KNOWLEDGE MEDIA
KMi
INSTITUTE

Multimedia Queries and Indexing

Prof Stefan Rüger

Multimedia and Information Systems
Knowledge Media Institute
The Open University

http://kmi.open.ac.uk/mmis

MMIS
Multimedia and Information Systems

KNOWLEDGE MEDIA KMi INSTITUTE

Multimedia queries and indexing

1. What are multimedia queries?
2. Fingerprinting
3. Image search and indexing
4. Evaluation
5. Browsing, search and geography

KNOWLEDGE MEDIA INSTITUTE

Multimedia queries and indexing

1. What are multimedia queries?
2. Fingerprinting
   - Overview
   - How Shazam works
   - Subfingerprinting
   - Locality sensitive hashing
   - Min hash algorithm
   - Sift image features
   - Surf as alternative
3. Image search and indexing
4. Evaluation
5. Browsing, search and geography

KNOWLEDGE MEDIA INSTITUTE

Kiosk at Voronezh State University

The Open University

KNOWLEDGE MEDIA INSTITUTE

How does near-duplicate detection work?

Fingerprinting technique

1 Compute salient points

2 Extract “characteristics” from vincinity (feature)

3 Make invariant under rotation & scaling

4 Quantise: create visterms

5 Index as in text search engines

6 Check/enforce spatial constraints after retrieval

KNOWLEDGE MEDIA INSTITUTE

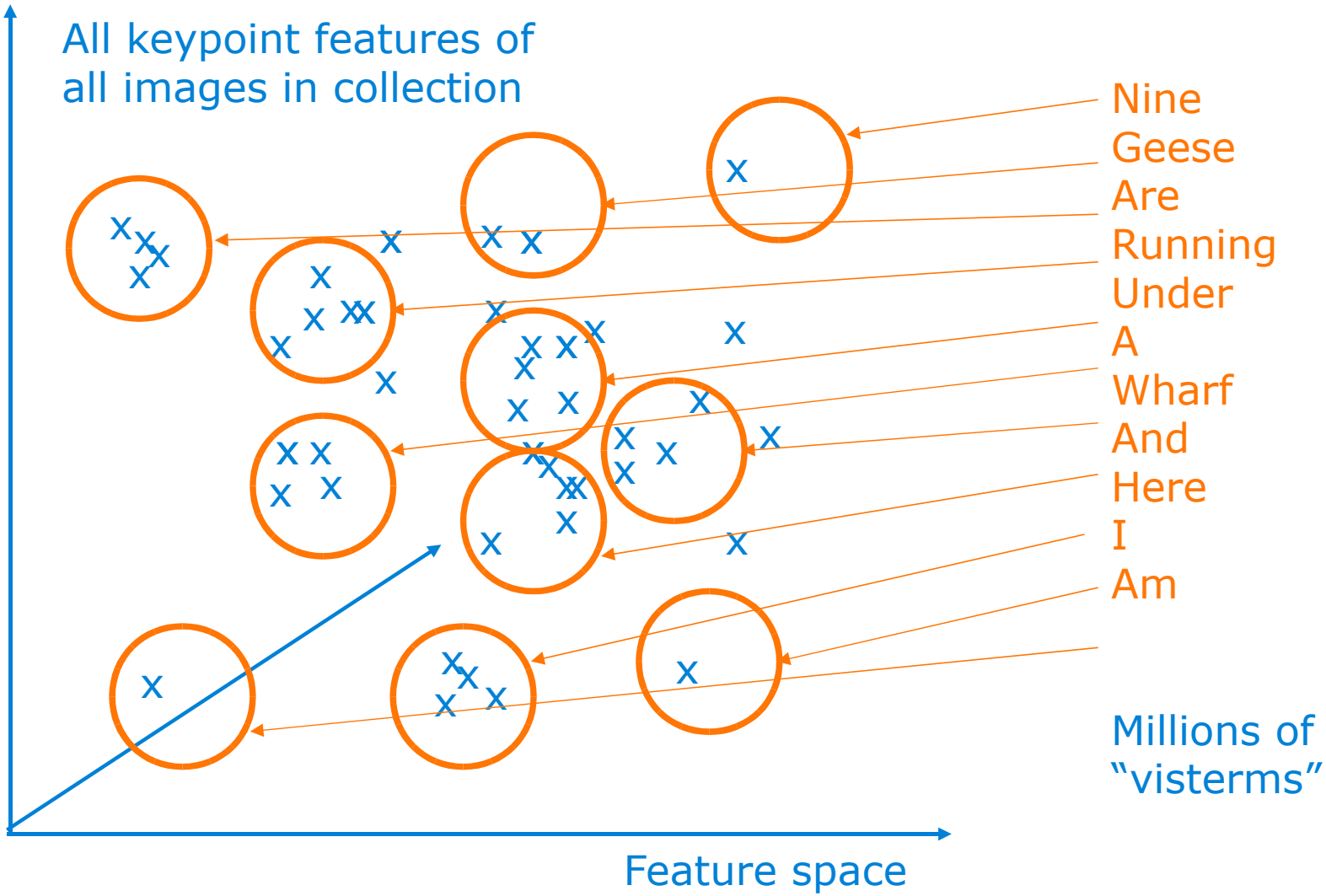
NDD: Compute salient points and features

Eg, SIFT features: each salient point described by a feature vector of 128 numbers; the vector is invariant to scaling and rotation

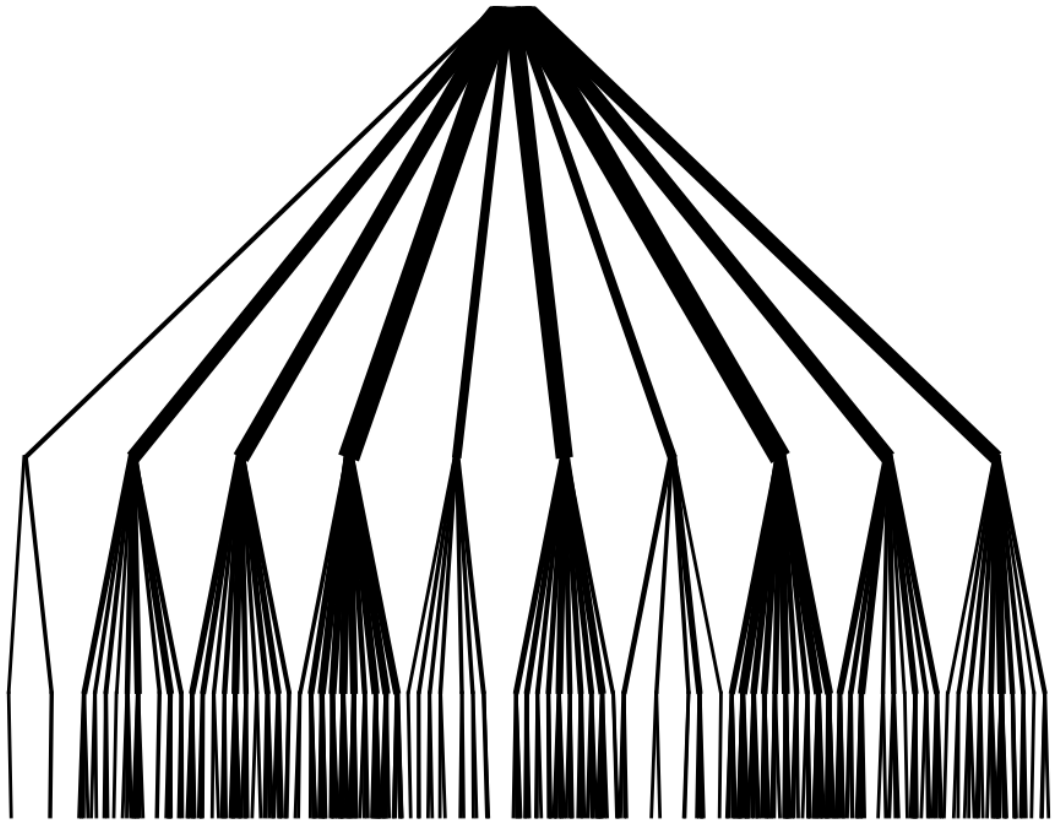
[Lowe2004 – http://www.cs.ubc.ca/~lowe/keypoints/]

