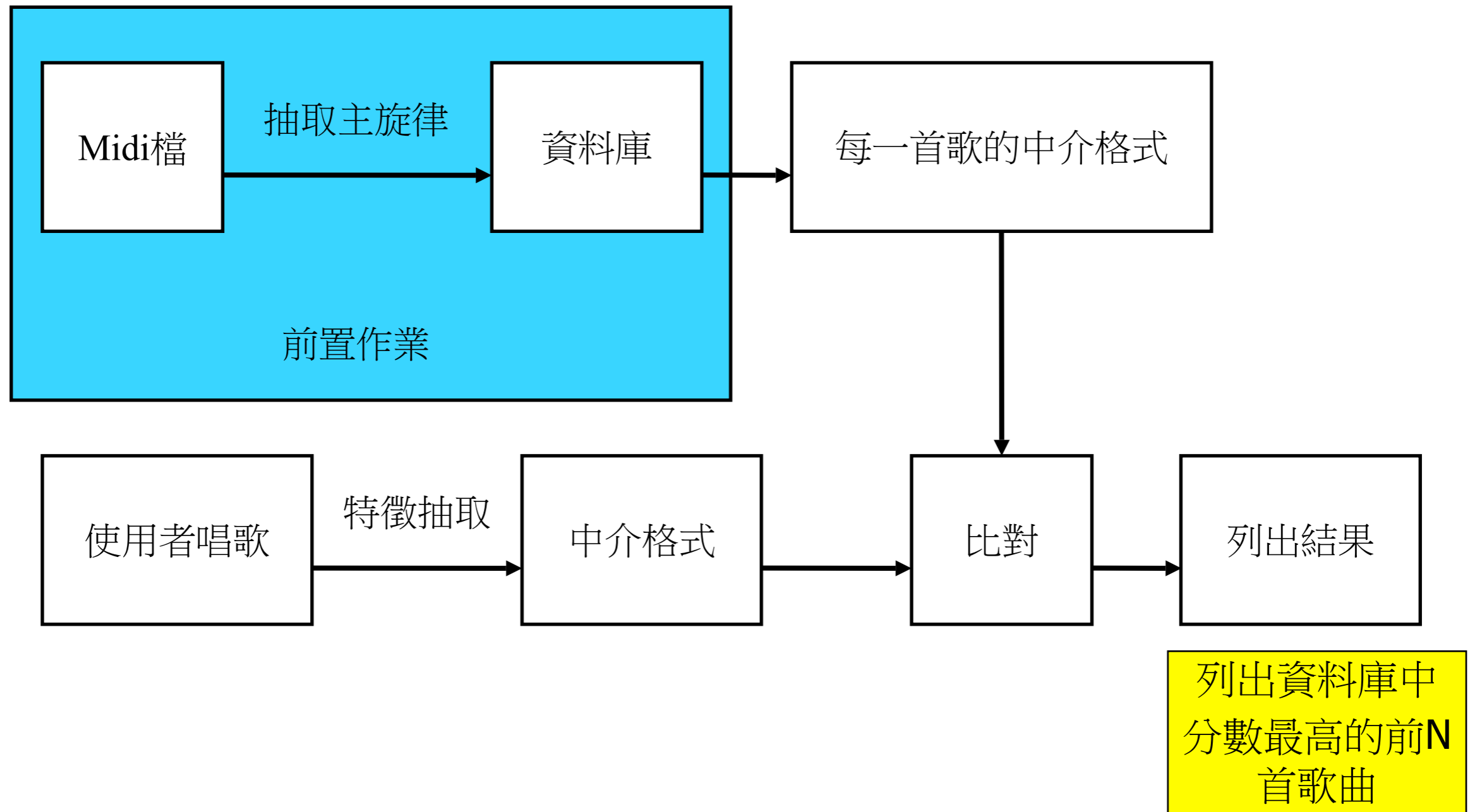
CBMR系統流程圖



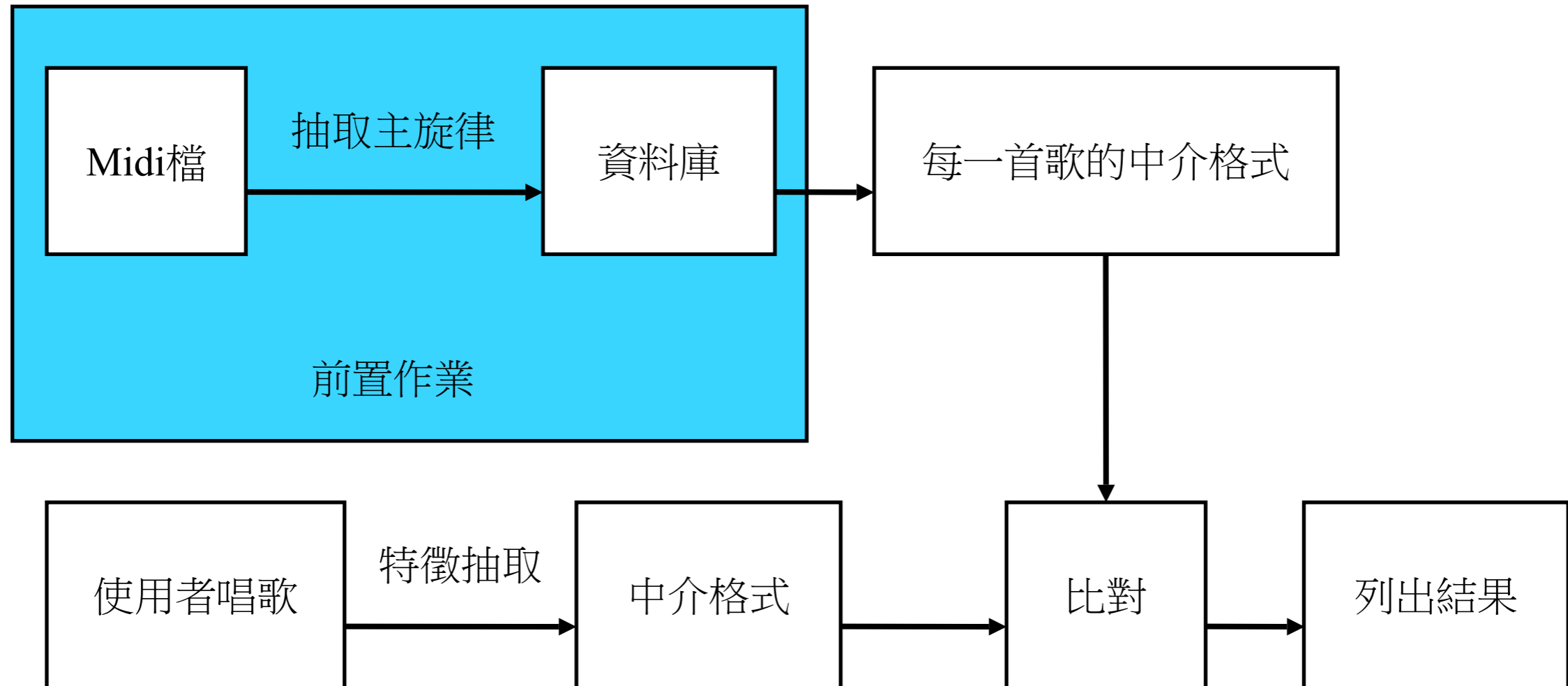
CBMR系統流程圖



CBMR系統流程圖



CBMR系統流程圖



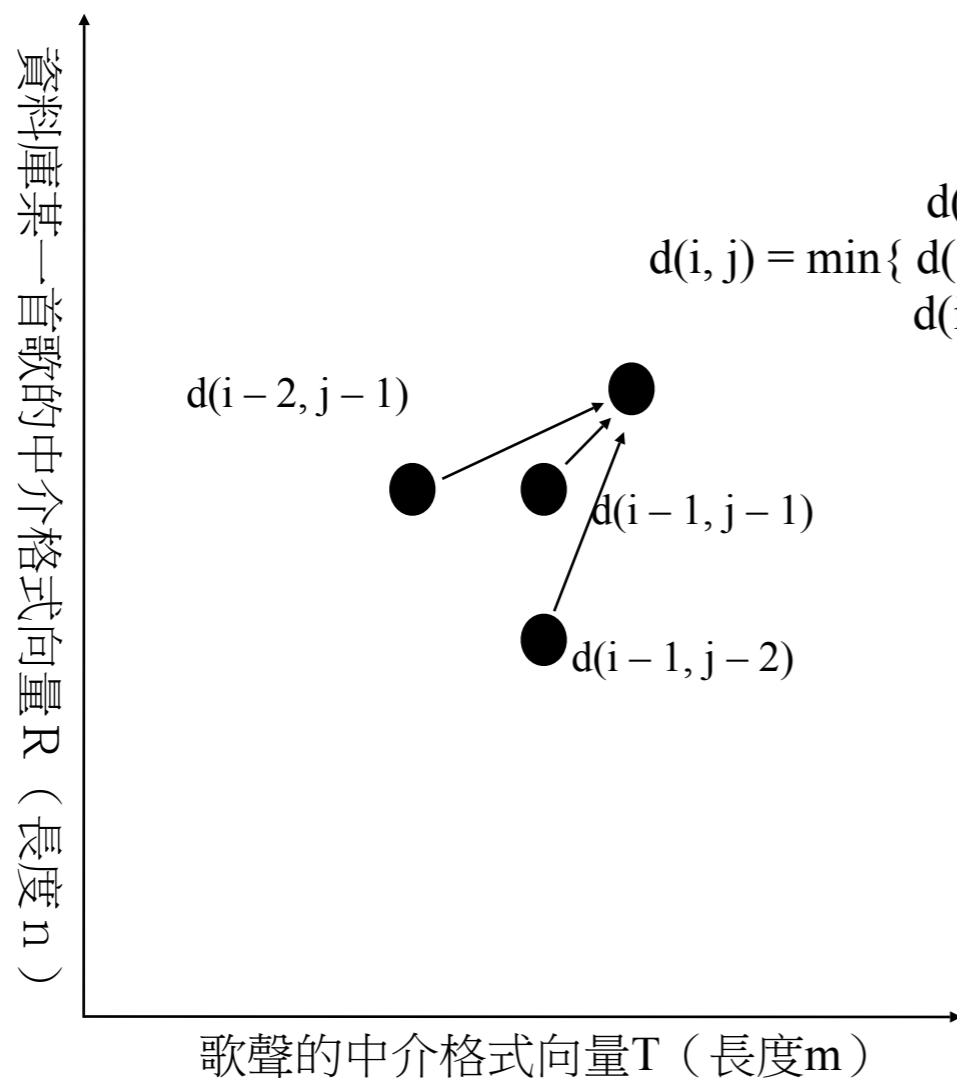
相關研究

名稱	使用限制	比對方法	資料庫歌曲數目
Query by Humming	必須唱 ta 或 da 斷音	Baeza-Yates & Perleberg (92)	183
MELDEX (Melody Indexing)	必須唱 ta 或 da 斷音	Dynamic programming	9400
SoundCompass	必須照節拍器唱	Pitch transitions & histograms for weighted average	10086
MELODISCOV	無	FIEXPAT, Rolland (99)	未知

前人的方法

- 克服節奏快慢不同的問題
- **Dynamic time warping**
- 克服音調高低不同的問題
- **Key transposition**
 - 將使用者的歌聲和資料庫中的歌轉換後的中介格式的平均值都平移到0，做一次**dtw**比對
 - 將資料庫中的歌的中介格式上下平移再做四次**dtw**比對，以找出最短的距離
- 全曲比對費時很久且準確率低
- 使用浮點數運算

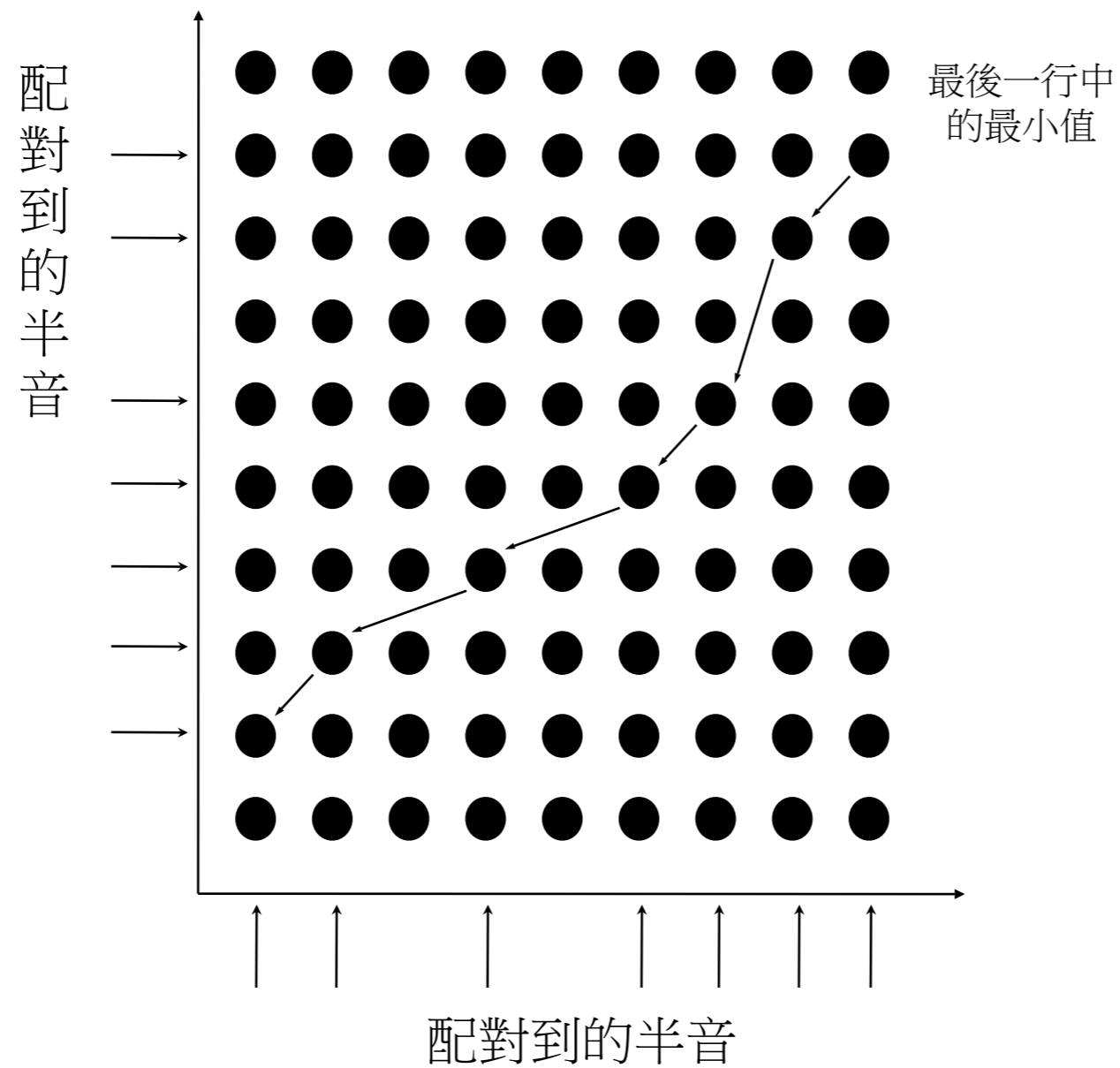
Dynamic Time Warping



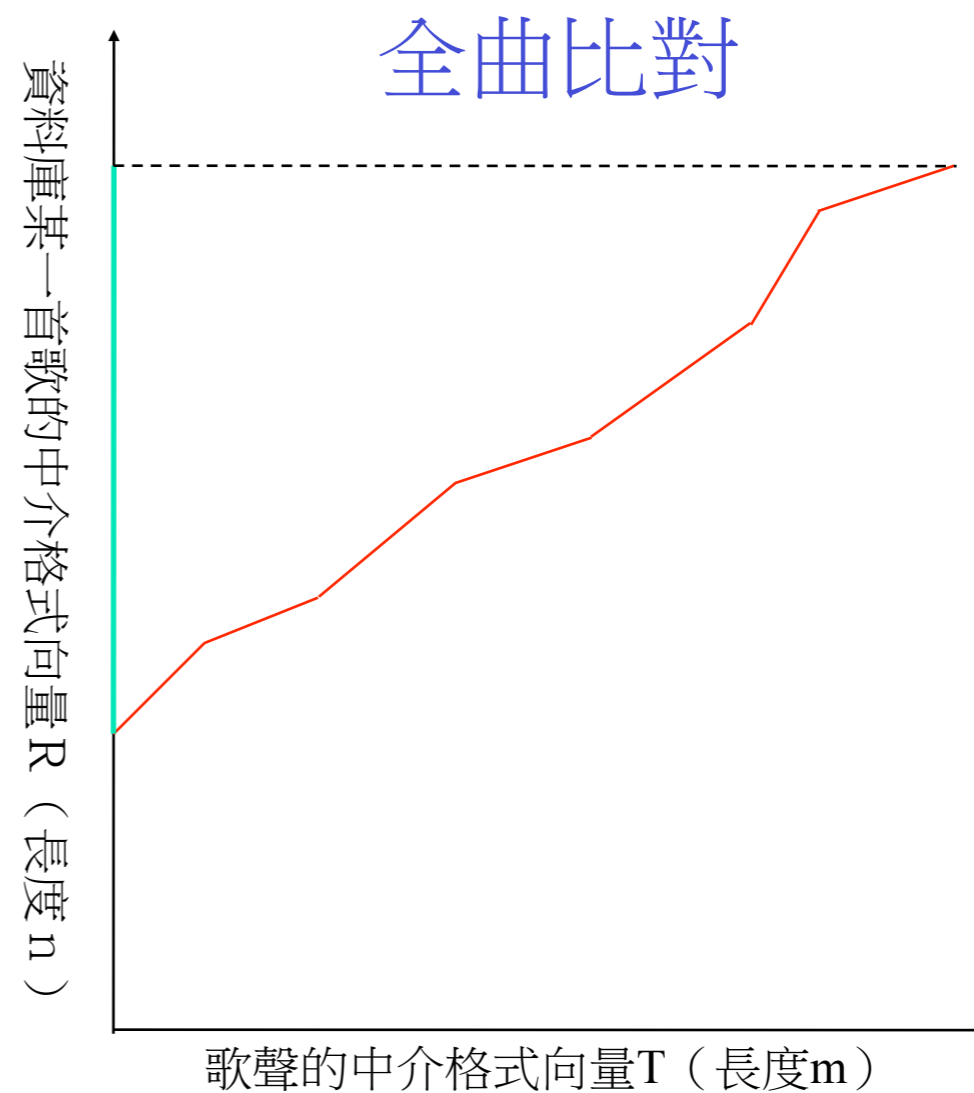
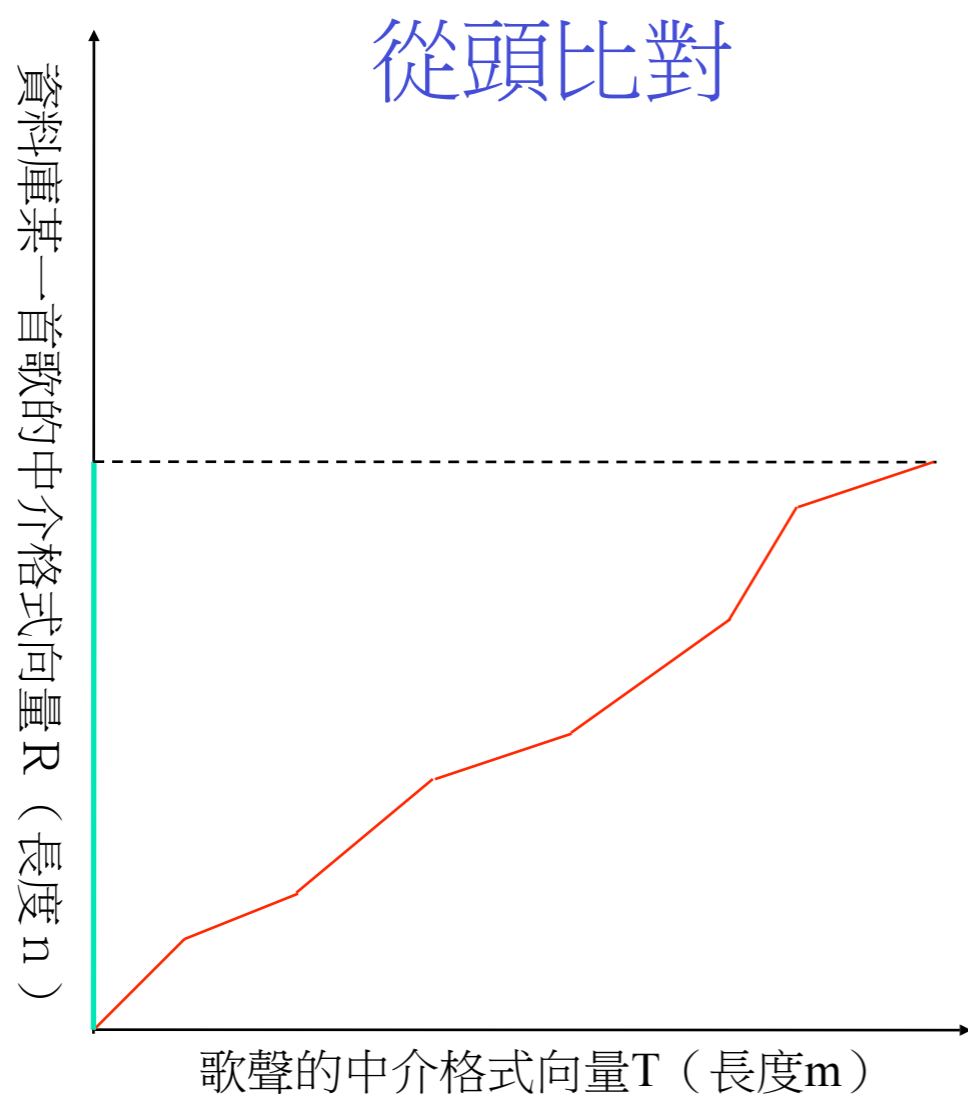
$$d(i, j) = \min \begin{cases} d(i-1, j-1) \\ d(i-2, j-1) \end{cases} + |T(i) - R(j)|$$

$$dtw(T, R) = \min_{1 \leq j \leq n} d(m, j)$$

Dynamic Time Warping

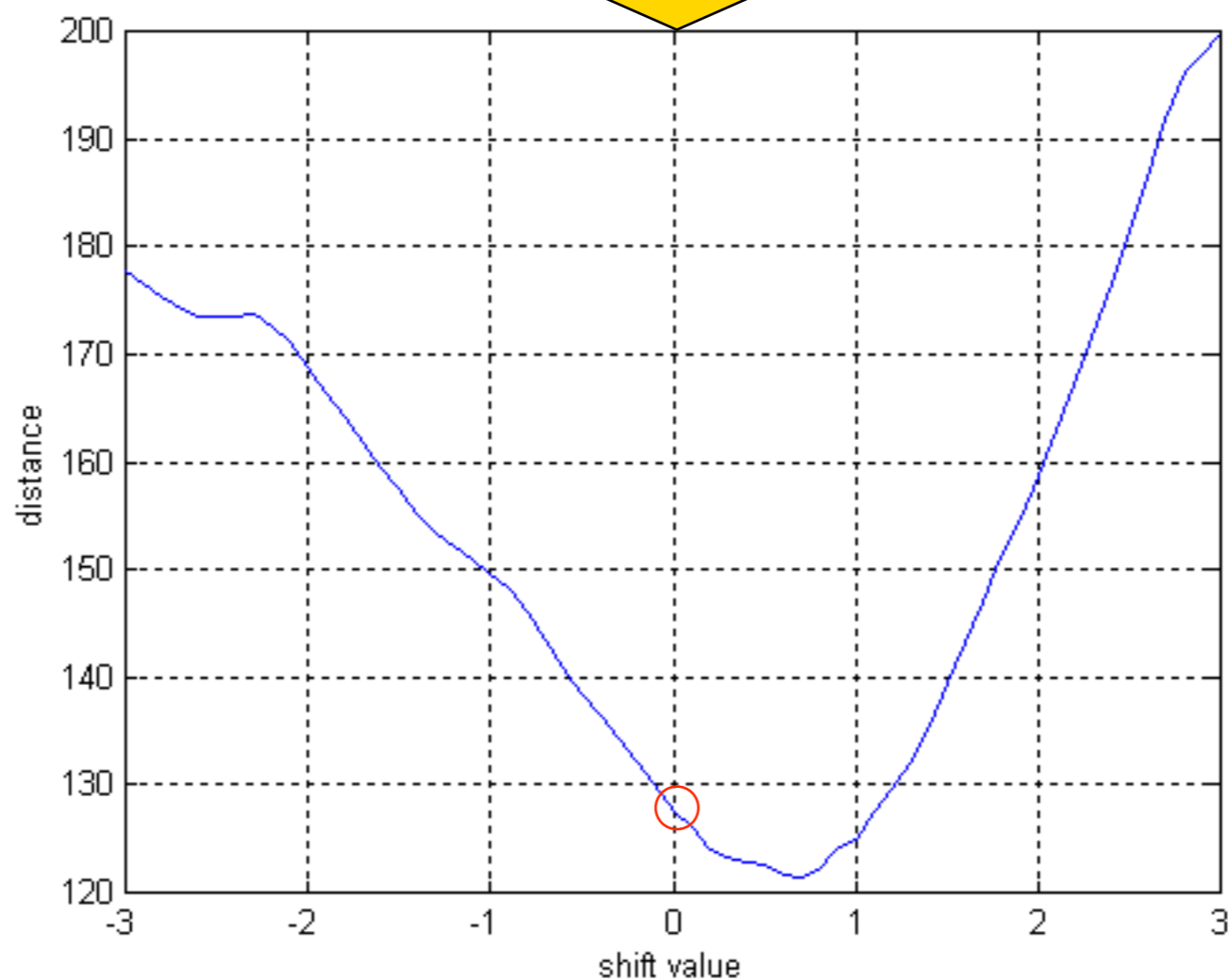


Dynamic Time Warping



Key Transposition

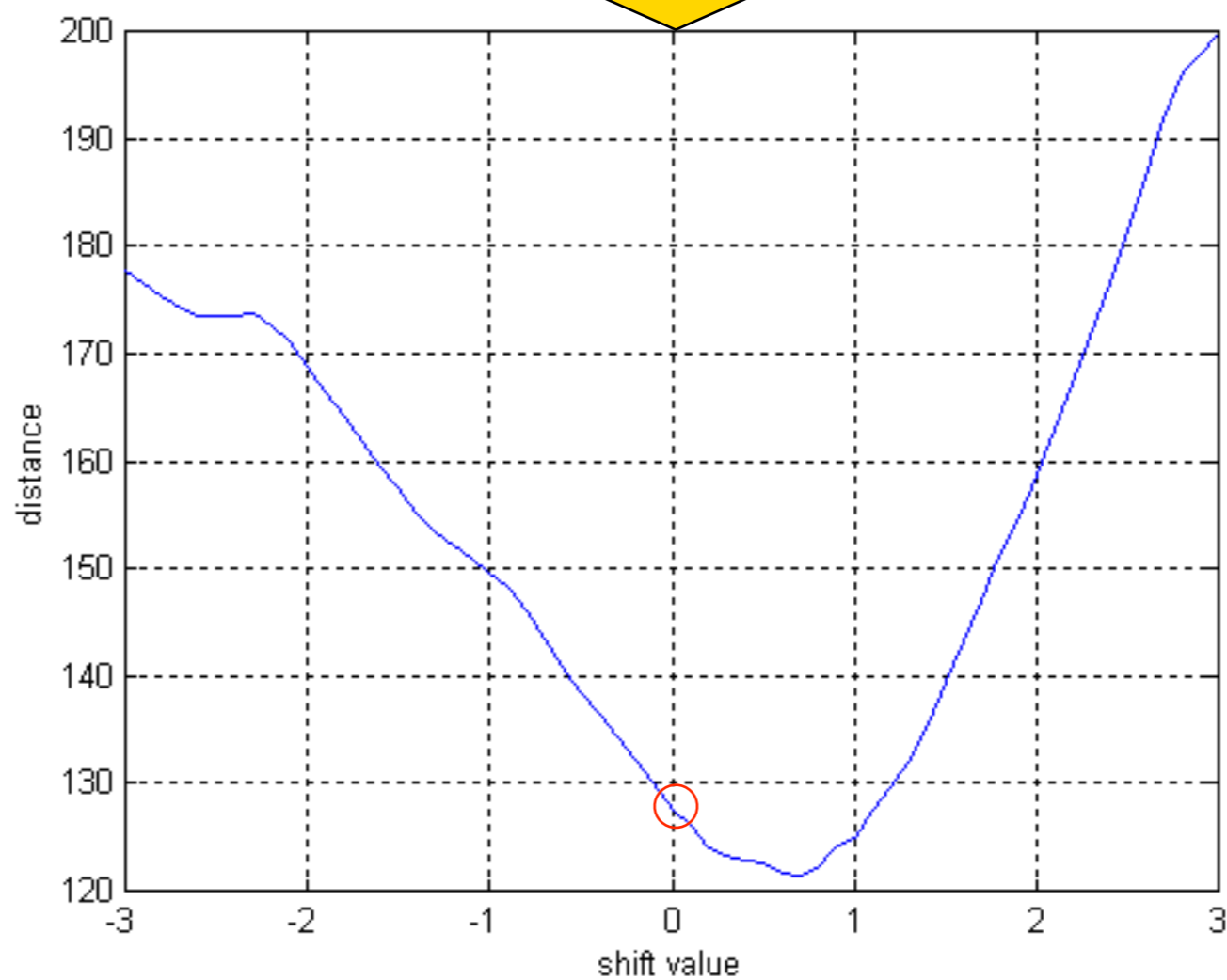
歌聲的中介格式



資料庫中某一首歌的中介格式

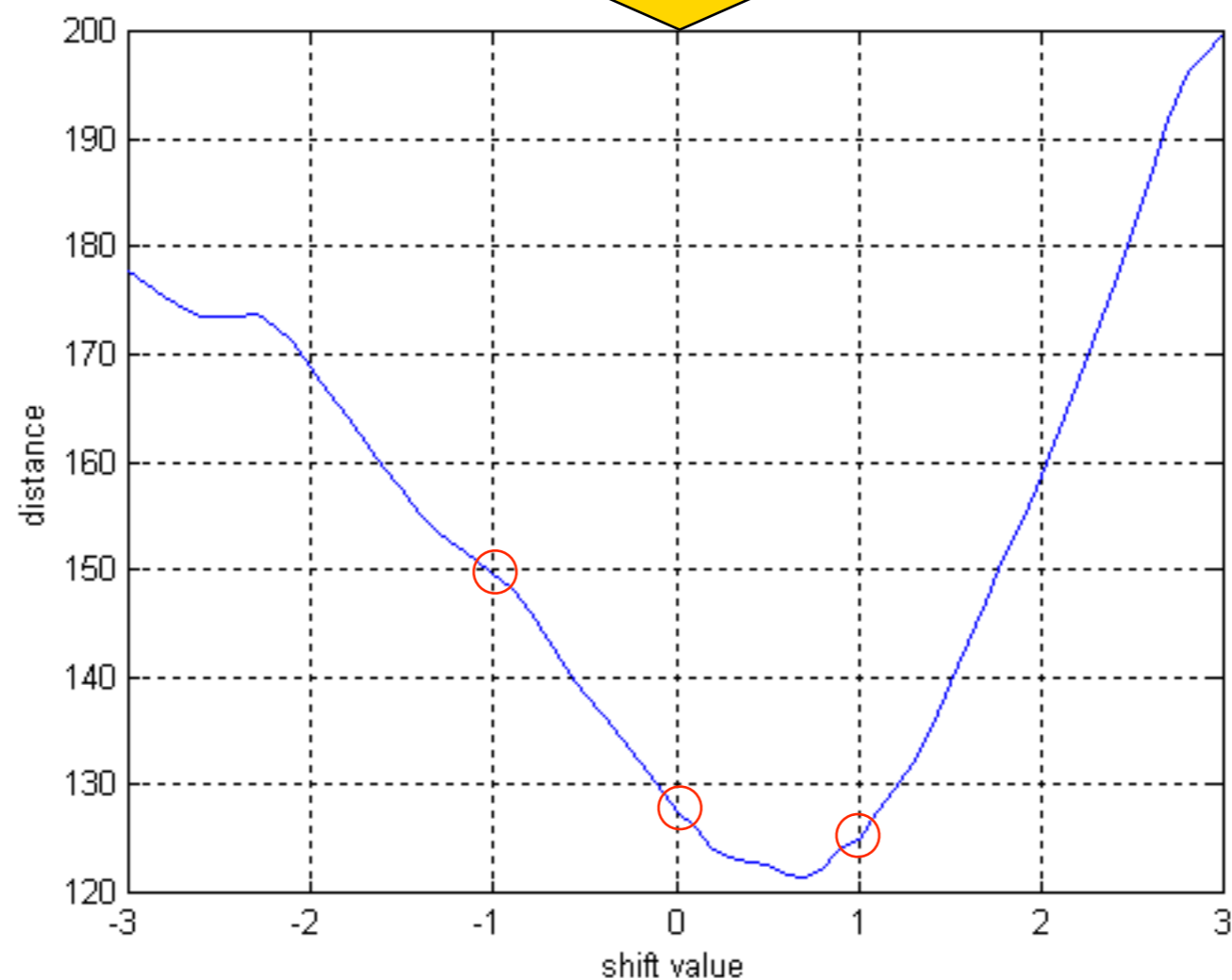
Key Transposition

歌聲的中介格式



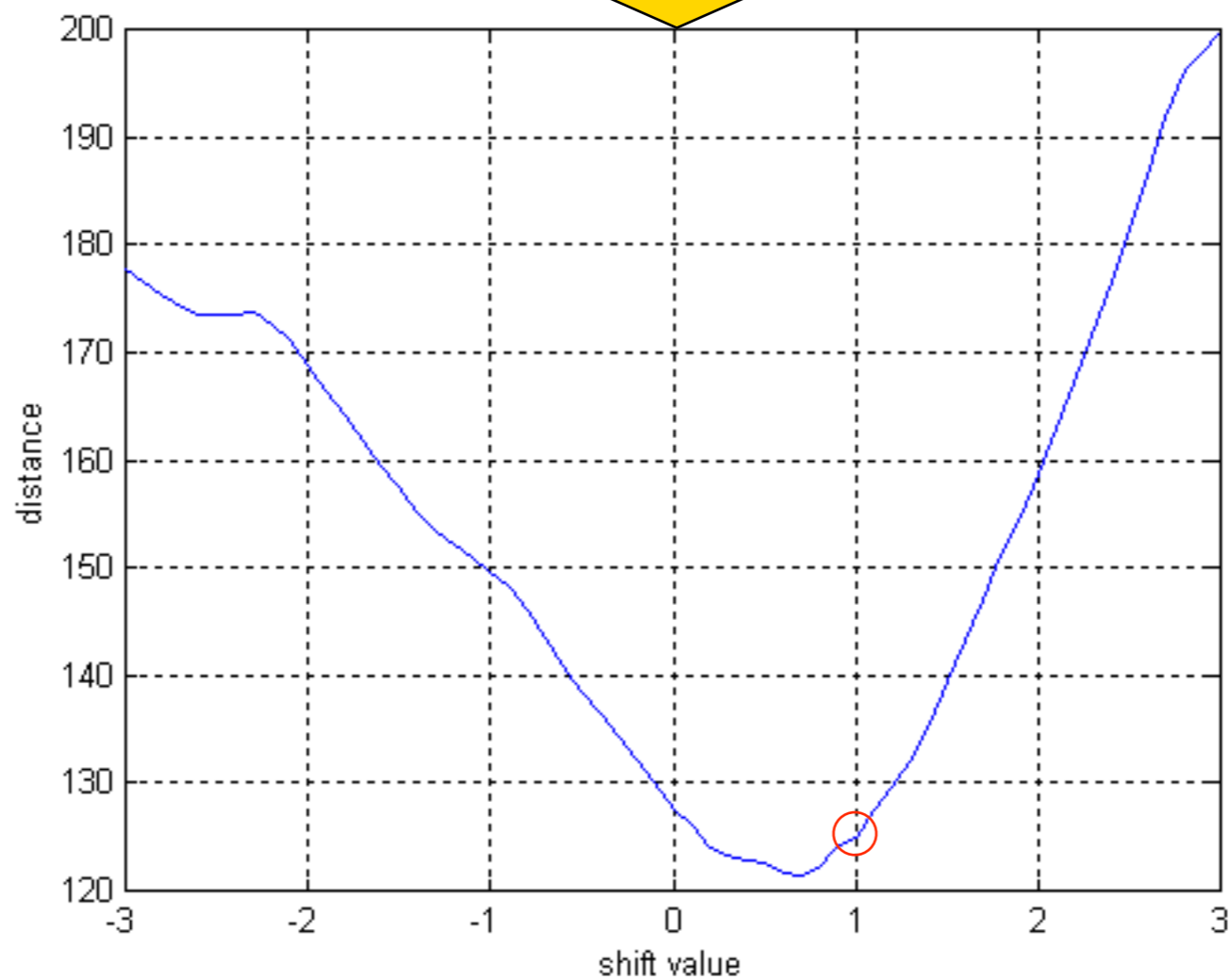
Key Transposition

歌聲的中介格式



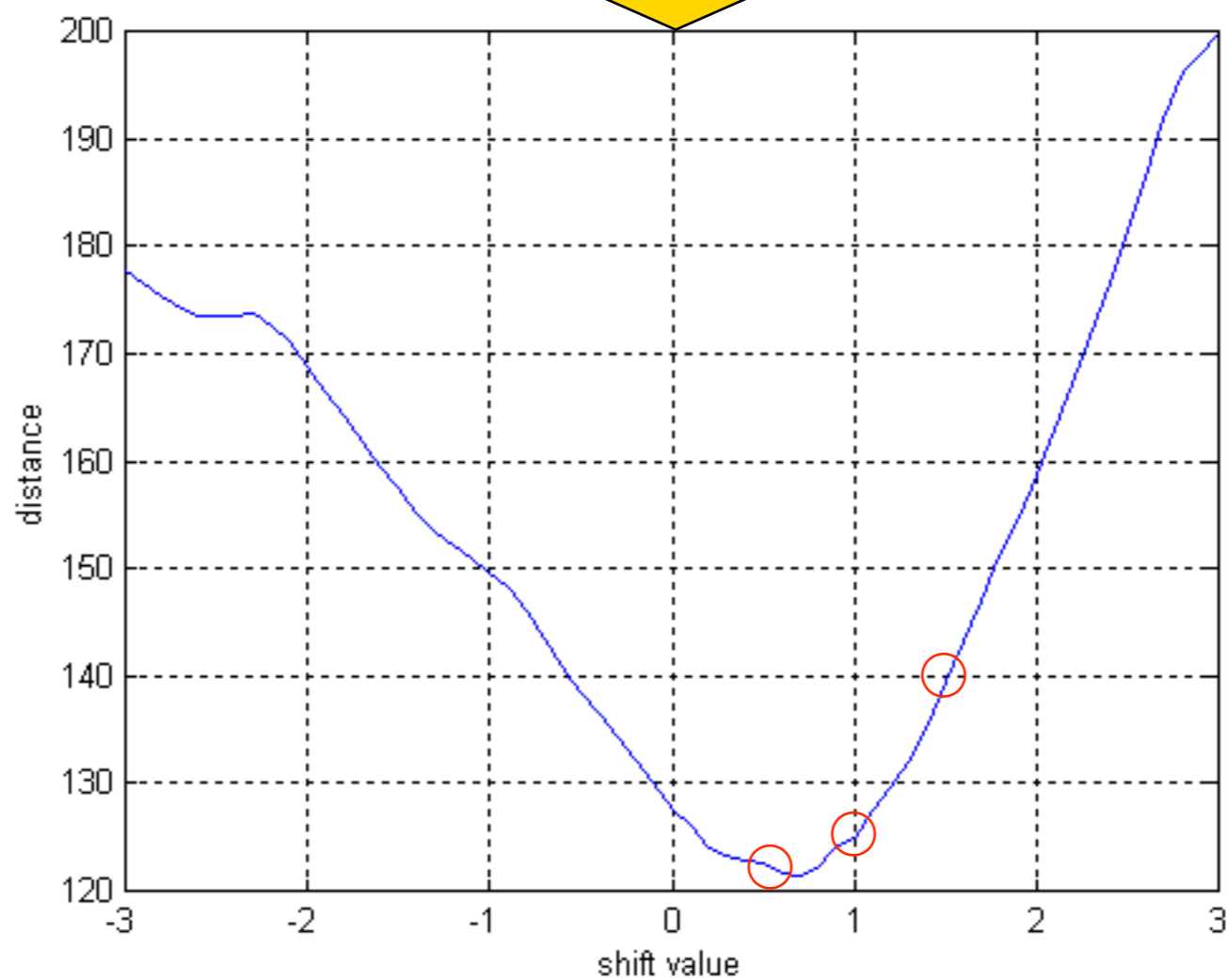
Key Transposition

歌聲的中介格式



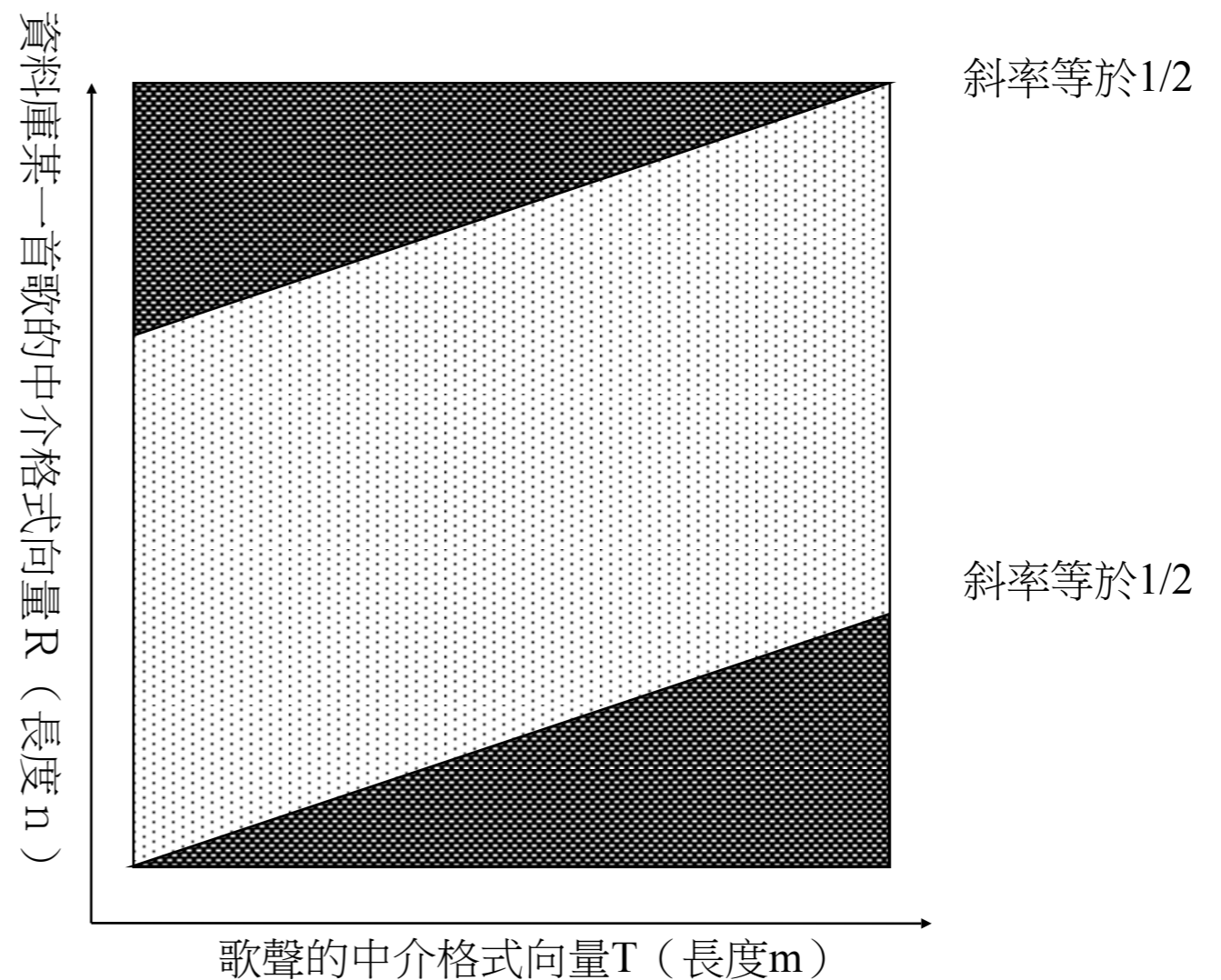
Key Transposition

歌聲的中介格式



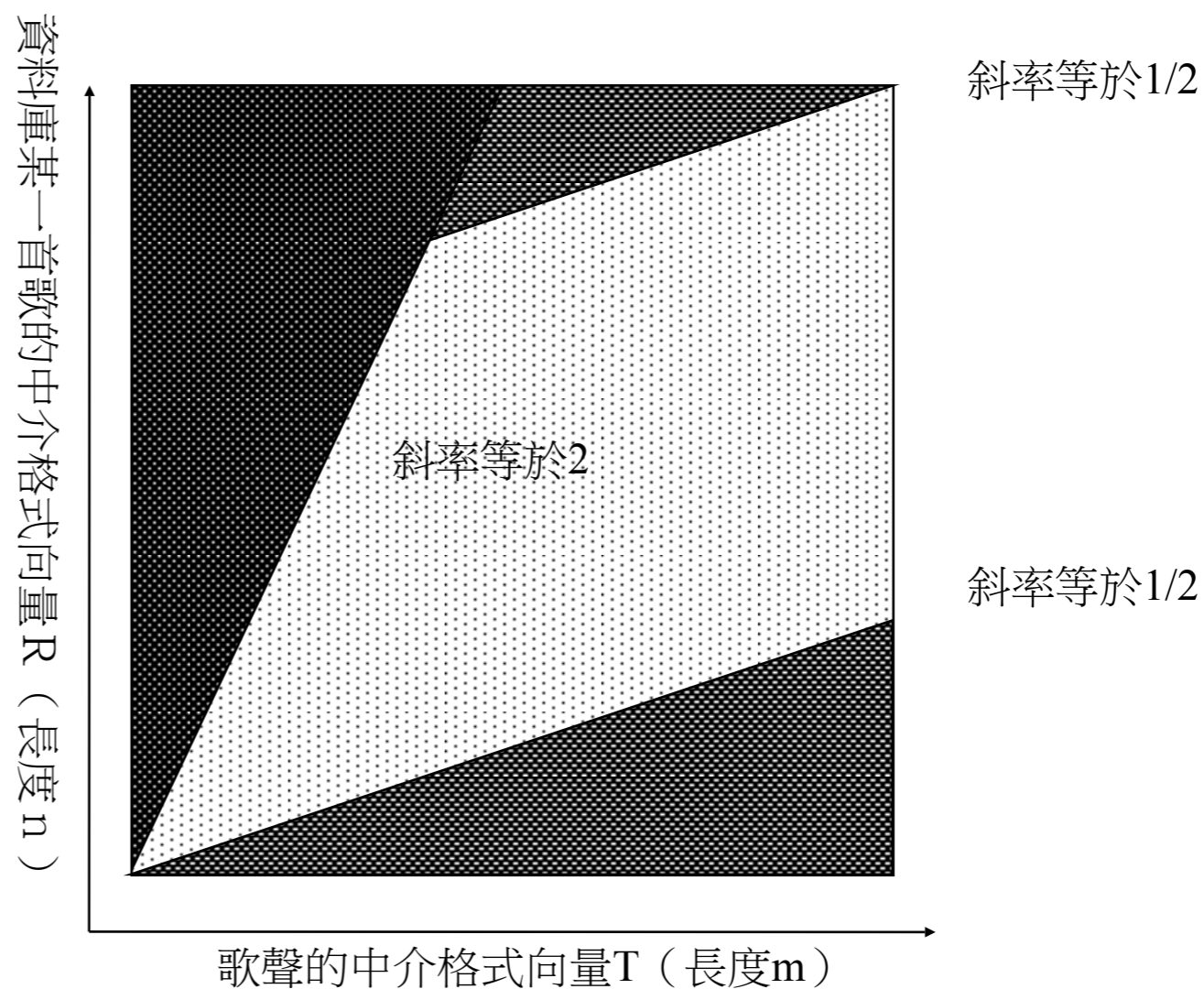
改進方法一

- 改用整數運算
 - 犧牲一點點準確率
- 改良式dtw
 - 因為T的頭尾必須match到R的某一段，所以顏色較深的地方不必計算



改進方法一

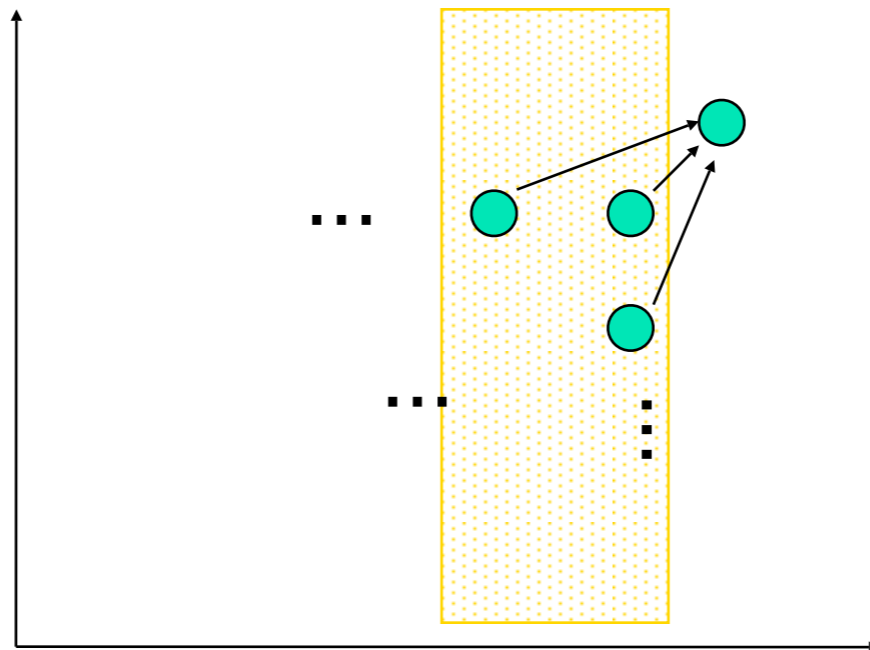
- 改用整數運算
 - 犧牲一點點準確率
- 改良式dtw
 - 因為T的頭尾必須match到R的某一段，所以顏色較深的地方不必計算



改進方法一

- 改良式dtw

- 用dtw計算每一首歌的距離時，若發現dtw table最近兩行(column)的最小值超過之前最短距離的前k名，則停止dtw





改進方法二

- 將資料庫中每一首歌的中介格式，從每一個音符為起點切成數個長度為 **D=72** 的片段

中介格式 (69, 69, 67, 67, 67, 71, 72,)

→ 片段1 (69, 69, 67, 67, 67, 71, 72, ...)

片段2 (67, 67, 67, 71, 72, ...)

片段3 (71, 72, ...)

.....



改進方法二

- 用兩階段的方法比對

- 第一階段：線性伸縮比對（**linear scaling**）