The Open University

NDD: Keypoint feature space clustering



Clustering
Hierarchical k-means



[Nister and Stewenius, CVPR 2006]

KNOWLEDGE MEDIA INSTITUTE

NDD: Encode all images with visterms

Jkjh Geese Bjlkj Wharf
Ojkkjhhj Kssn Klkekjl Here
Lkjkll Wjjkll Kkjlk Bnm
Kllkgjg Lwoe Boerm ...

The Open University

KNOWLEDGE MEDIA INSTITUTE

NDD: query like text

At query time compute salient points, keypoint features and visterms

Query against database of images represented as bag of vistems

Query

Joiu Gddwd Bipoi Wueft
Oiooiuui Kwwn Kpodoip Hdfd
Loiopp Wiiopp Koipo Bnm
Kppoyiy Lsld Bldfm ...

KNOWLEDGE MEDIA INSTITUTE

NDD: Check spatial constraints

[with Suzanne Little, SocialLearn project]

The Open University

KNOWLEDGE MEDIA INSTITUTE

How does near-duplicate detection work?

Fingerprinting technique

1 Compute salient points

2 Extract “characteristics” from vincinity (feature)

3 Make invariant under rotation & scaling

4 Quantise: create visterms

5 Index as in text search engines

6 Check/enforce spatial constraints after retrieval

KNOWLEDGE MEDIA KMi INSTITUTE

How Shazam works - Spectrogram

Compute energy for all (frequency,time) pairs using a Fourier transform under a Hann window w

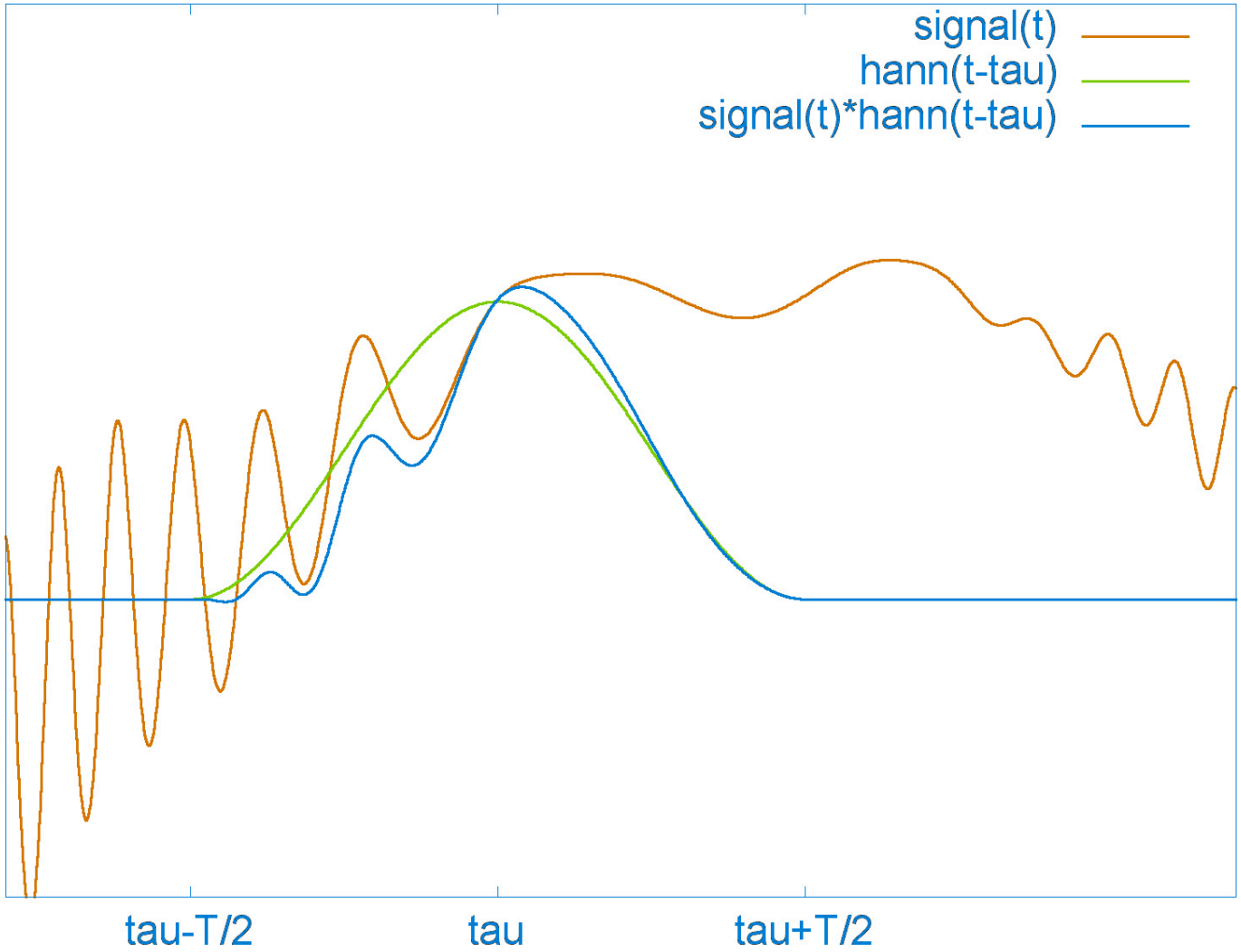
$$\text{spectrogram}(f,\tau) = \left| \int_{-\infty}^{\infty} s(t)w(t-\tau)e^{jft}dt \right|^2$$

$$w(t) = \begin{cases} \frac{1}{2} + \frac{1}{2}\cos\left(\frac{2\pi t}{T}\right) & \text{if } t \in [-T/2, T/2] \\ 0 & \text{otherwise} \end{cases}$$