- 將歌聲的中介格式伸縮**11**次（長度為原來的**0.75**倍到**1.25**倍），分別取出前 **D** 點後為 T_i ($1 \leq i \leq 11$)，假設資料庫中的第 **j** 首歌的中介格式有 n_j 個片段為 R_{jk} ($1 \leq j \leq$ 資料庫中的歌曲數目, $1 \leq k \leq n_j$)，令 T_i 和 R_{jk} 的距離為

$$\text{dist}(T_i, R_{jk}) = \sqrt{\sum_{t=1}^D \left((T_{it} - \bar{T}_i) - (R_{jkt} - \bar{R}_{jk}) \right)^2}$$



改進方法二

- 令資料庫第 j 首歌的分數為 $100 - \min_{\substack{1 \leq i \leq 11 \\ 1 \leq k \leq n_j}} \frac{\text{dist}(T_i, R_{jk})}{D}$ ，令

$$k' = \arg \min_{\substack{1 \leq i \leq 11 \\ 1 \leq k \leq n_j}} \text{dist}(T_i, R_{jk})$$

- 篩選出前 $n=200$ 首分數最高者做第二階段的比對
- 缺點：每和資料庫中第 j 首歌做比對就要計算 $n_j * 11$ 次距離
- 第二階段：dynamic time warping
 - 將篩選後的每一首歌的最接近片段平移音調4次，總共和歌聲原本的中介格式計算5次距離，找出最小值並轉換成分數，當成資料庫該首歌最後的分數



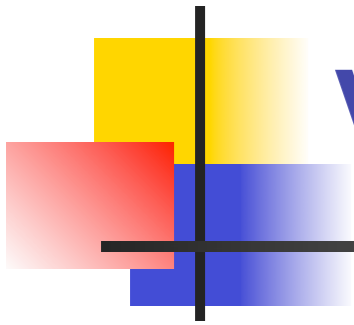
針對全曲比對的加速方法

- 用兩階段式比對
- 在第一階段將資料庫歌曲的每一個片段看成一個個 D 度空間中的資料點，歌聲伸縮後的中介格式則是空間中的查詢點，利用快速找最近鄰居的方法找出每首歌最接近的片段
 - Vantage-point tree
 - Branch-and-bound tree
 - Equal-average hyperplane partitioning method

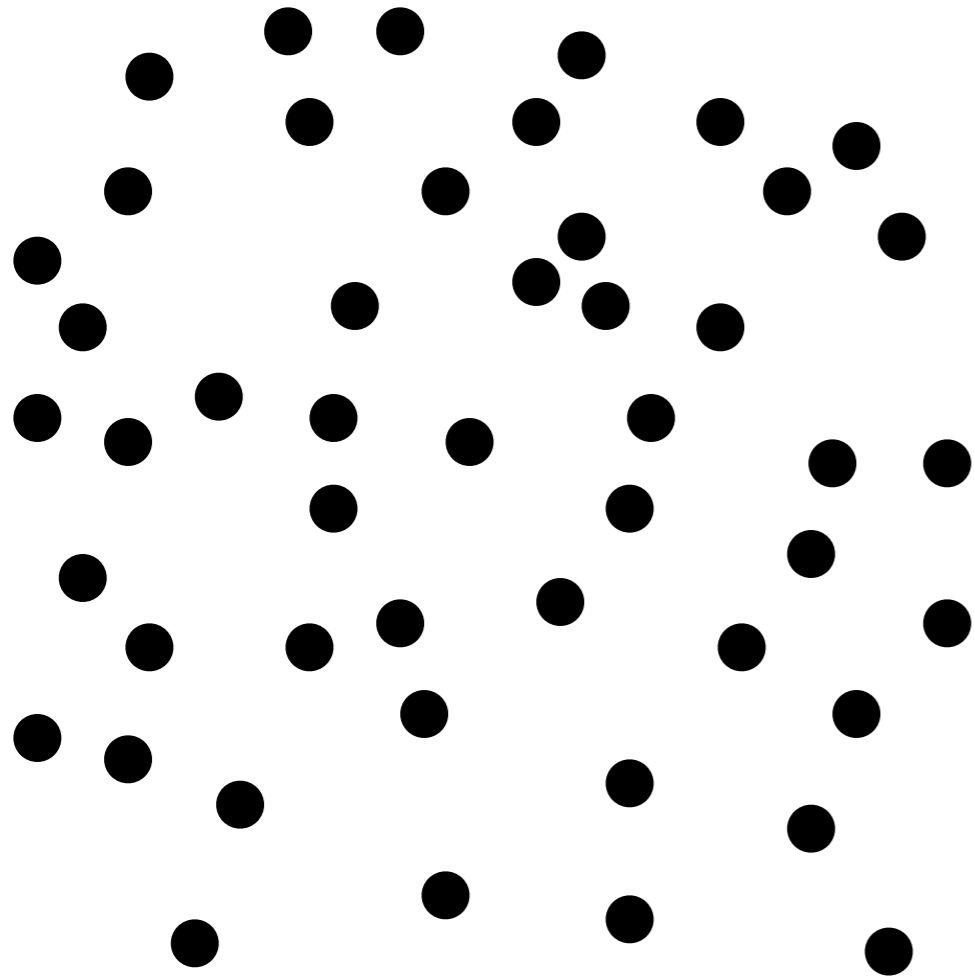


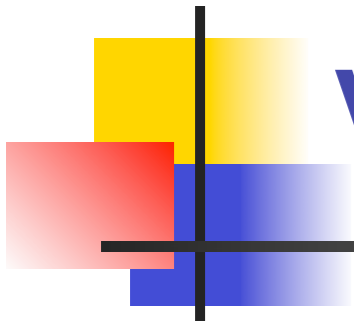
Vantage-Point Tree

- 將資料點建立成一個平衡的二元樹，利用距離滿足三角不等式的關係，可以減少計算查詢點到資料點的距離的次數
 - 缺點：建樹麻煩，使用時耗記憶體
 - 困難：資料點少，查詢點到所有資料點的距離都差不多，加速效果不顯著

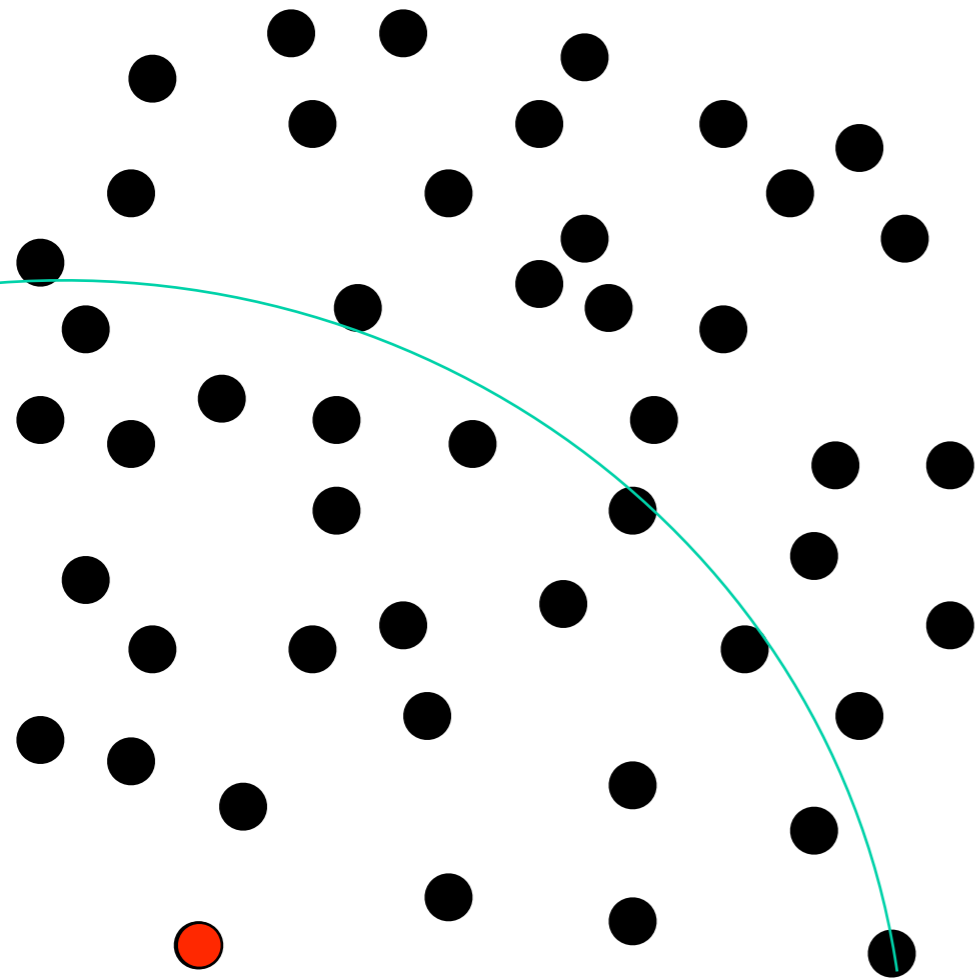


Vantage-Point Tree

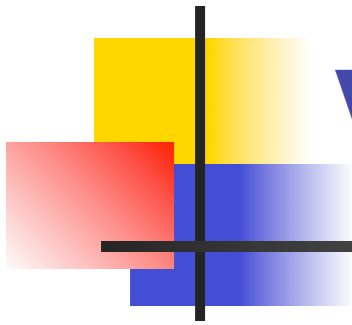




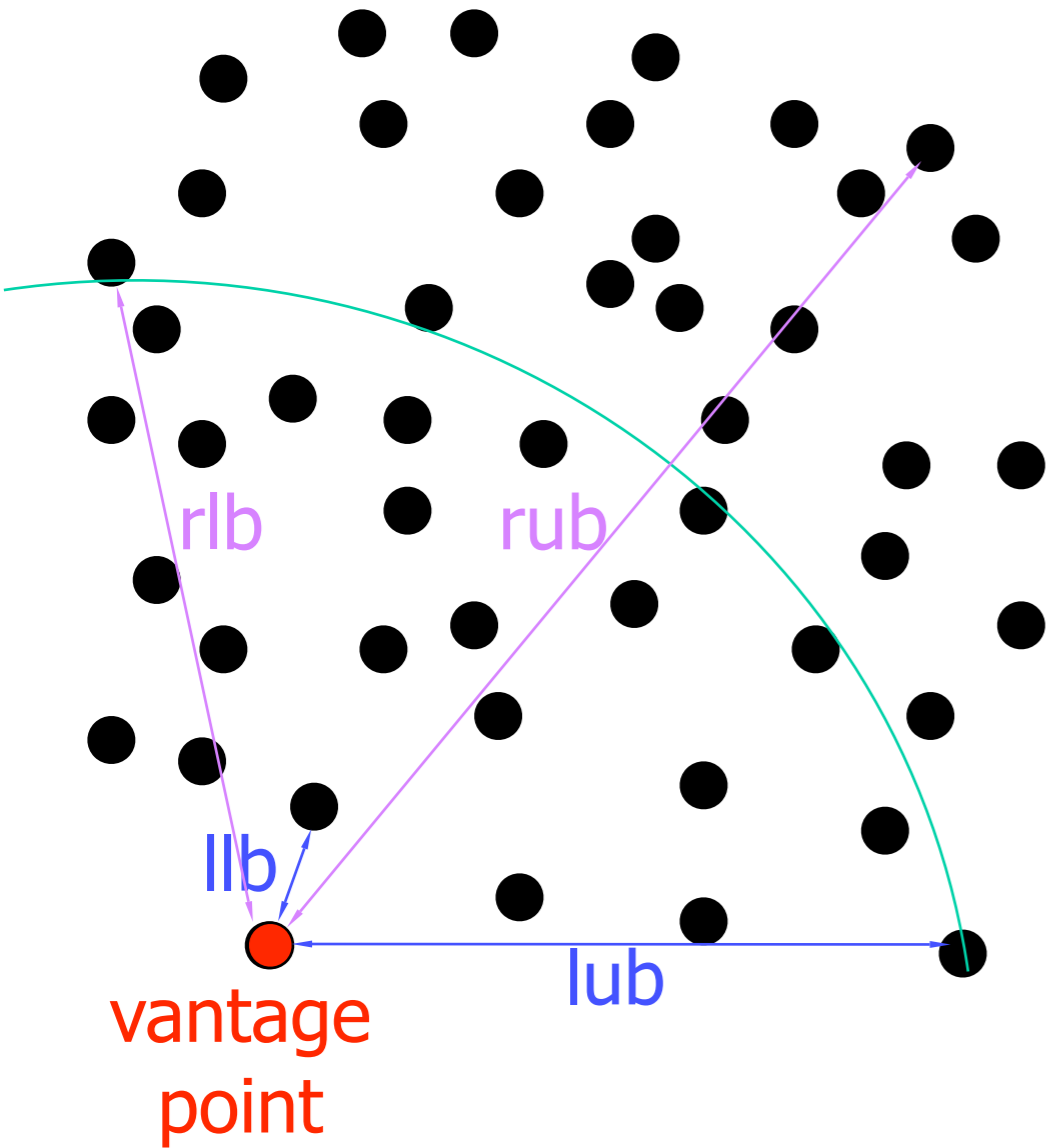
Vantage-Point Tree



vantage
point

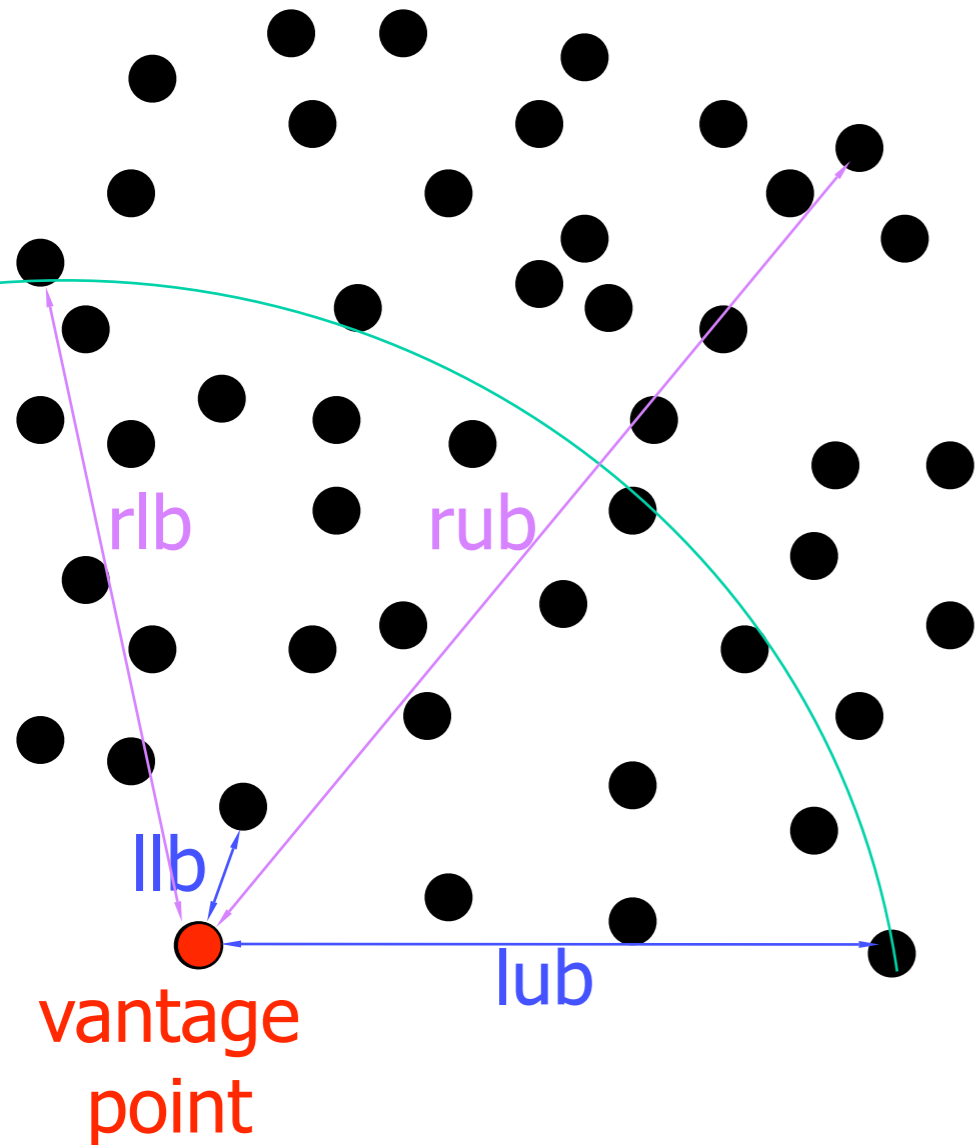
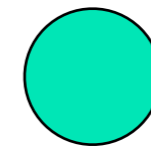


Vantage-Point Tree



Vantage-Point Tree

Vantage point
Left lower bound
Left upper bound
Right lower bound
Right upper bound



Vantage-Point Tree

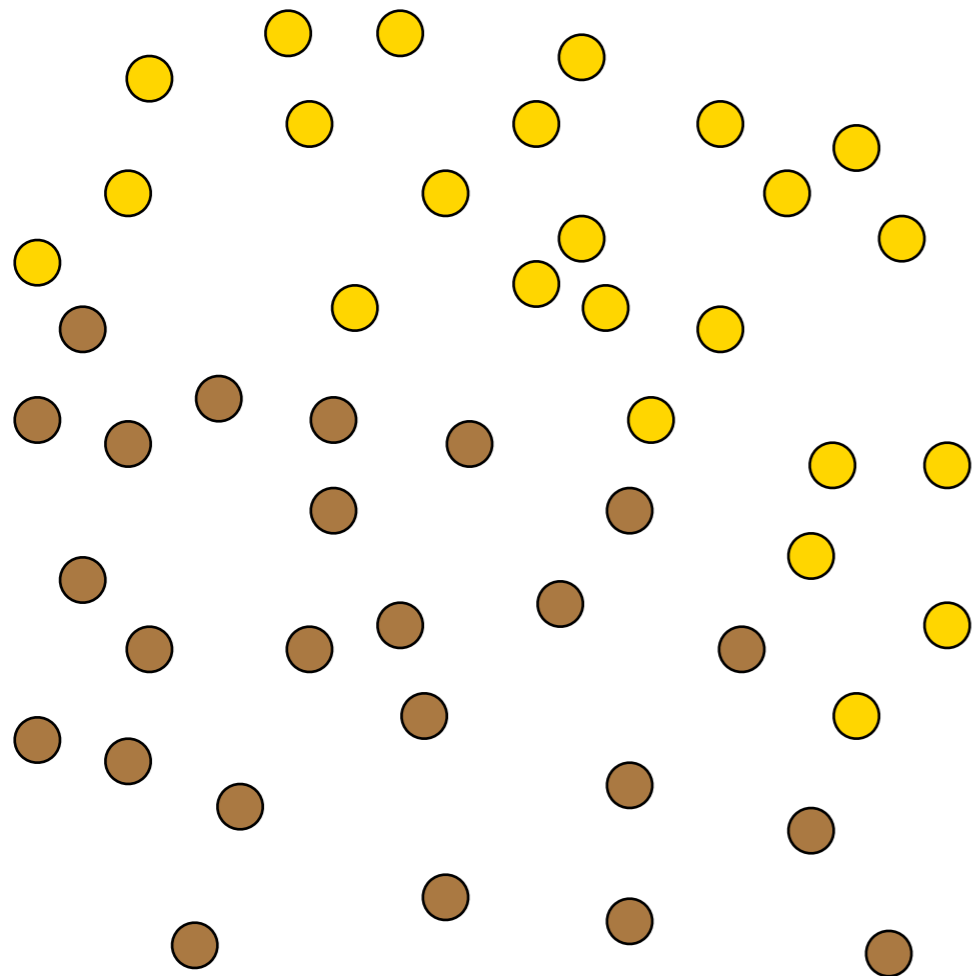
Vantage point

Left lower bound

Left upper bound

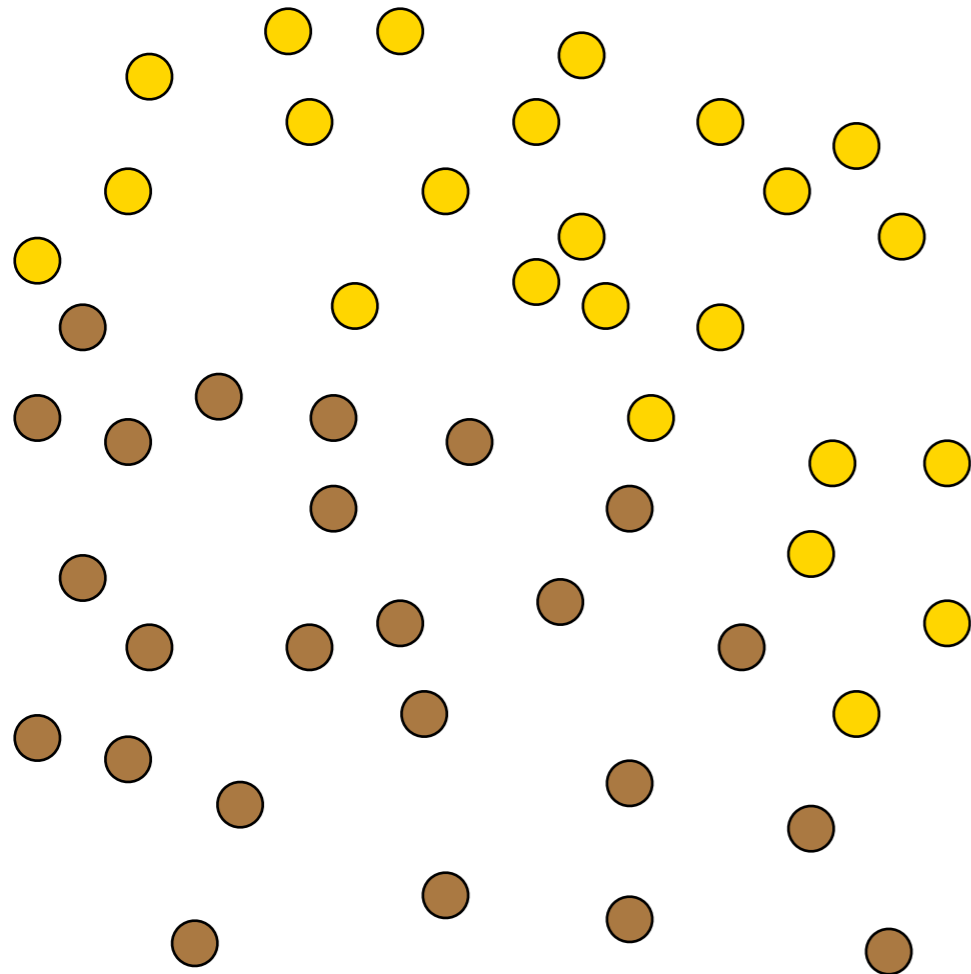
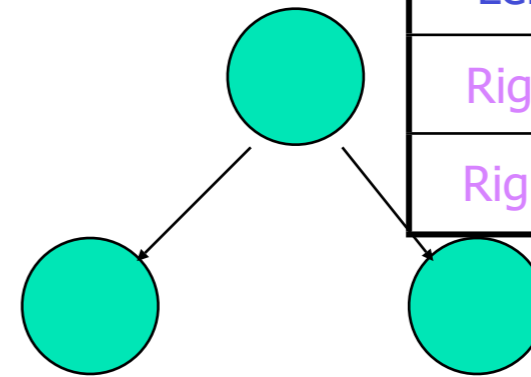
Right lower bound

Right upper bound



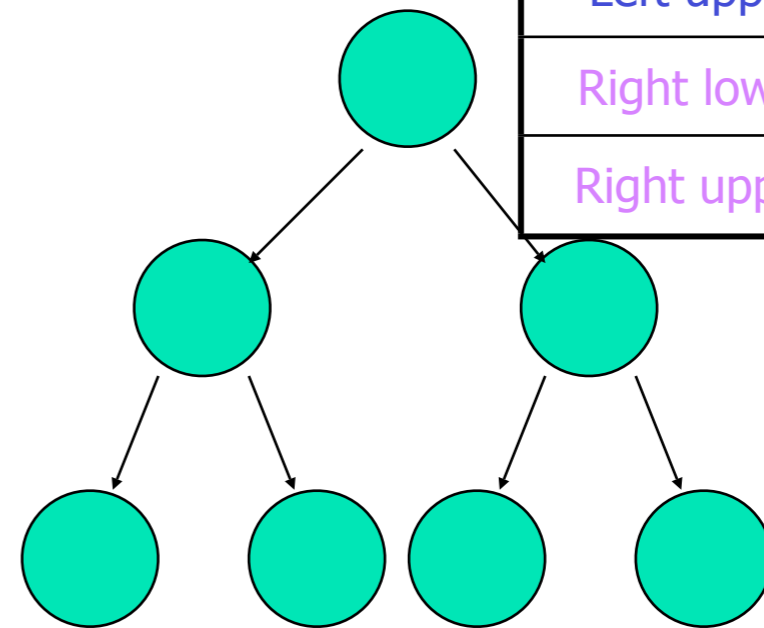
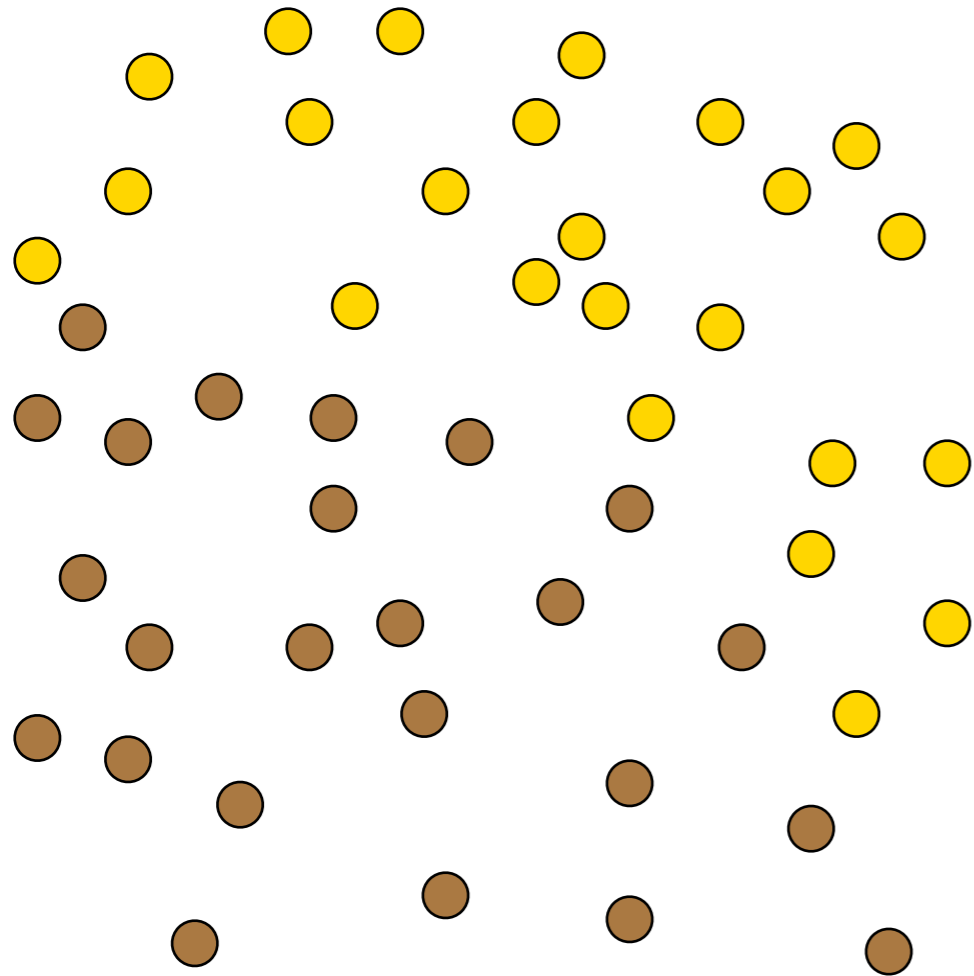
Vantage-Point Tree