The Open University

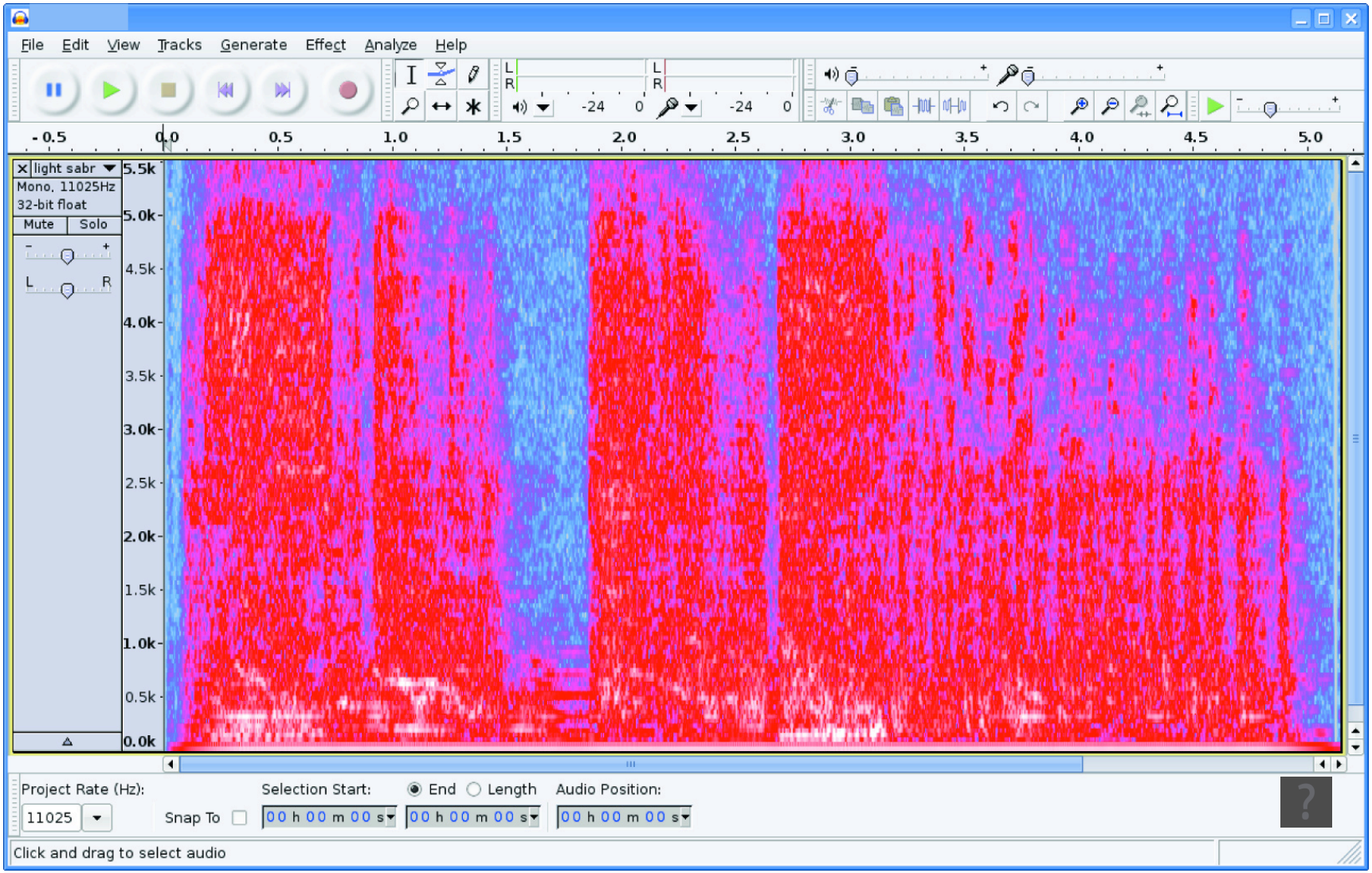
KNOWLEDGE MEDIA
KMi
INSTITUTE

Hann window application



KNOWLEDGE MEDIA
KMi
INSTITUTE

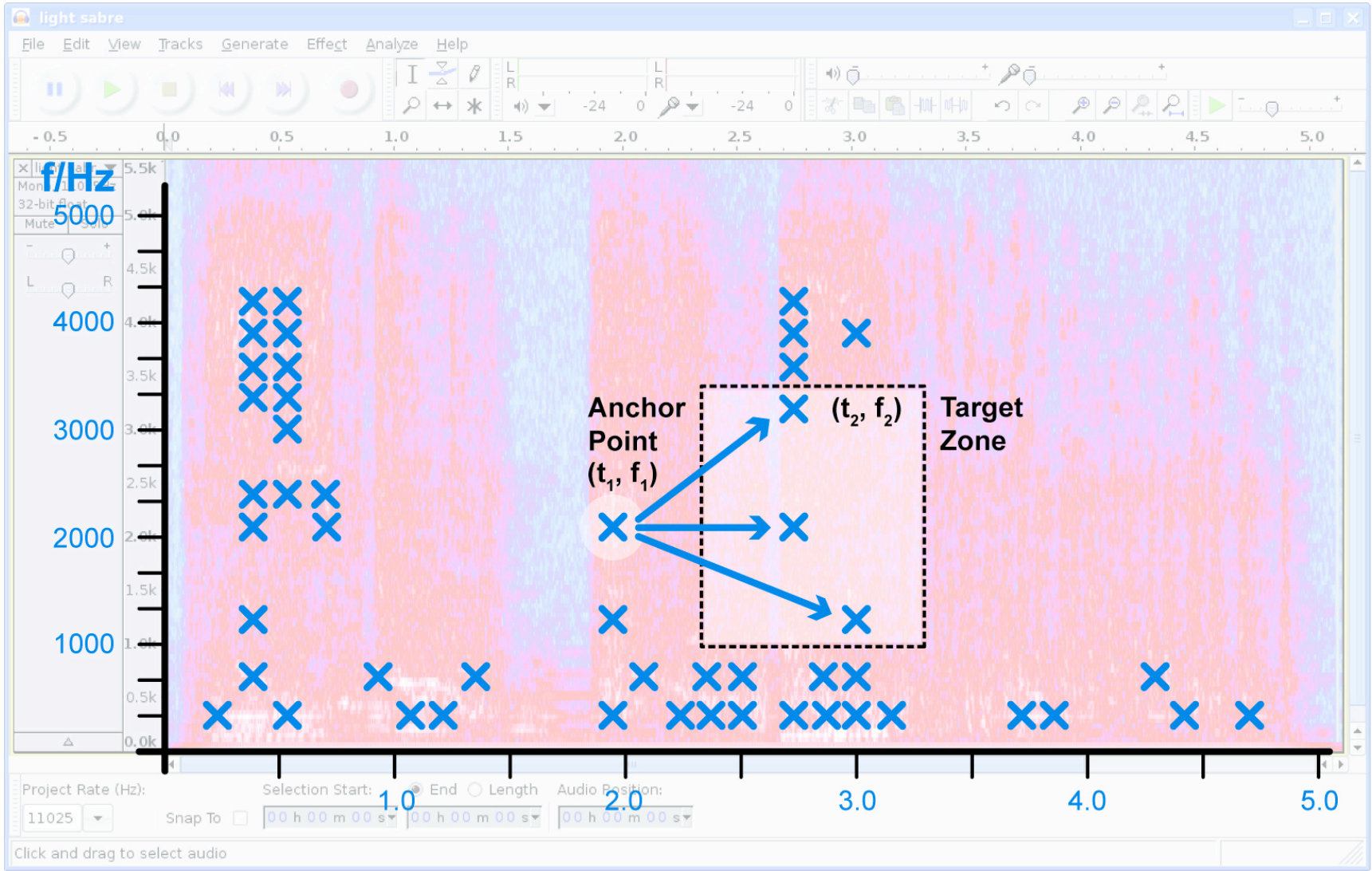
How Shazam works: audio fingerprinting



The Open University

KNOWLEDGE MEDIA KMi INSTITUTE

Salient points



Encoding: (f_1, f_2, t_2-t_1) hashes to (t_1, id)

[Wang(2003), An industrial-strength search algotithm, ISMIR]

The Open University

KNOWLEDGE MEDIA KMi INSTITUTE

Temporal consistency check of query

Every query vector (f_1,f_2, t^q_2-t^q_1) is matched to the database.