Vantage point
Left lower bound
Left upper bound
Right lower bound
Right upper bound



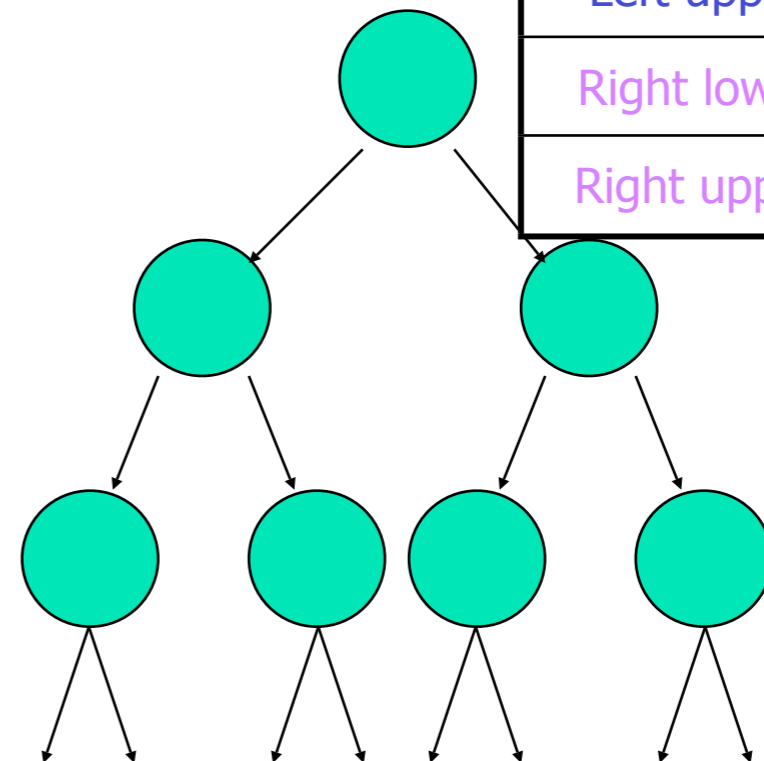
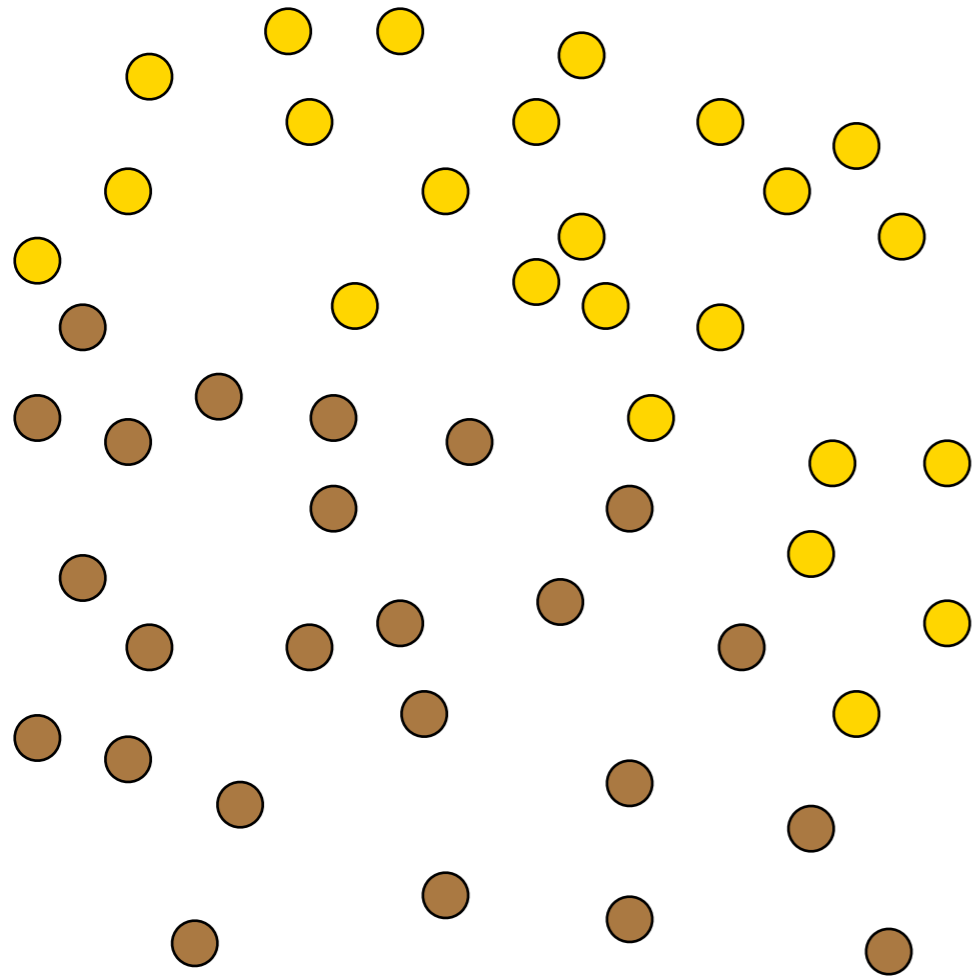
Vantage-Point Tree

Vantage point
Left lower bound
Left upper bound
Right lower bound
Right upper bound



Vantage-Point Tree

Vantage point
Left lower bound
Left upper bound
Right lower bound
Right upper bound



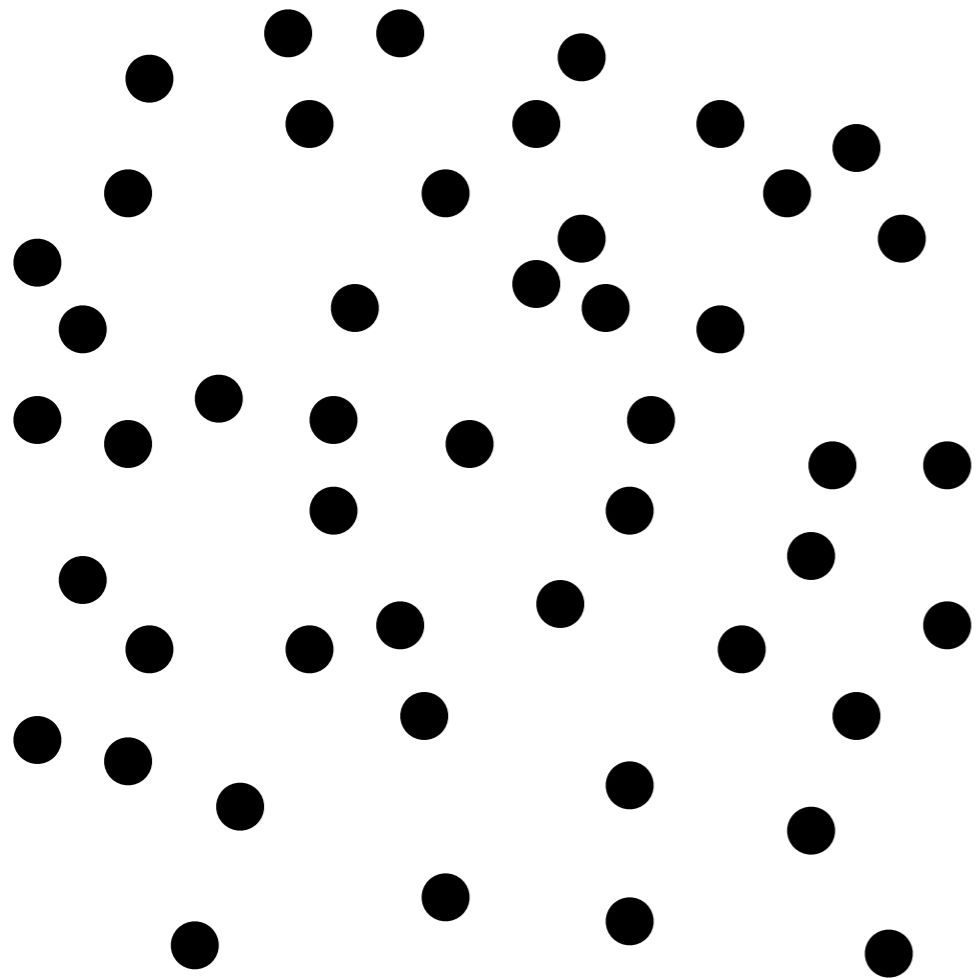


Branch-and-Bound Tree

- 原理類似 **vantage-point tree**，將資料點建立成一個多元樹，利用距離滿足三角不等式的關係，可以減少計算查詢點到資料點的距離的次數
 - 缺點：建樹時使用 **K-means**，麻煩耗時，使用時耗記憶體
 - 困難：資料點少，查詢點到所有資料點的距離都差不多，加速效果不顯著

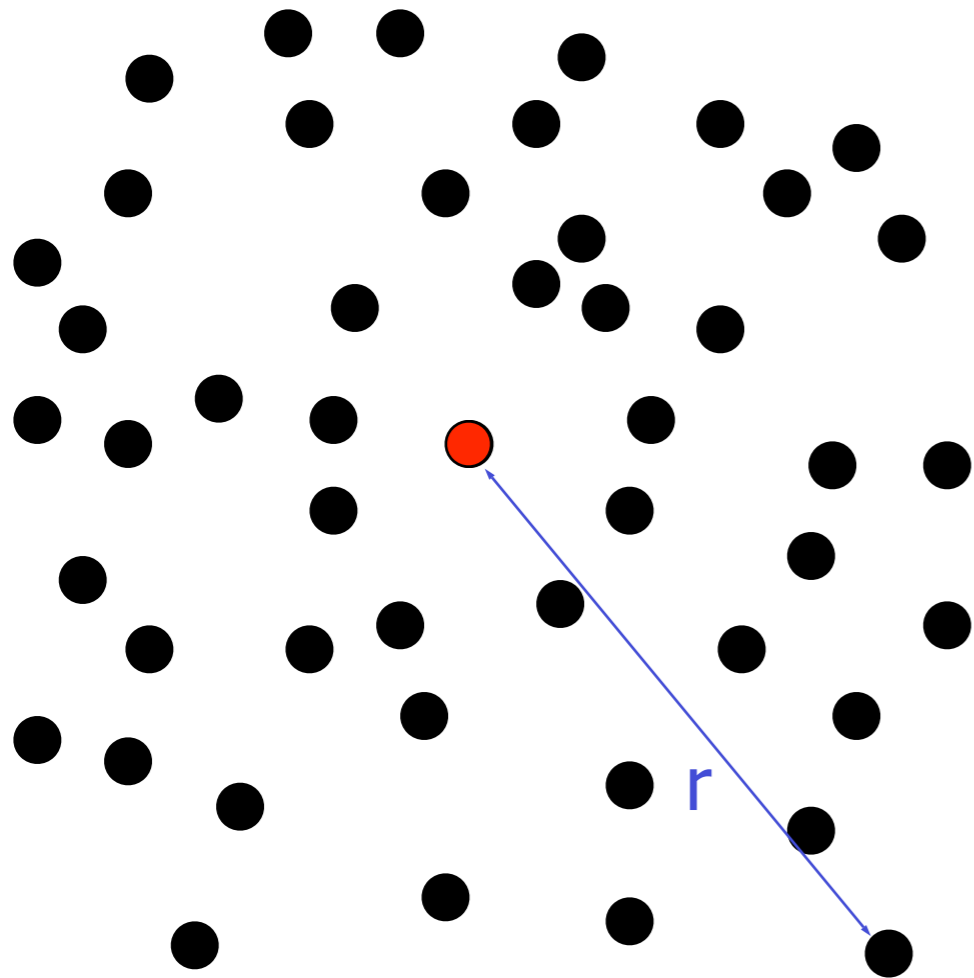


Branch-and-Bound Tree

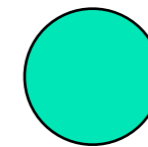
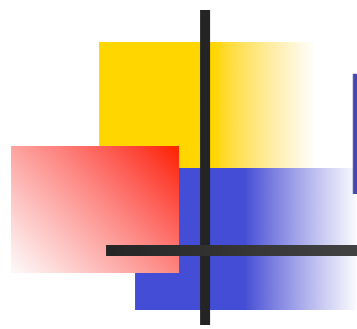




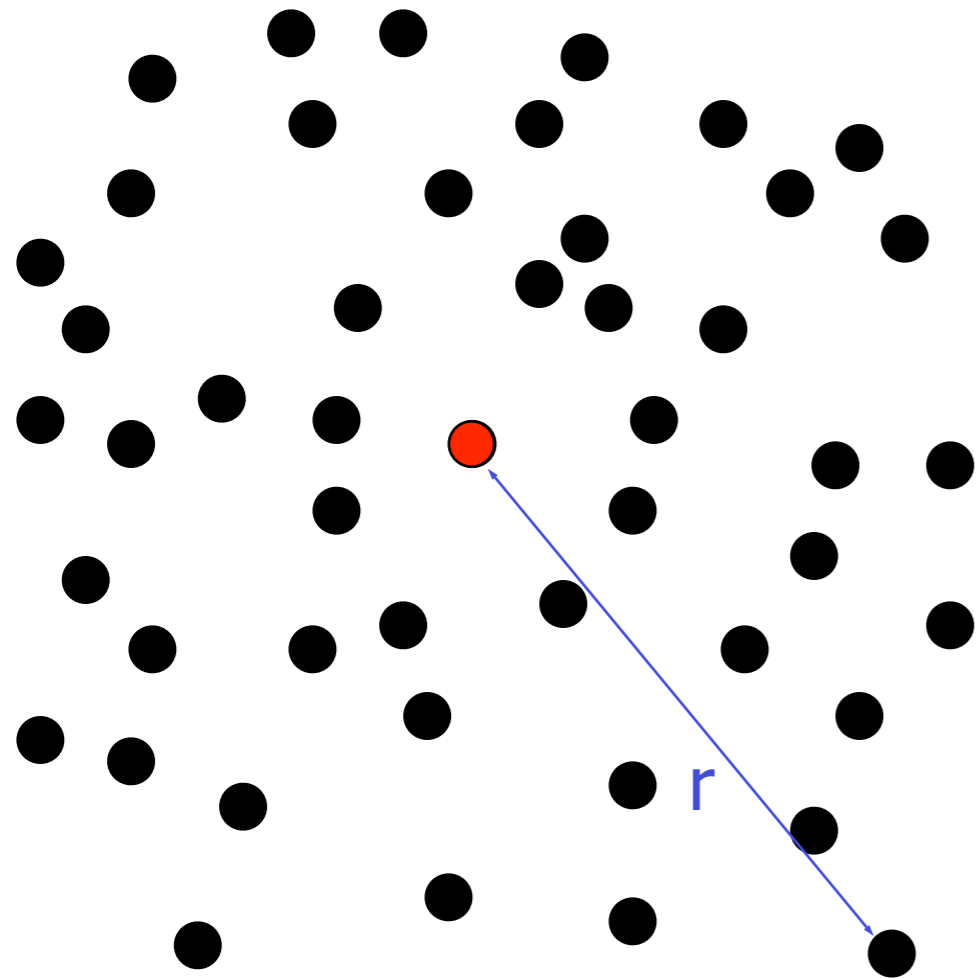
Branch-and-Bound Tree



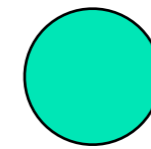
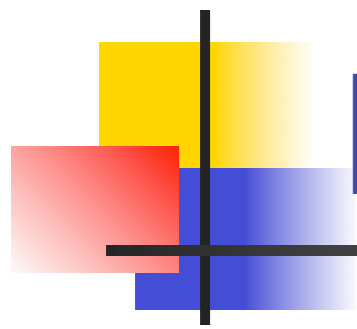
Branch-and-Bound Tree



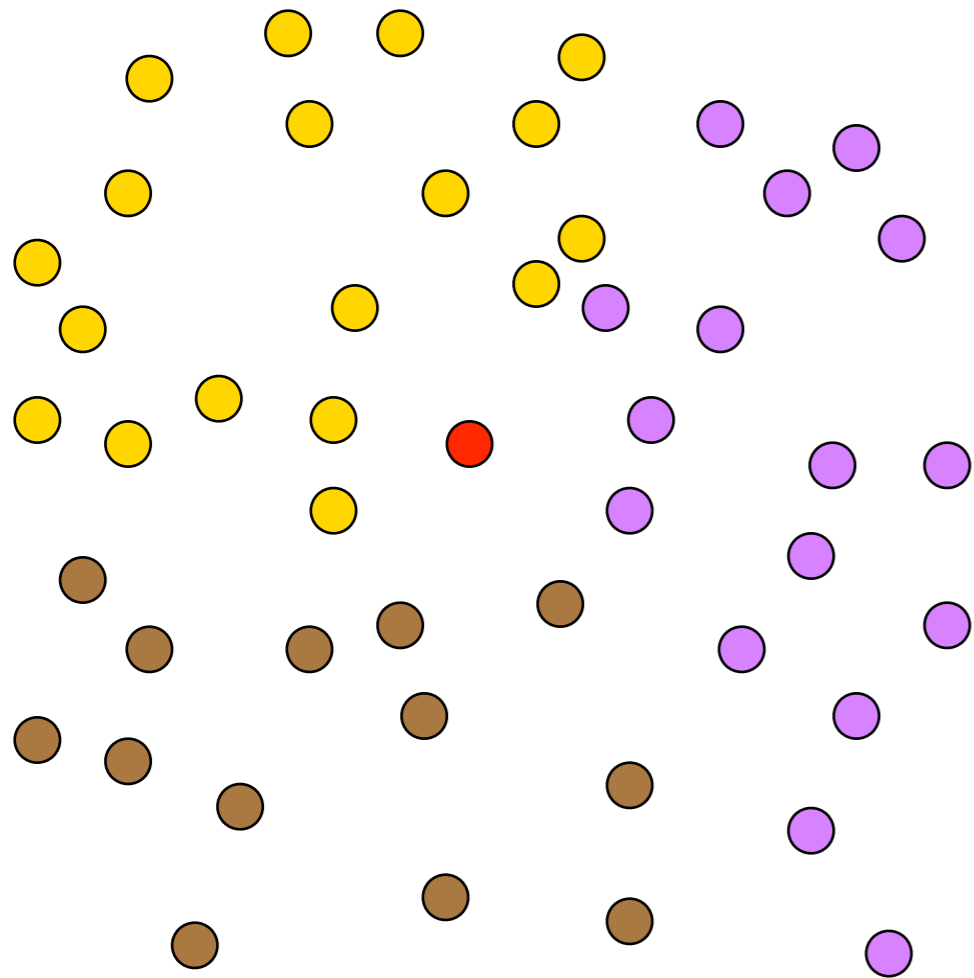
Center
Radius



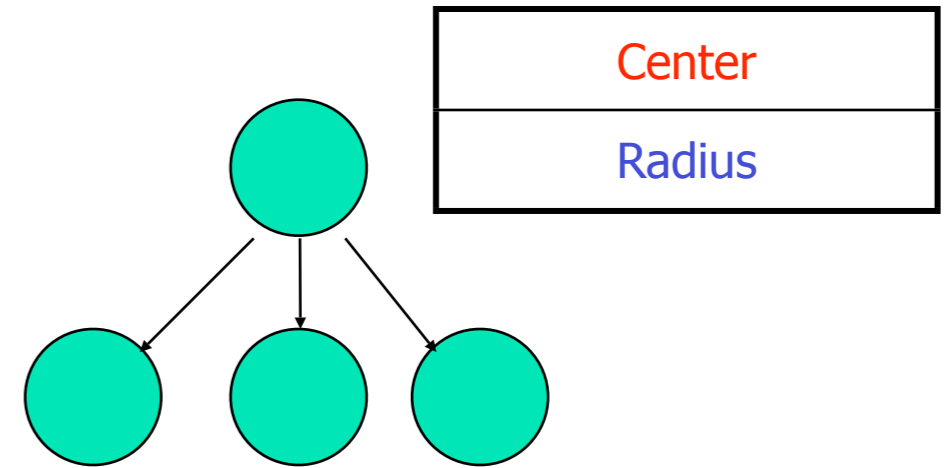
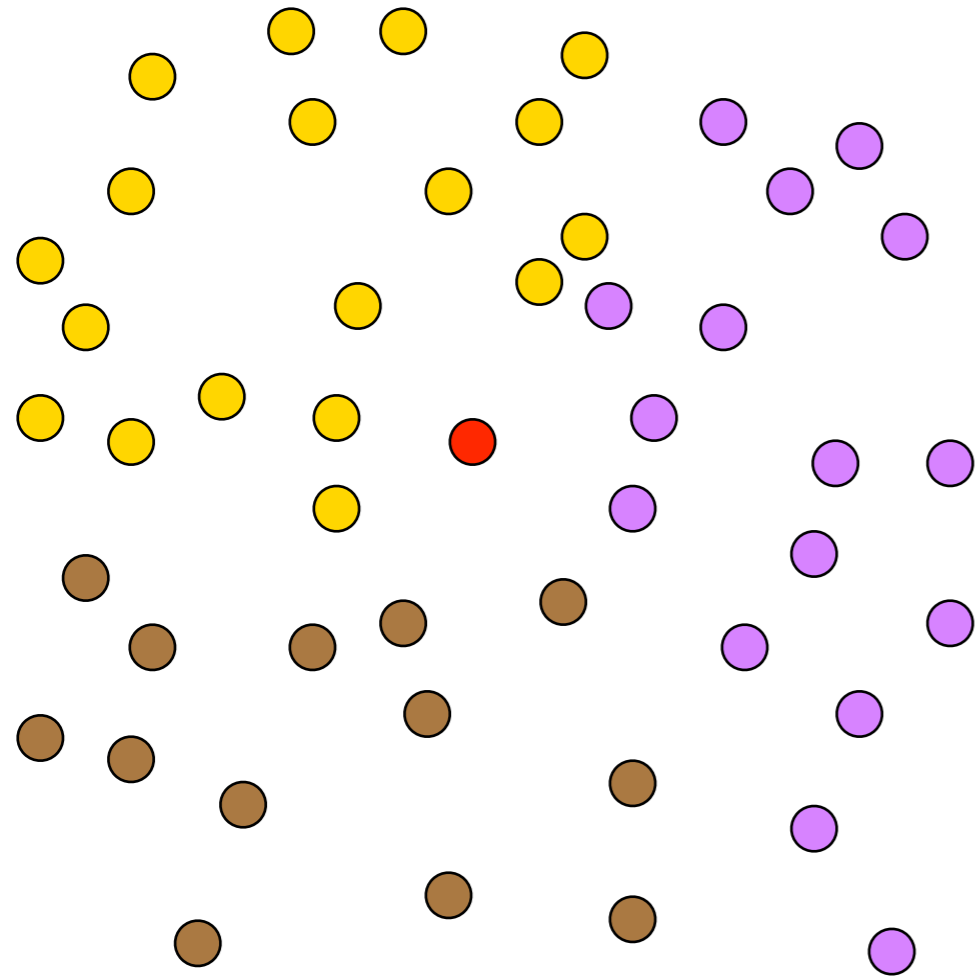
Branch-and-Bound Tree



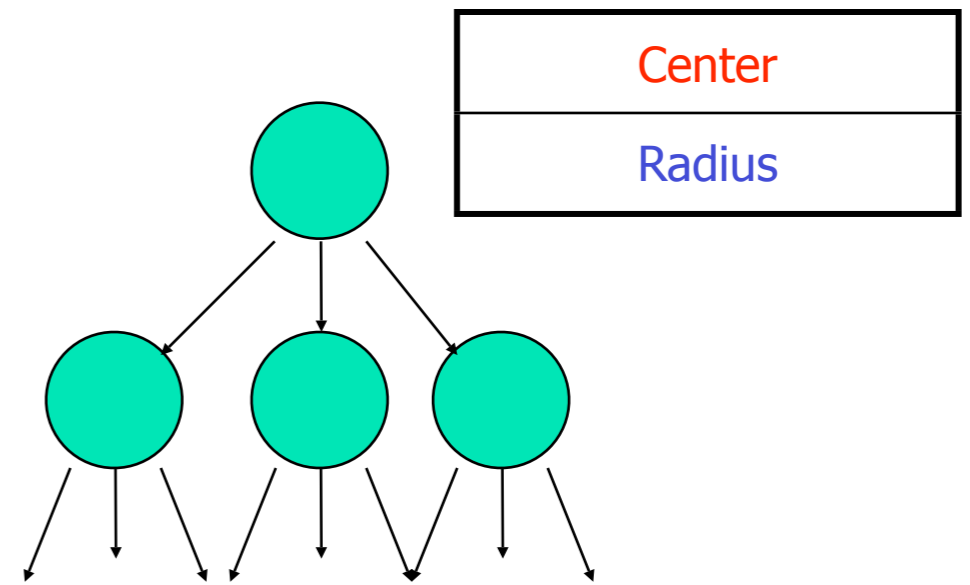
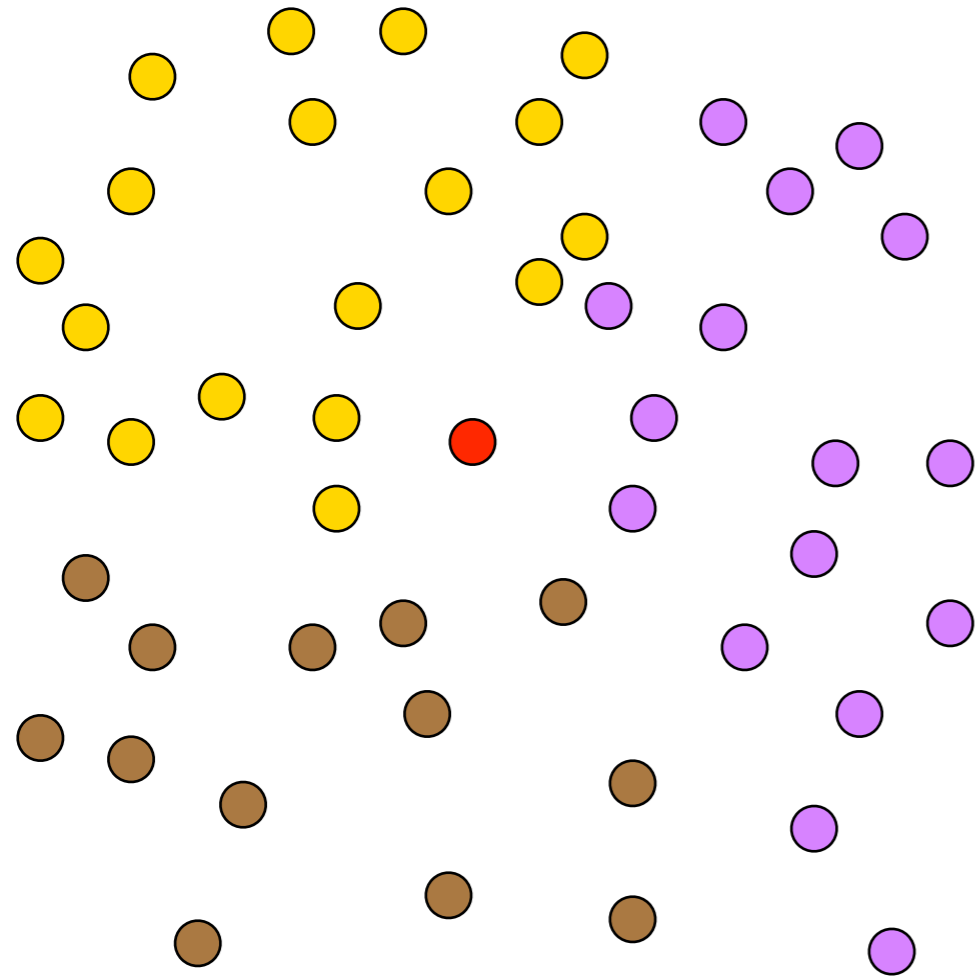
Center
Radius



Branch-and-Bound Tree



Branch-and-Bound Tree



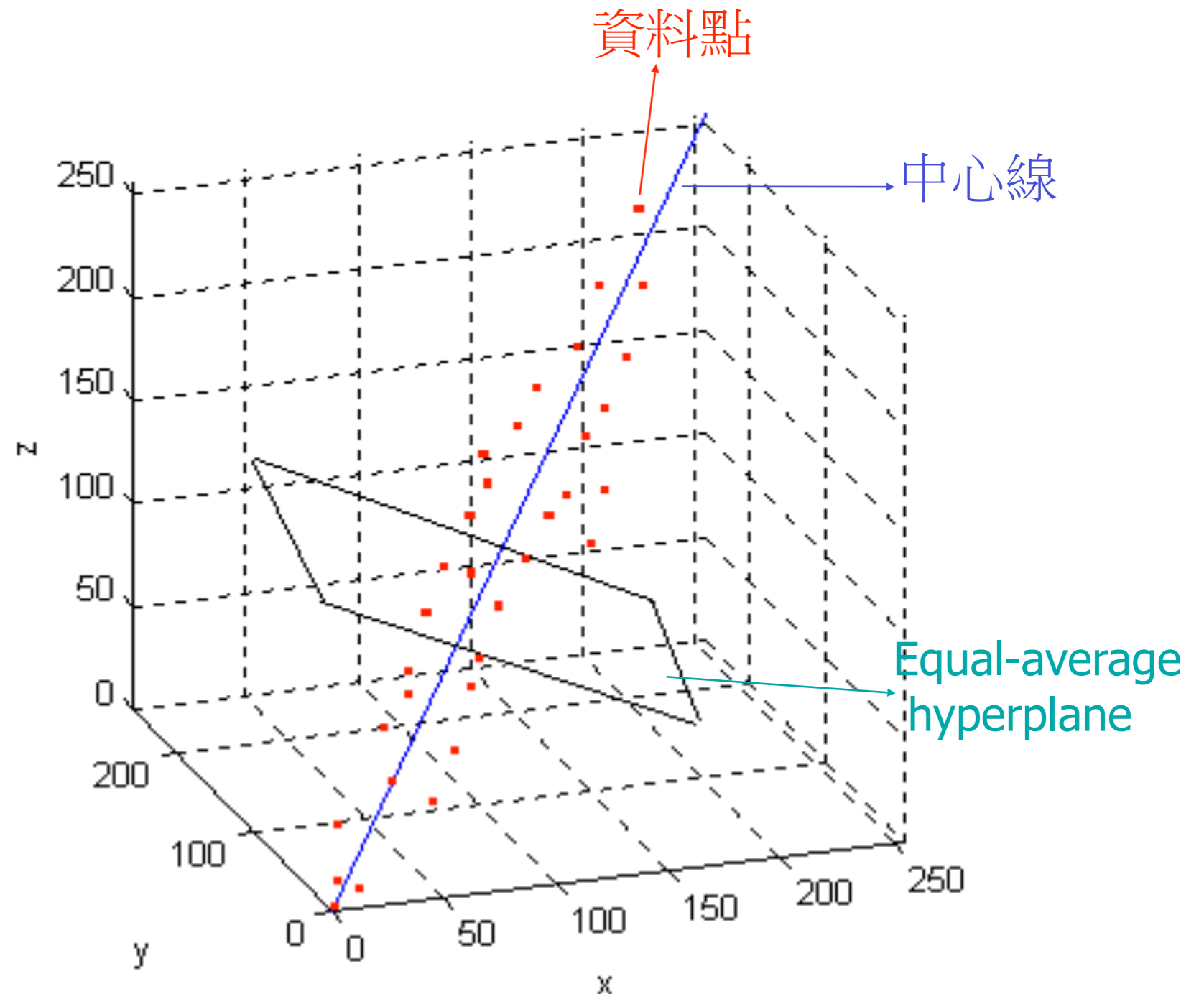


Equal-average Hyperplane Partitioning Method

- 讓資料點沿一直線分散開來，事先算出每一個資料點投影到直線上的位置。利用兩資料點間的距離會大於或等於這兩點投影到該直線上的距離的關係，可以減少計算查詢點到資料點的距離的次數
 - 優點：實作容易，省記憶體
 - 缺點：使用時機較狹隘

Equal-average Hyperplane Partitioning Method

- 中心線 L 是一條通過原點和 $(1, 1, 1, \dots, 1)$ 的直線
- Equal-average hyperplane是和 L 垂直的面



Equal-average Hyperplane Partitioning Method

- 假設 $\mathbf{a}=(s, s, \dots, s)$ 為 L 上的一個點
- 則和 L 相交於 \mathbf{a} 點的 equal-average hyperplane 其方程式為

$$H(x) = s \sum_{i=1}^D x_i - Ds^2 = 0$$

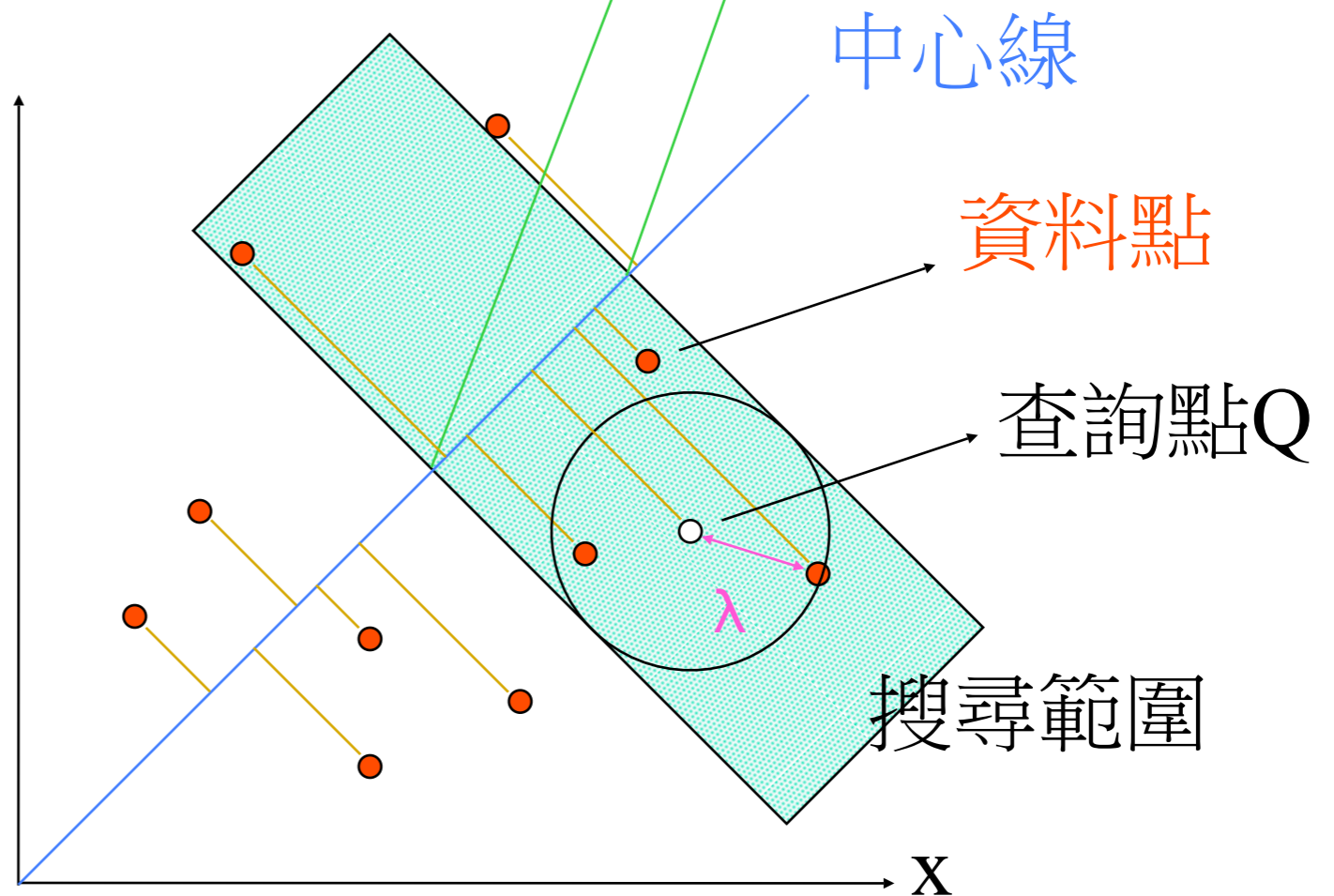
$$\Rightarrow s = \frac{1}{D} \sum_{i=1}^D x_i \quad \leftarrow \text{座標平均值}$$

- 座標平均值相同的點會落在同一個 equal-average hyperplane

Equal-average Hyperplane Partitioning Method

- 座標平均值落在 $[S_{\min}, S_{\max}]$ 之外的點不可能是最近鄰居

$$S_{\min} = \frac{1}{D} \left(\sum_{i=1}^D q_i \right) - \frac{\lambda}{\sqrt{D}}$$
$$S_{\max} = \frac{1}{D} \left(\sum_{i=1}^D q_i \right) + \frac{\lambda}{\sqrt{D}}$$