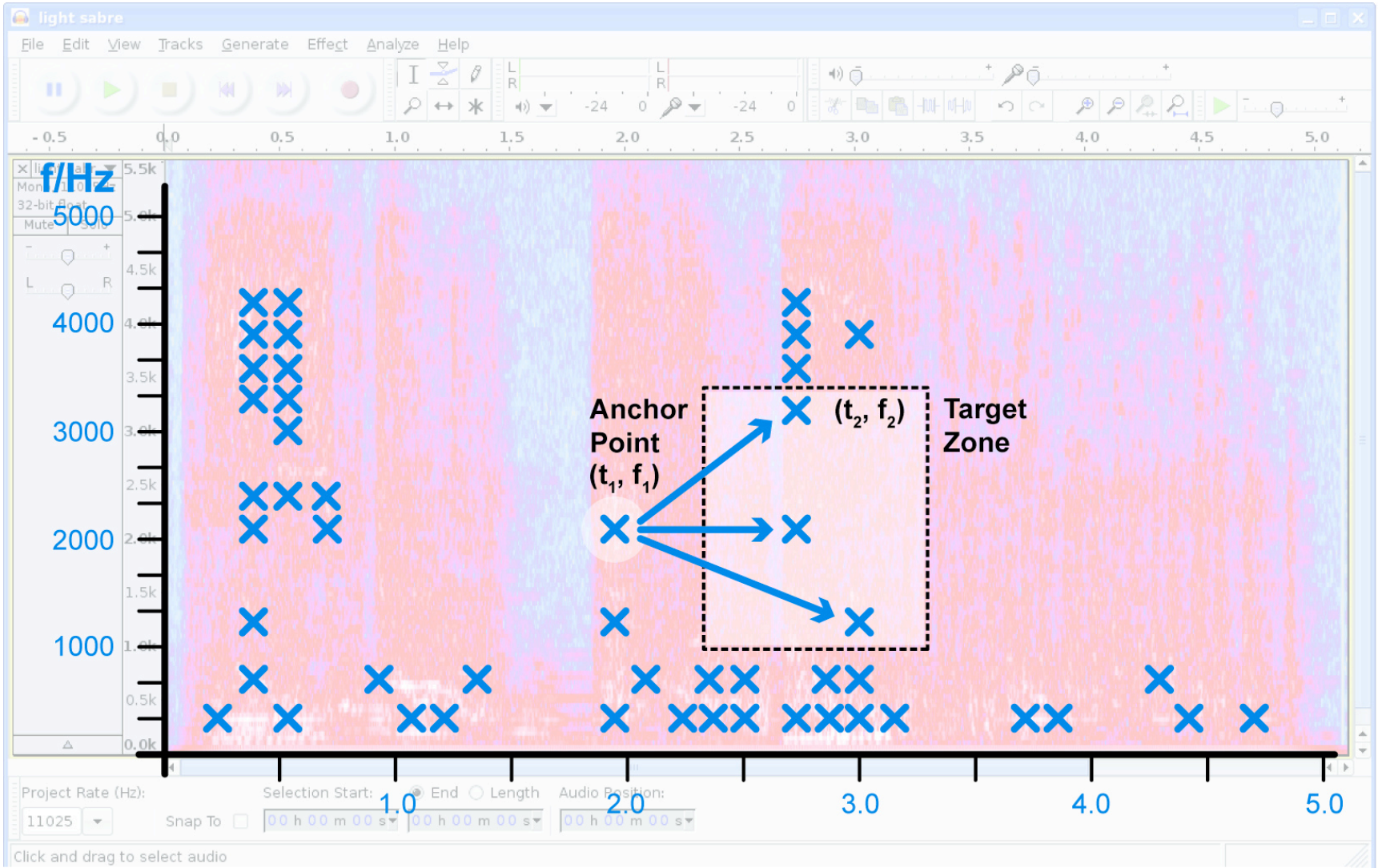
You get a list of possible (t^{id}_1, id) values (some are false positives).

Create a histogram of t^{id}_1-t^q_1 values (temporal consistency check!)

A substantial peak in this histogram means that the query has matched song id at time offset t^{id}_1-t^q_1.

KNOWLEDGE MEDIA KMi INSTITUTE

Entropy considerations



Specificity: Encoding (f_1, f_2, t_2-t_1) to use 30 bit

The Open University

KNOWLEDGE MEDIA KMi INSTITUTE

Exercise
Shazam's constellation pairs

Assume that the typical survival probability of each 30-bit constellation pair after deformations that we still want to recognise is p, and that this process is independent per pair. Which encoding density, ie, the number of constellation pairs per second, would you need on average so that a typical query of 10 seconds exhibits at least 10 matches in the right song with a probability of at least 99.99%? Under these assumptions, further assuming that the constellation pair extraction looks like a random independent and identically distributed number, what is the false positive rate for a database of 4 million songs each of which is 5 minutes long on average?

Philips Research

Divide frequency scale into 33 frequency bands between 300 Hz and 2000 Hz
Logarithmic spread – each frequency step is 1/12 octave, ie, one semitone

Divide time axis into blocks of 256 windows of 11.6 ms (3 seconds)

E(m,n) is the energy of the m-th frequency at n-th time in spectrogram

For each block extract 256 sub-fingerprints of 32 bits each

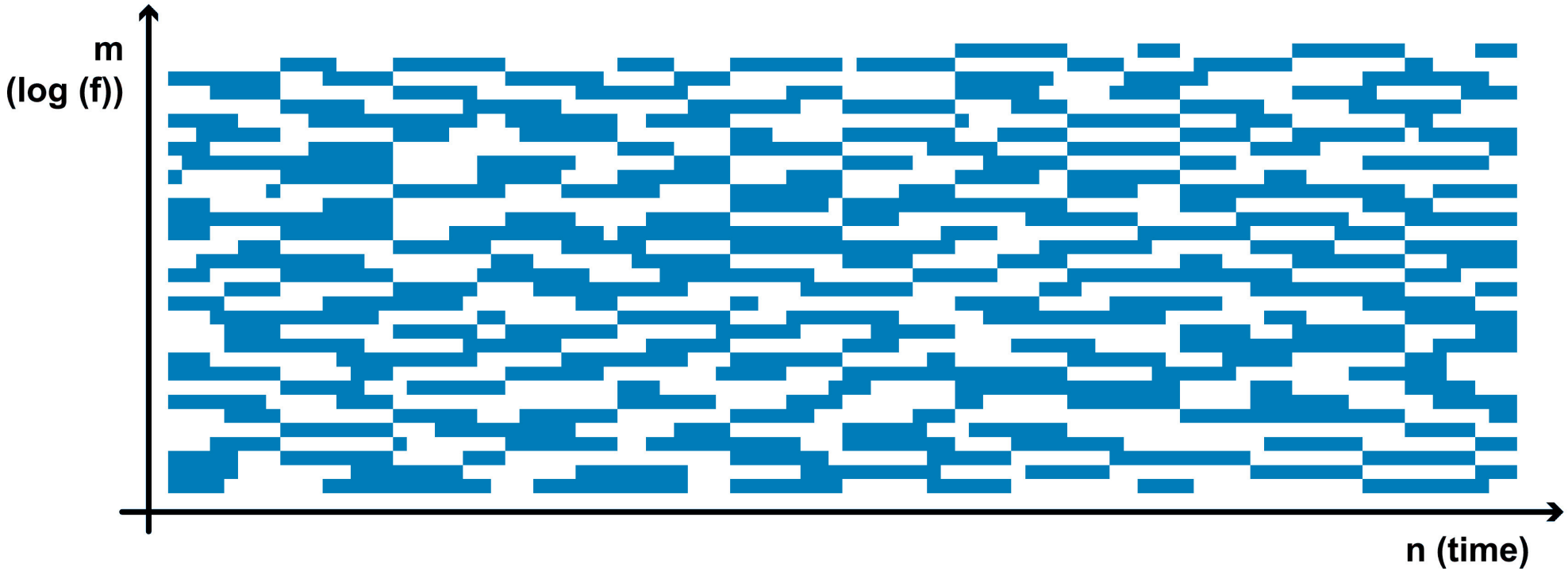
$$b(m,n) = \text{sign}\left([E(m,n) - E(m+1,n)] - [E(m,n+1) - E(m+1,n+1)]\right)$$

$0 \leq m \leq 31$ (frequency)
$0 \leq n \leq 255$ (time)

[Haitsma and Kalker, 2003]

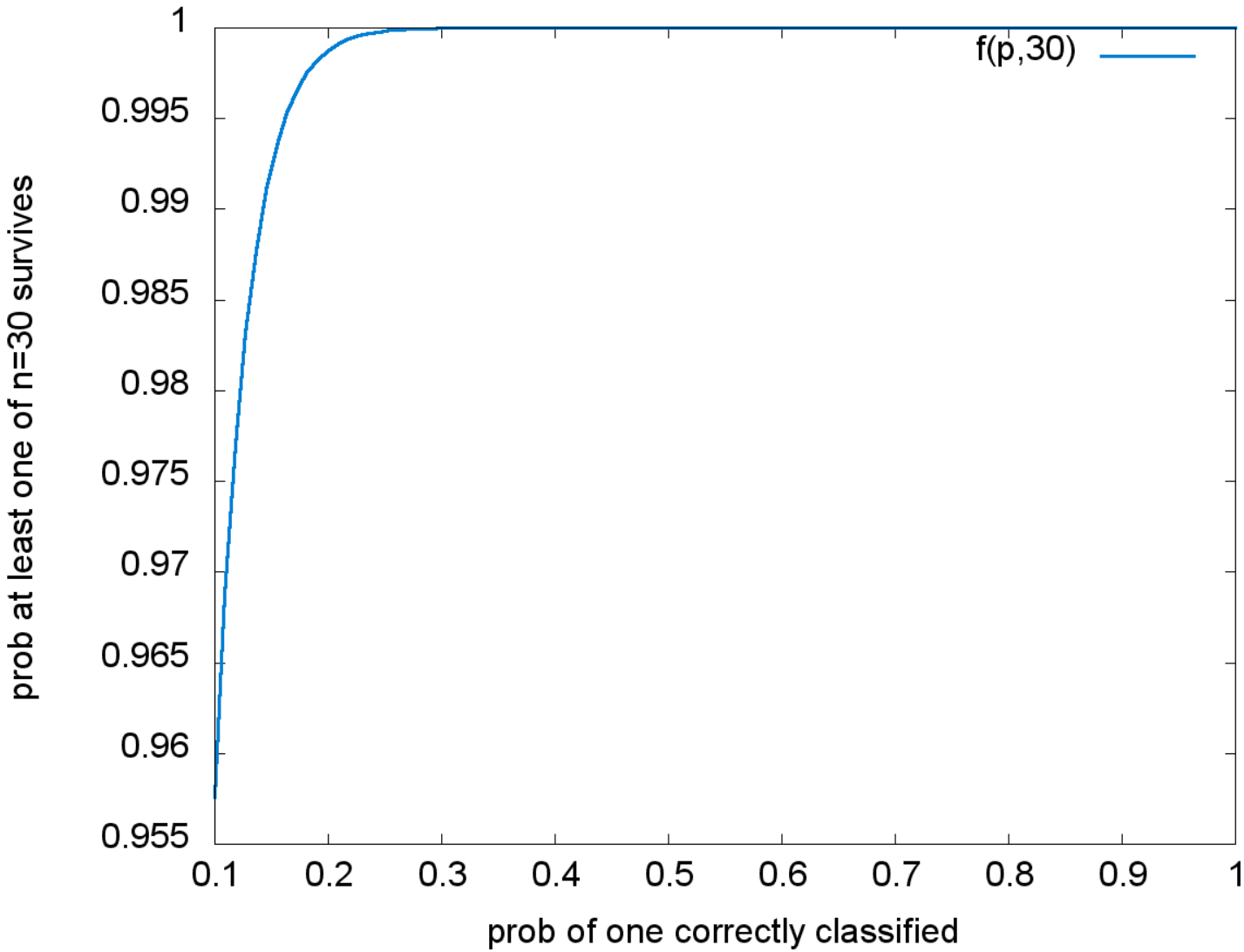
KNOWLEDGE MEDIA
KMi
INSTITUTE

Partial fingerprint block



KNOWLEDGE MEDIA INSTITUTE

Redundancy
Probability of matching



The Open University

KNOWLEDGE MEDIA KMi INSTITUTE

Exercise: fingerprint block probabilities

Assuming bit errors are independently identically distributed at the rate of b. Show that the probability $p(k, b)$ of having no more than k bit errors in one sub-fingerprint is

$$p(k, b) = \sum_{i=0}^{k} \binom{32}{i} (1-b)^{32-i} b^i.$$

Show that the probability that among 256 sub-fingerprints at least one survives with no more than k bit errors is given by

$$1 - (1 - p(k, b))^{256}.$$

Verify, using above formulas, the following claim: Even though a high bit error rate of $b = 0.3$ causes the probability $p(4)$ that no more than 4 bits were flipped to drop under 2%, it is the case that when you look at 256 sub-fingerprints, at least one of them will have no more than 4 bit errors with more than 99% probability.