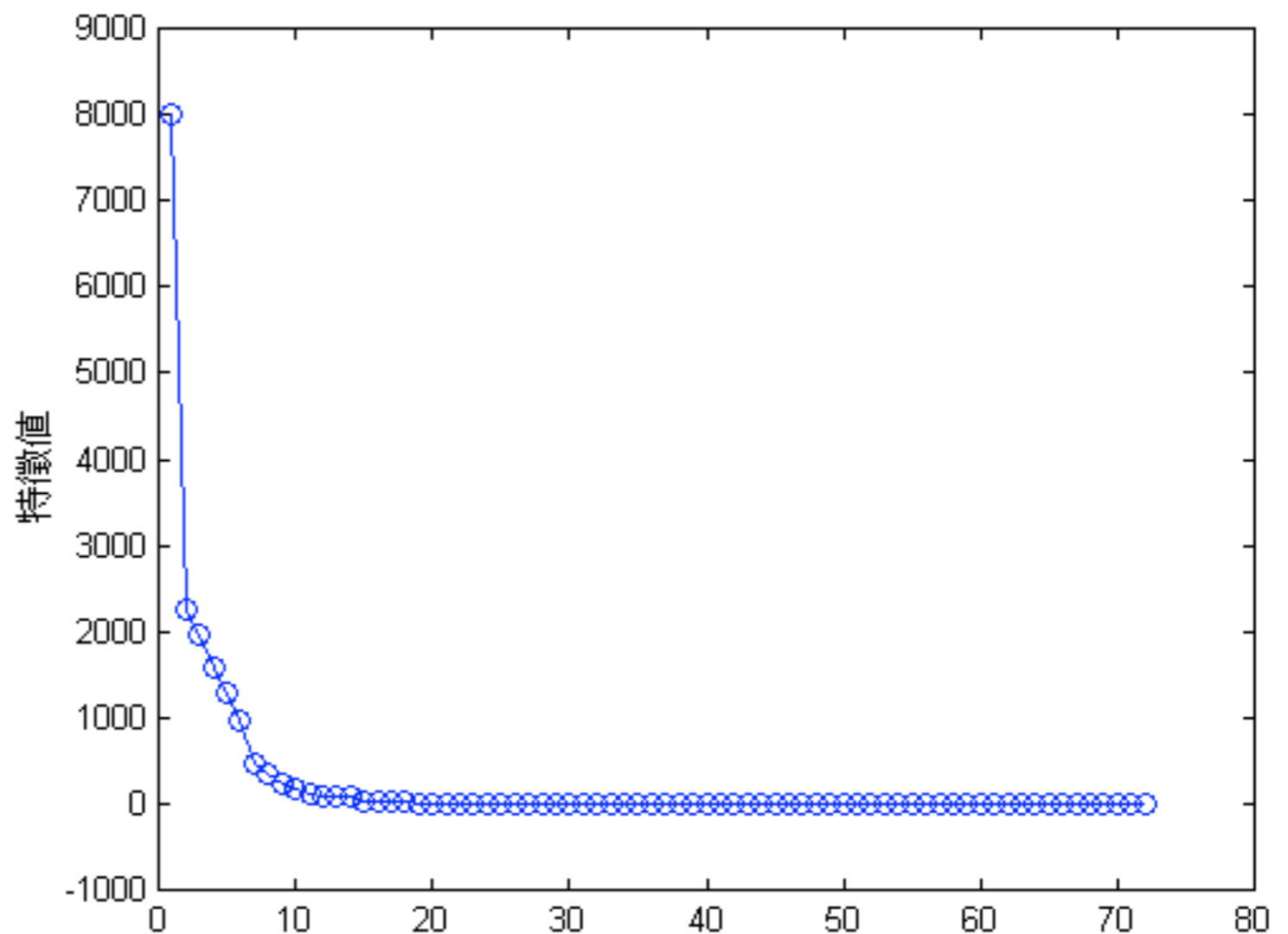


Equal-average Hyperplane Partitioning Method

- 因為我們將每個資料點平移到平均值為0，所以必須另外找一條直線 L 代替原本的中心線
 - 用 **principal component analysis** 找出 L
 - 令 M 矩陣的每一列為資料點 R_{jk} 的座標
 - 求出 $M^T M$ 的eigenvalue和eigenvector
 - 對應到最大eigenvalue的eigenvector即為所求
 - 用 $\frac{R_{jk}}{\sqrt{D}}$ 代替座標平均值

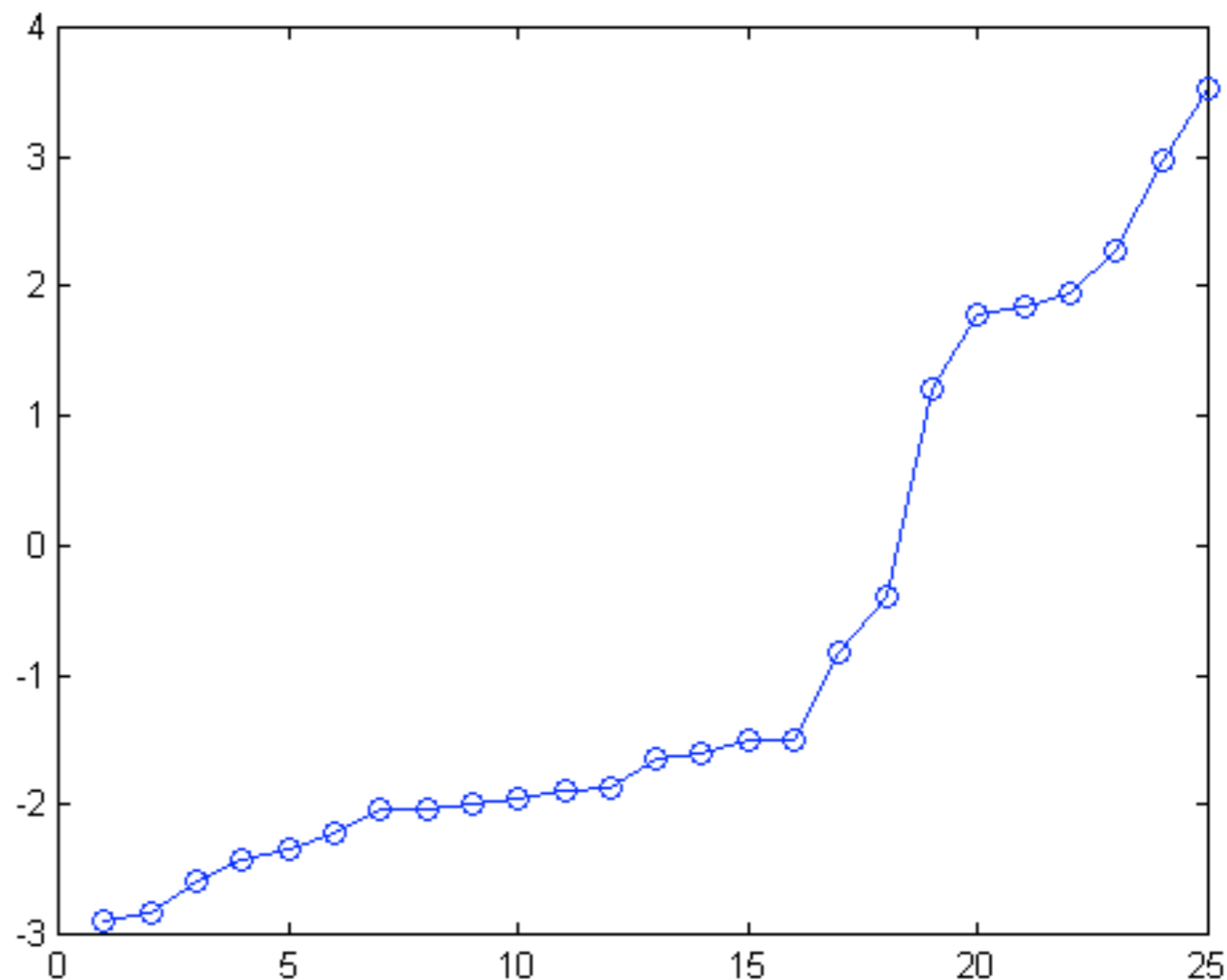
Equal-average Hyperplane Partitioning Method

- 資料庫中某一首歌曲有**25**個片段，每個片段長度為**72**
- 用**PCA**找出來的**72**個特徵值從大到小排序



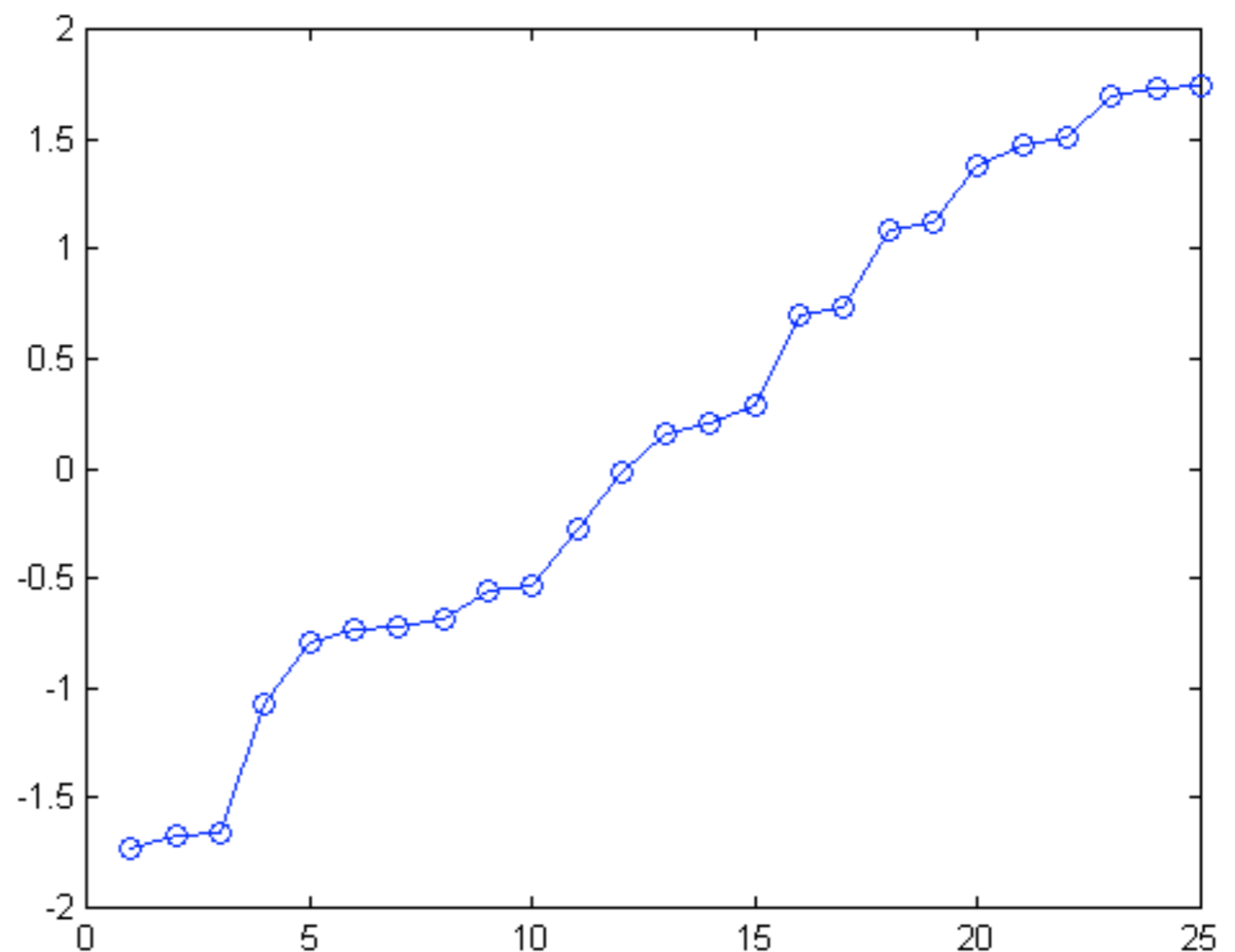
Equal-average Hyperplane Partitioning Method

- 25個片段投影在對應到最大特徵值的特徵向量後的座標平均值從小到大排序



Equal-average Hyperplane Partitioning Method

- 25個片段投影在對應到第二大特徵值的特徵向量後的座標平均值從小到大排序





實驗結果

- 實驗環境

- 電腦 P III 800 , 256MB RAM , Windows 2000

- 資料庫

- 8552首 , 包括中文、台語、英文和日文歌曲

- 測試歌聲wav檔

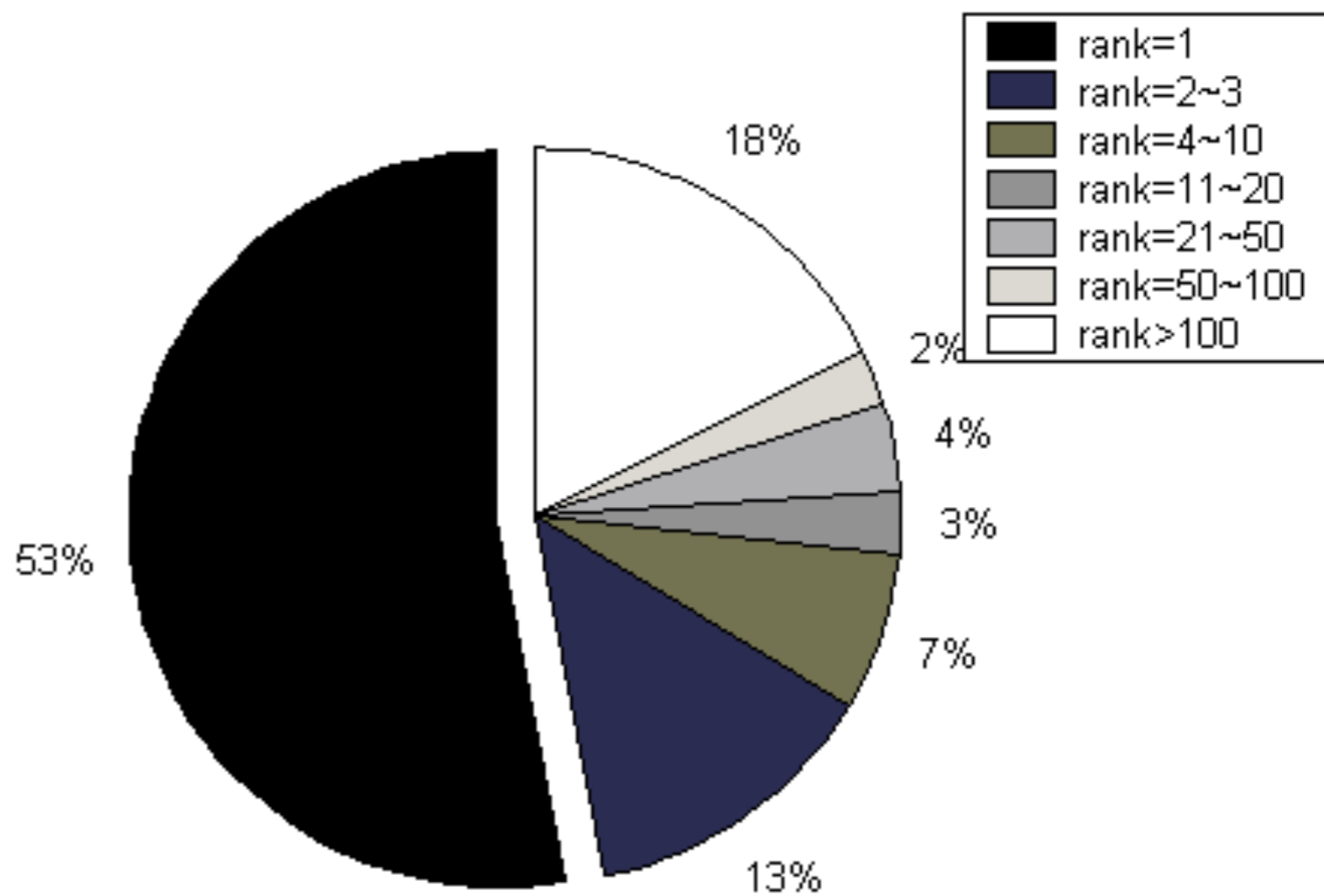
- 從頭唱 1054 首

- 從任意處唱 1650 首

- 每一首長 8 秒鐘

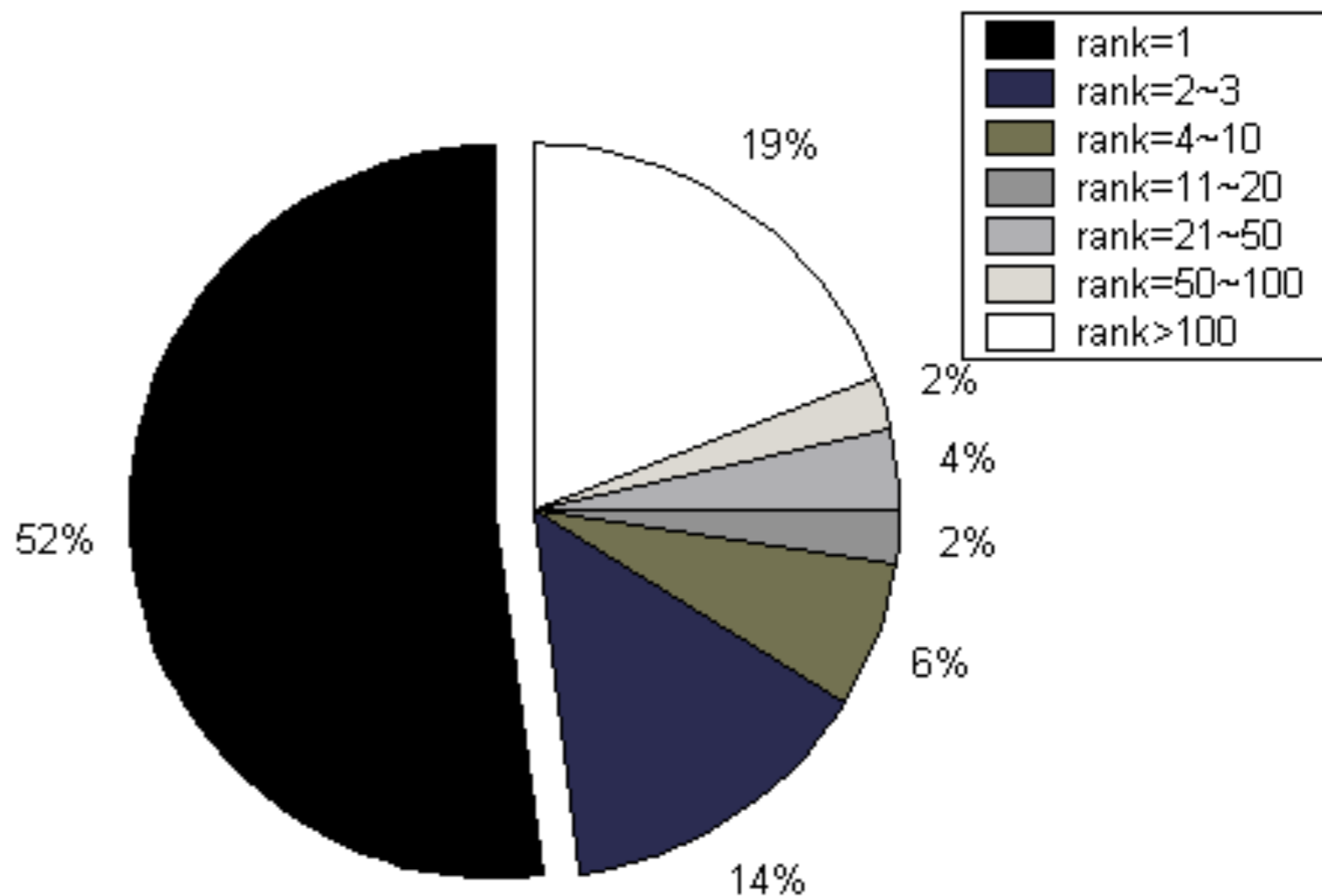
實驗結果—從頭比對

- 只用dtw
- 前三名**66.41%**
- 平均搜尋時間
16.71秒



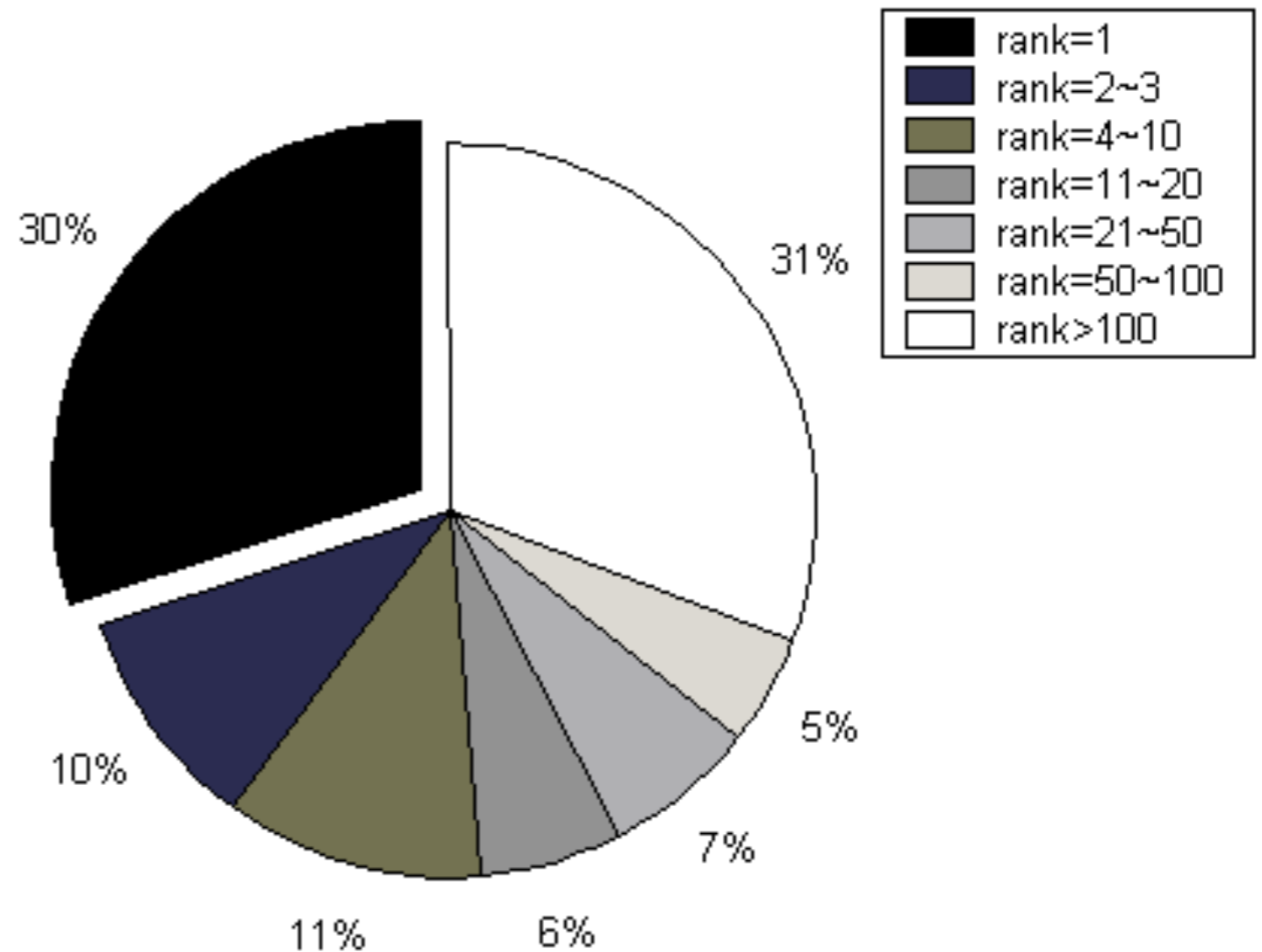
實驗結果—從頭比對

- 只用改良式dtw，保留100名距離
- 前三名66.13%
- 平均搜尋時間6.96秒



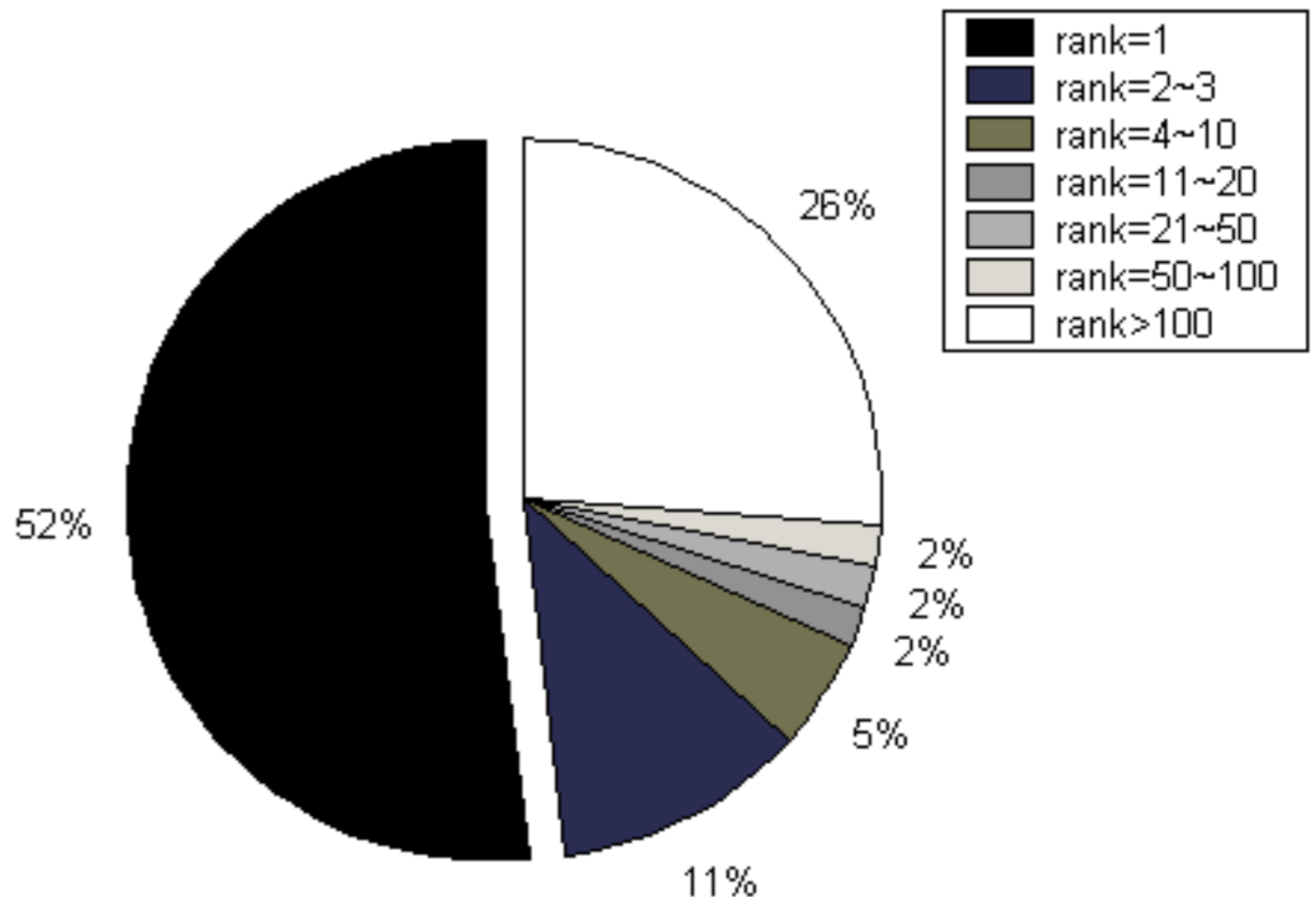
實驗結果—從頭比對

- 只用linear scaling
- 前三名39.94%
- 平均搜尋時間0.1秒

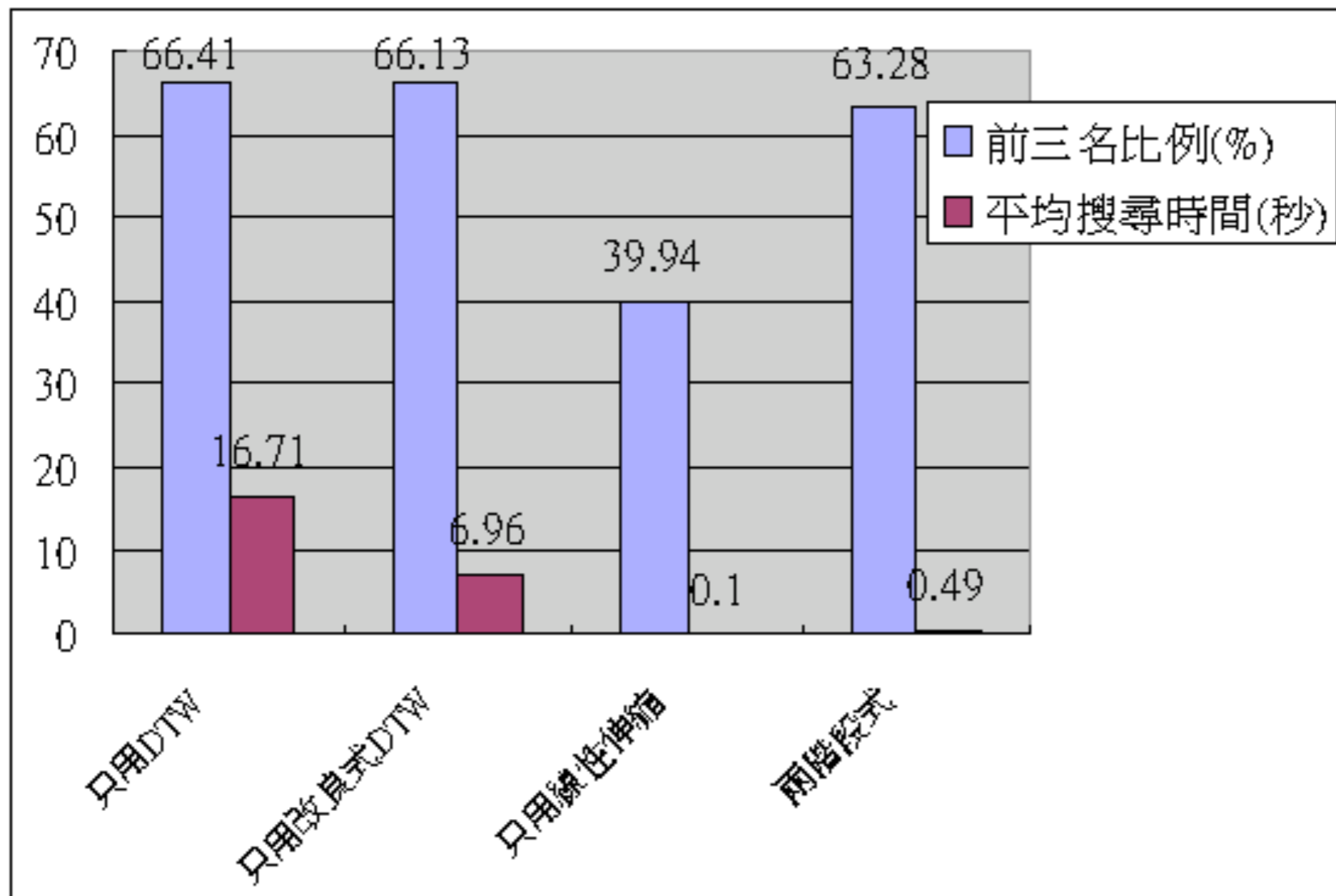


實驗結果—從頭比對

- 用兩階段式比對
- 第一階段保留**200**首歌進入第二階段
- 前三名**63.28%**
- 平均搜尋時間**0.49**秒

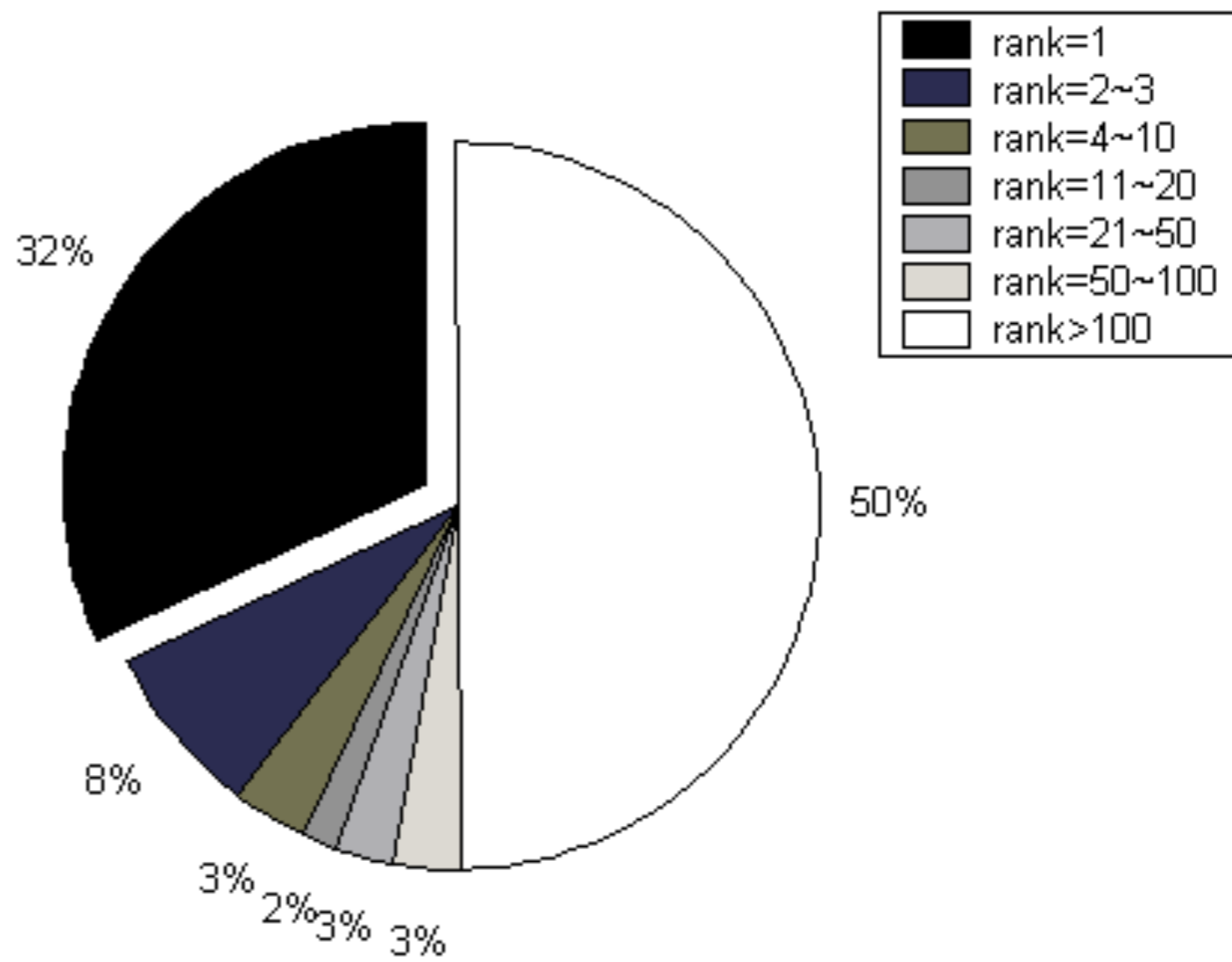


實驗結果—從頭比對比較圖

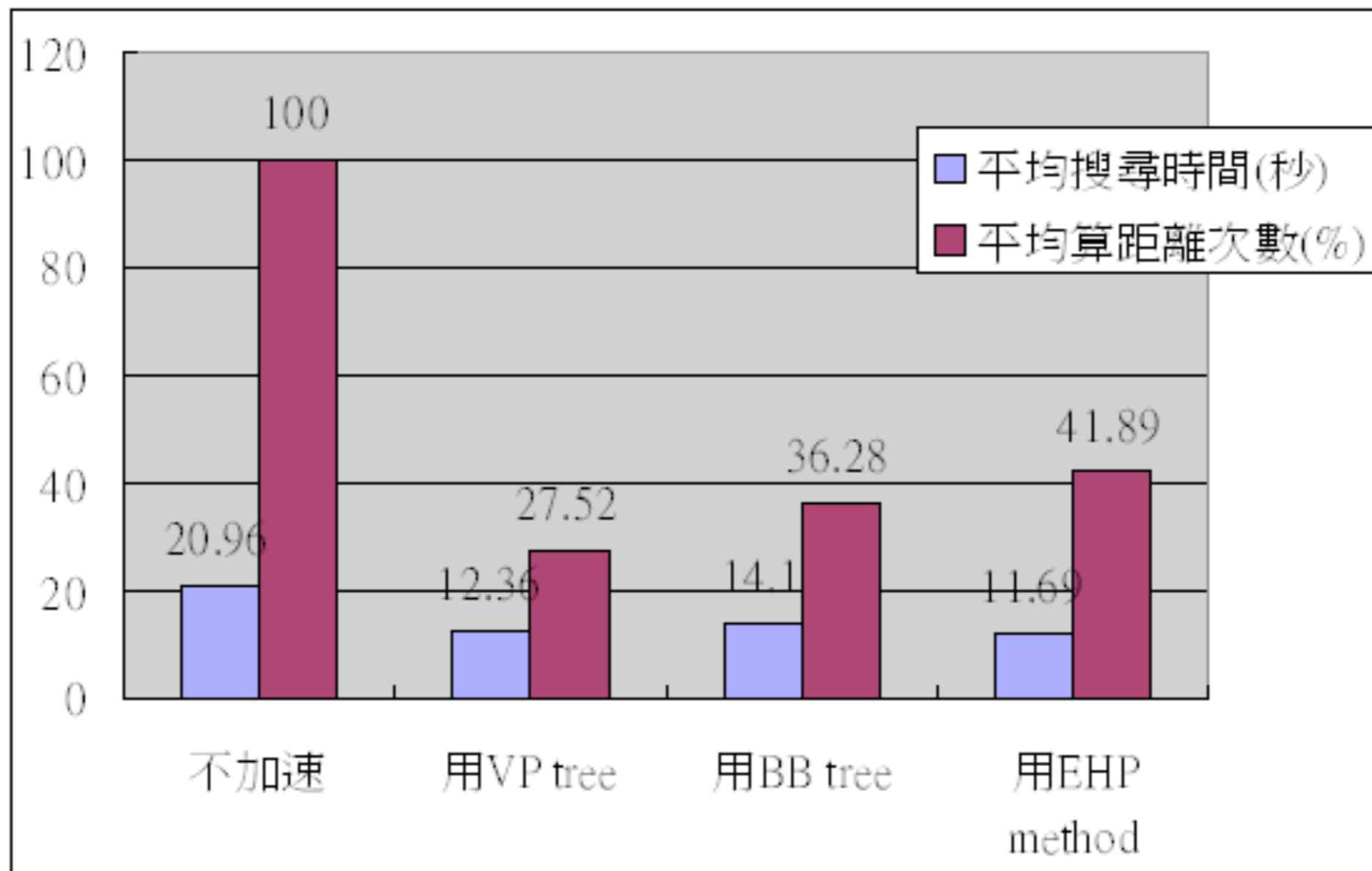


實驗結果—全曲比對

- 用兩階段式比對
- 前三名**39.63%**



實驗結果—全曲比對比較圖





錯誤分析

- 資料庫錯誤
 - 有前奏
 - 同一時間不只一個音符
 - 主旋律錯誤
 - 相同歌曲但不同歌名
- 測試歌聲不良
 - 有雜音
 - 氣不足、音不準
 - 唱錯
 - 拍子不準



結論

- 從頭比對

- 只用dtw → linear scaling + 改良式dtw

- 大約快 34 倍
- 前三名比例下降不到 4%

- 全曲比對

- 只用改良式dtw → linear scaling + 改良式dtw

- 大約快 23 倍
- 第一階段不用加速方法 → 第一階段用equal-average hyperplane partitioning method
 - 大約快 1.8 倍
 - 平均一首歌約多佔 2.7 K byte 的記憶體



3. Video retrieval techniques



Differences and relations between image and video

- Images are static, but video are dynamic.
- Video stream can be viewed as sequence of image frames.



Structuralizing video data

- There are several semantic content layers such as scenes and shots in a video program.
 - These layers are erased when they are displayed for audience, which weakens the ability for user dealing with raw video data.



Fundamental definitions in video structurization

- Frame
- Shot
- Key frame
- Scene
- Group



Common workflow of video structurelization



Homework

- Implement a naïve image retrieval system
 - Using color moments