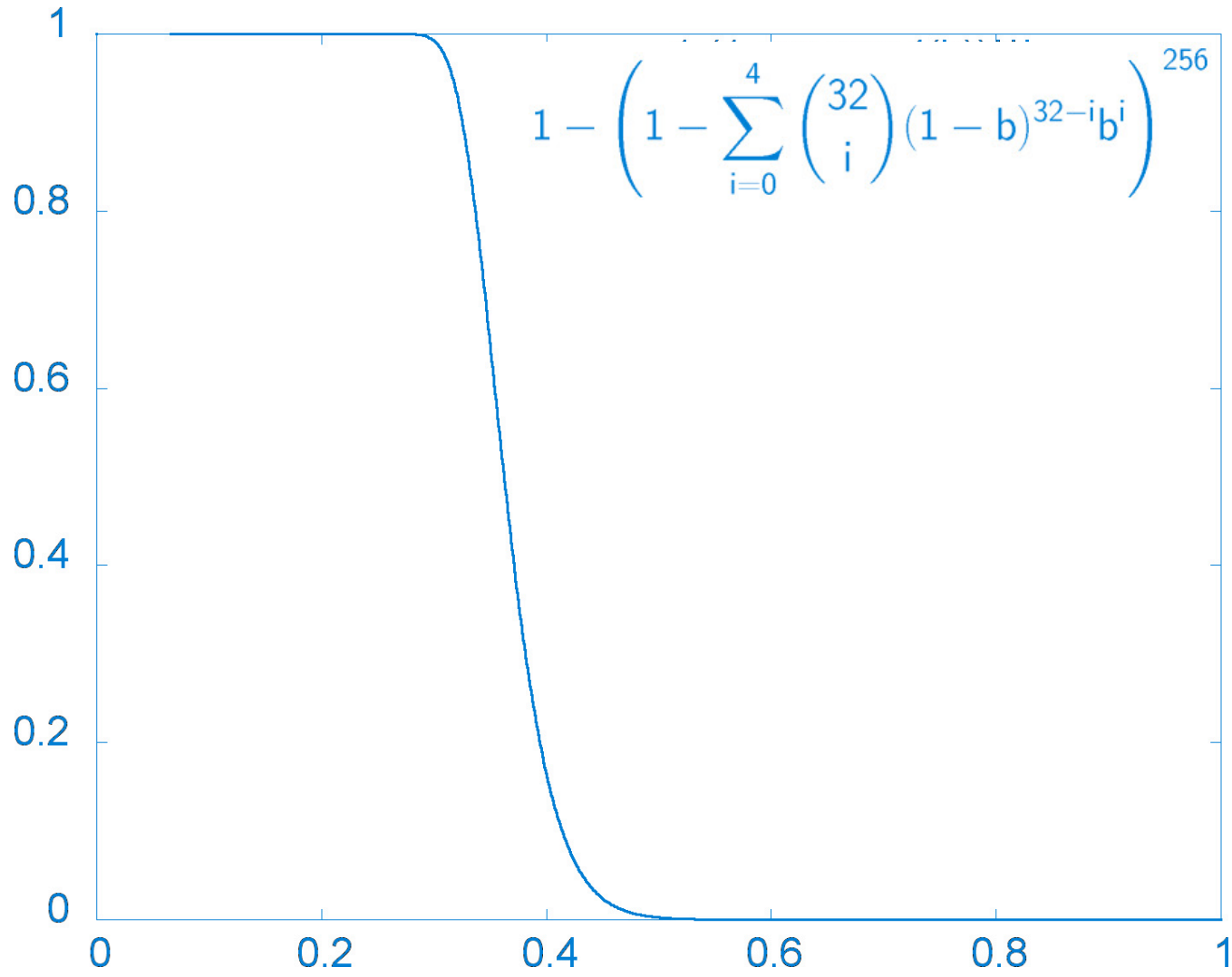
KNOWLEDGE MEDIA
KMi
INSTITUTE

Exercise: fingerprint block false positives

Assuming that the fingerprint block extraction process yields random, independent and identically distributed bits, what is the probability that a randomly modified fingerprint block matches a *different* random block in the database that consists of, say, 10^{11} overlapping fingerprint blocks (4 million songs with around 5 minutes each)? The bit error rate for the random modification is assumed to be 35%.

KNOWLEDGE MEDIA KMi INSTITUTE

Probability of at least one sub-fingerprint surviving with no more than 4 errors



The Open University

KNOWLEDGE MEDIA
KMi
INSTITUTE

How audio matching with sub-fingerprinting works

The Open University

KNOWLEDGE MEDIA INSTITUTE

Quantisation through locality sensitive hashing (LSH)

$$h^i : \mathbb{R}^d \rightarrow \mathbb{Z}$$

$$v \mapsto h^i(v) = \left\lfloor \frac{a^i v + b^i}{w} \right\rfloor$$

$a^i \in \mathbb{R}^d$ is a random Gaussian-distributed vector

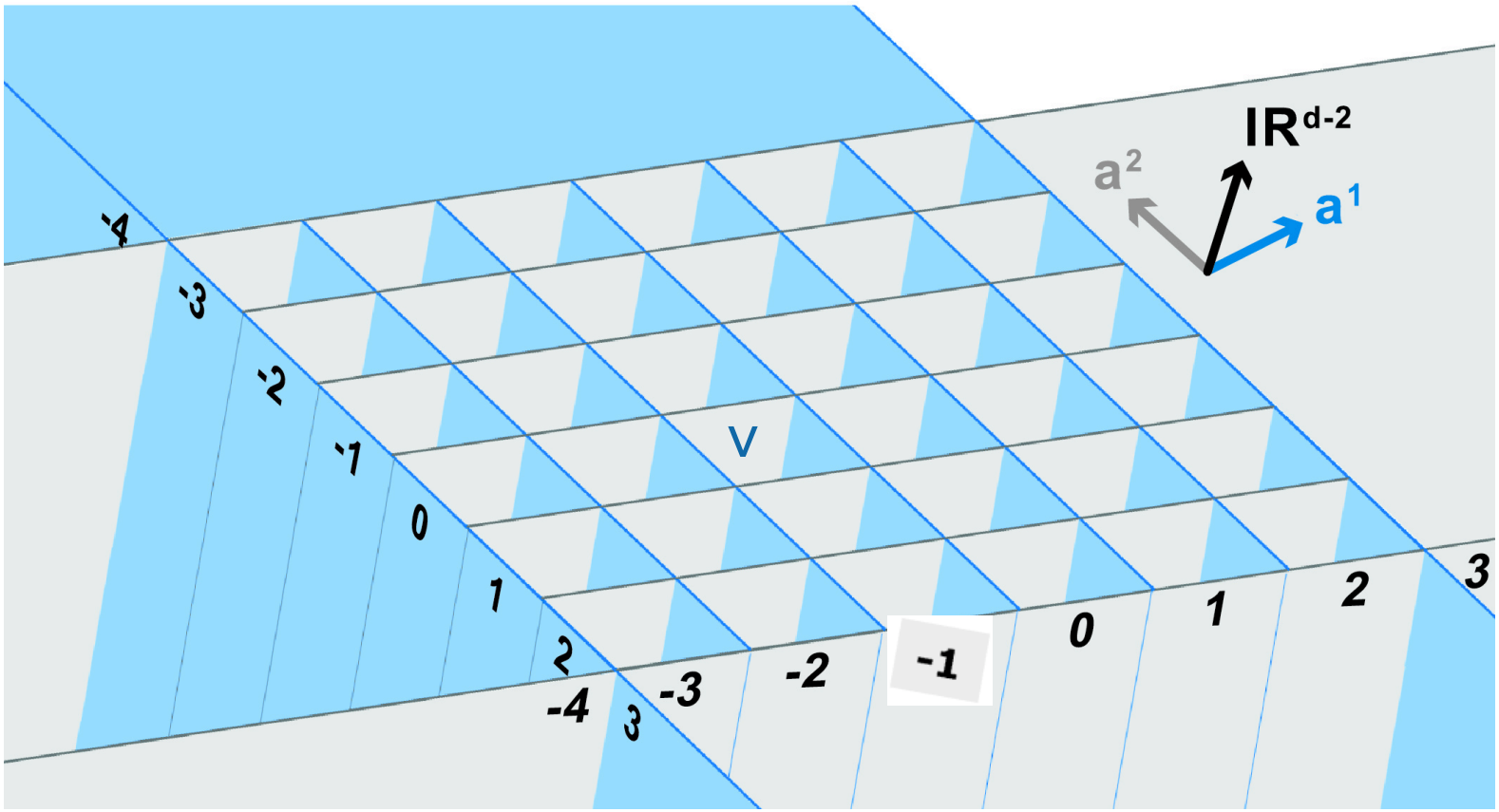
$w \in \mathbb{R}^+$ is a constant

$b^i \in [0, w)$ is a random number

$h(v) = (h^1(v), h(^2(v), \ldots, h^k(v))$ is the LSH hash vector.

KNOWLEDGE MEDIA KMi INSTITUTE

Quantisation LSH hashes



Vector v: h(v) = (-1, 0)

The Open University

Redundancy is key

Use L independent hash vectors of k components each both for the query and for each multimedia object.

Database elements that match at least m out of L times are candidates for nearest neighbours.

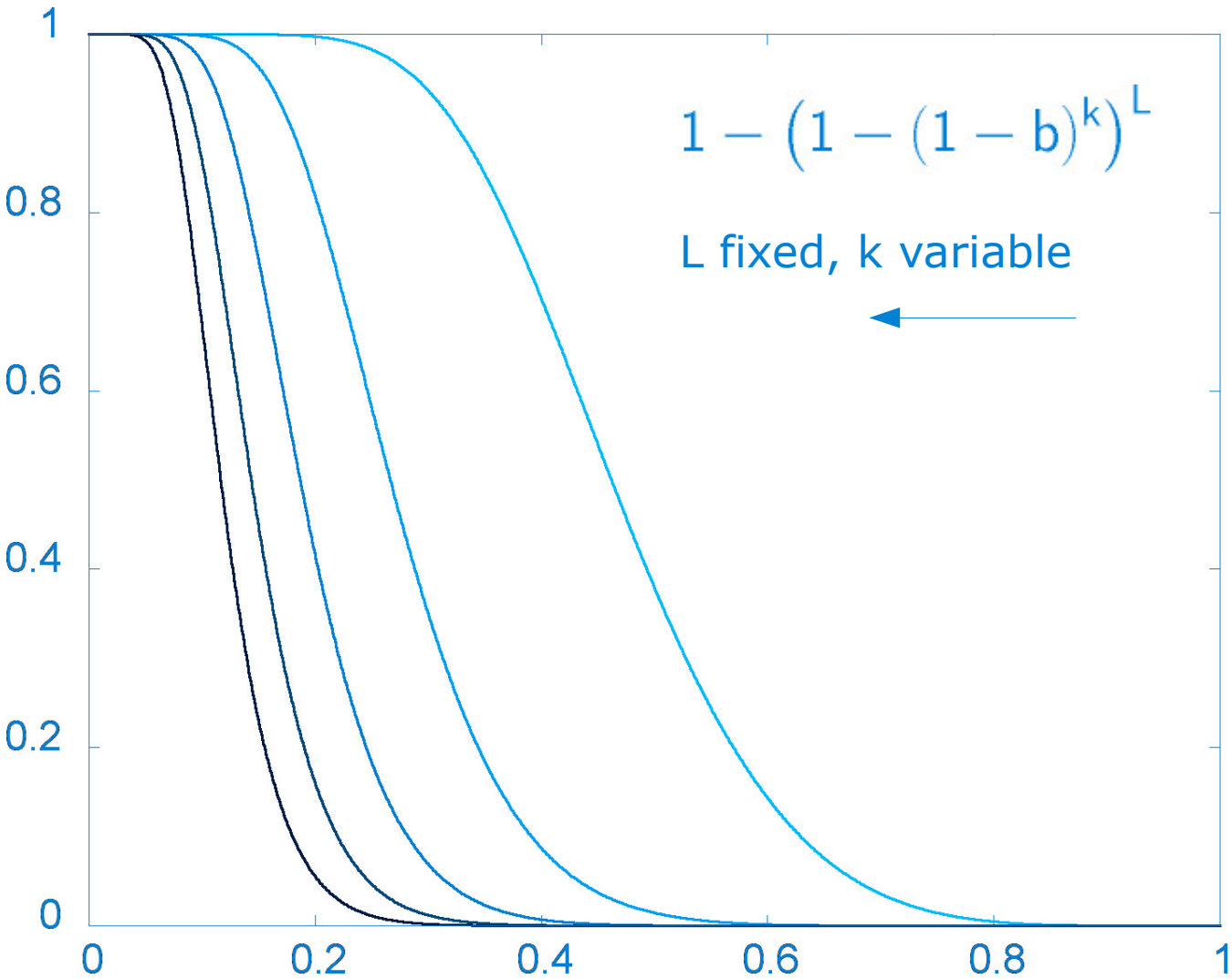
Chose w, k and L (wisely) at runtime

- w determines granularity of bins, ie, # of bits for $h^i(v)$

- k and L determine probability of matching

KNOWLEDGE MEDIA INSTITUTE

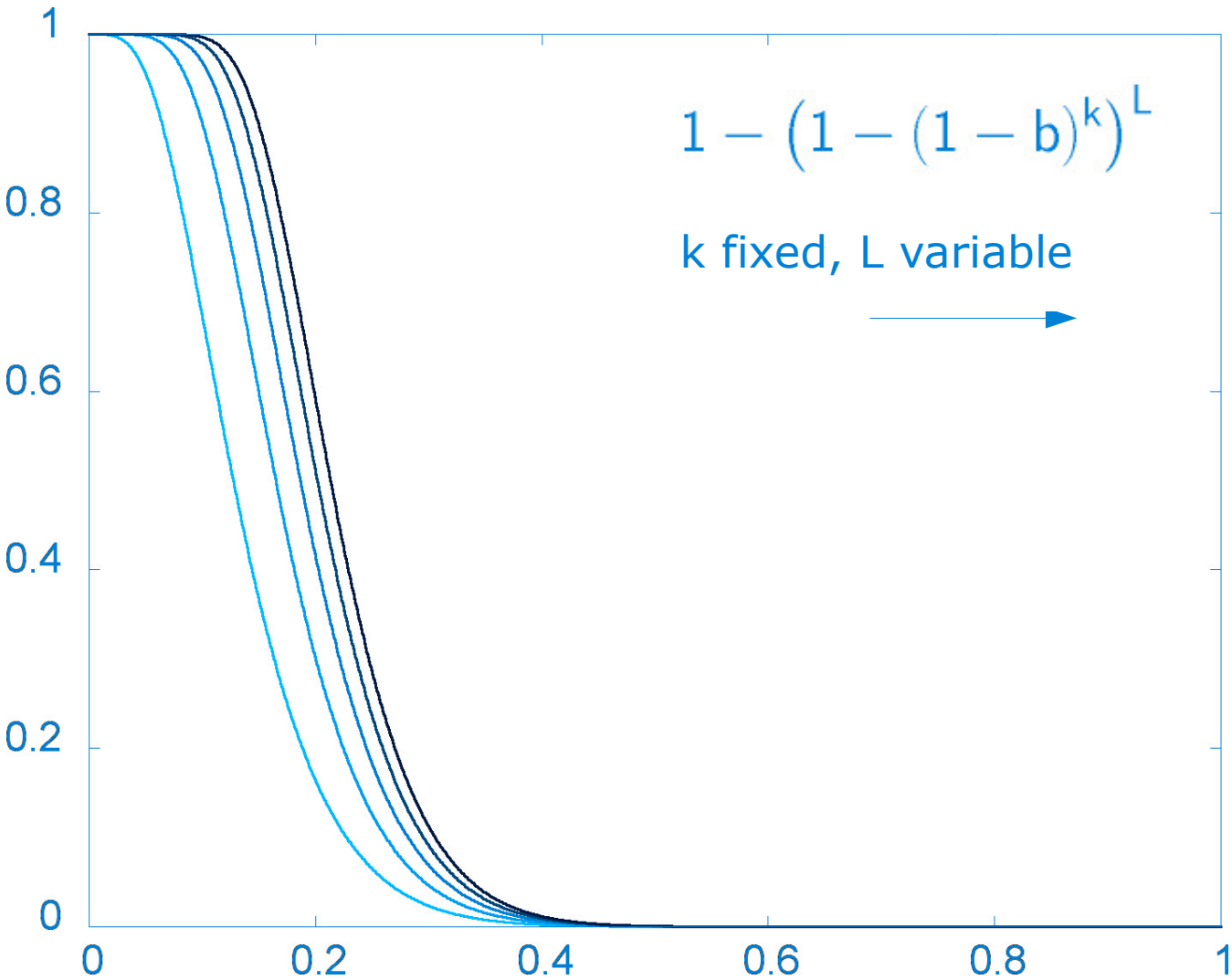
Prob(min 1 match out of L)



The Open University

KNOWLEDGE MEDIA INSTITUTE

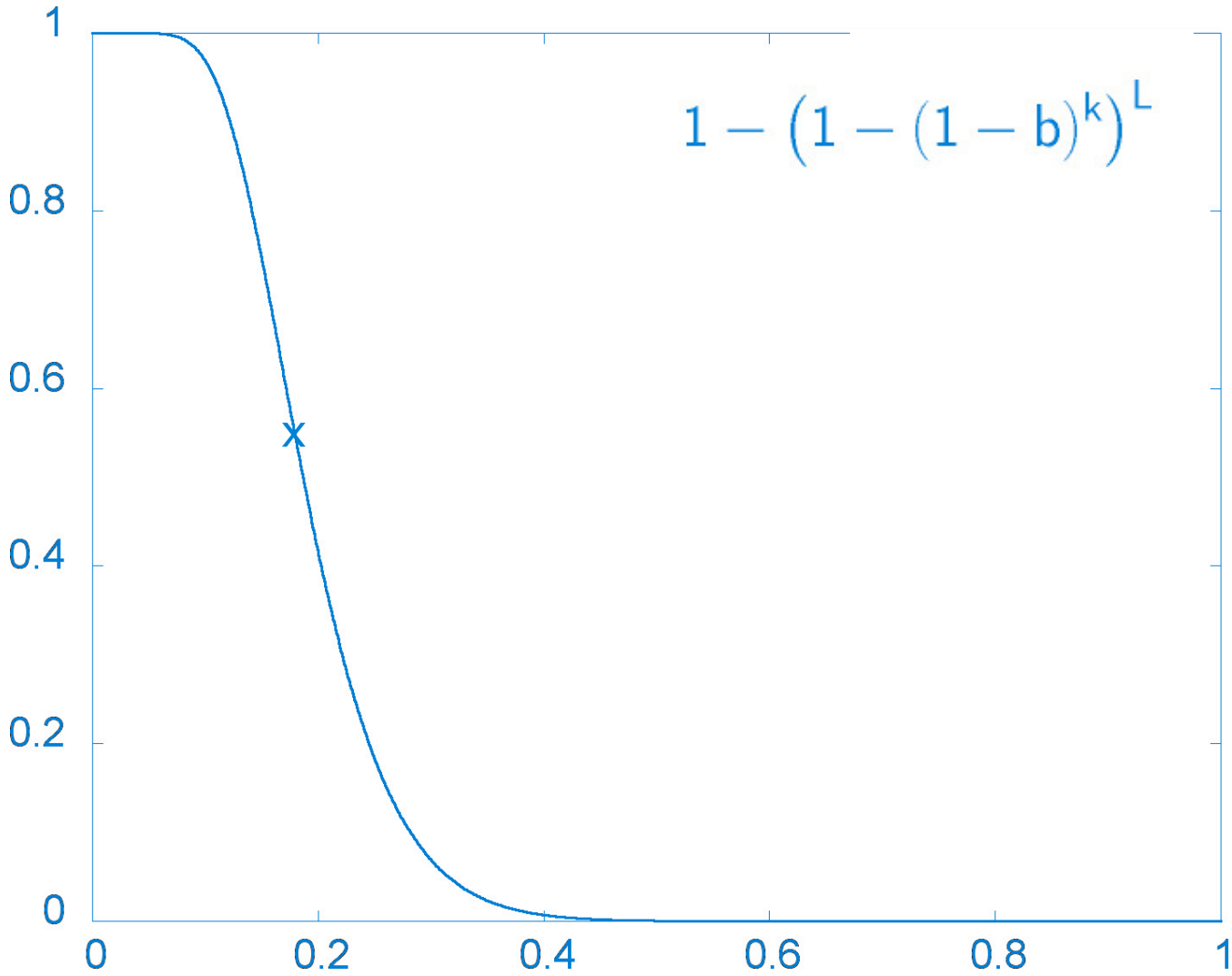
Prob(min 1 match out of L)



The Open University

KNOWLEDGE MEDIA INSTITUTE

Exercise: compute inflection point



Min hash
Estimate discrete set overlap

$$\mathrm{sim}(A_i, A_j) = \frac{|A_i \cap A_j|}{|A_i \cup A_j|}$$

KNOWLEDGE MEDIA INSTITUTE

An example
4 documents

D_1 = Humpty Dumpty sat on a wall,

D_2 = Humpty Dumpty had a great fall.

D_3 = All the King's horses, And all the King's men

D_4 = Couldn't put Humpty together again!

Surrogate docs after stop word removal and stemming

A_1 = {humpty, dumpty, sat, wall}

A_2 = {humpty, dumpty, great, fall}

A_3 = {all, king, horse, men}

A_4 = {put, humpty, together, again}

KNOWLEDGE MEDIA KMi INSTITUTE

Equivalent term-document matrix

	A_1	A_2	A_3	A_4
humpty	1	1	0	1
dumpty	1	1	0	0
sat	1	0	0	0
wall	1	0	0	0
great	0	1	0	0
fall	0	1	0	0
all	0	0	1	0
king	0	0	1	0
horse	0	0	1	0
men	0	0	1	0
put	0	0	0	1
together	0	0	0	1
again	0	0	0	1

KNOWLEDGE MEDIA KMi INSTITUTE

Similarity between two docs

$$sim(A_i, A_j) = \frac{c_{11}}{c_{11}+c_{10}+c_{01}}$$

c_{xy} = number of (x,y) rows

Important observation
c_{00} is unused!

	A_2	A_4
humpty	1	1
dumpty	1	0
sat	0	0
wall	0	0
great	1	0
fall	1	0
all	0	0
king	0	0
horse	0	0
men	0	0
put	0	1
together	0	1
again	0	1

KNOWLEDGE MEDIA INSTITUTE

Estimation of similarity through random permutations

$\pi_1 =$ (dumpty, men, again, put, great, humpty, wall, horse, king, sat, fall, together, all)

$\pi_2 =$ (fall, put, all, again, dumpty, sat, men, great, wall, king, horse, humpty, together)

$\pi_3 =$ (horse, dumpty, wall, humpty, great, again, sat, all, men, together, put, king, fall)

$\pi_4 =$ (king, humpty, men, together, great, fall, horse, all, dumpty, wall, sat, again, put)

KNOWLEDGE MEDIA KMi INSTITUTE

Surrogate documents form random permutations

Keep first occurring word of A_i in π_j for dense surrogate representation

	A_1	A_2	A_3	A_4
π_1	dumpty	dumpty	men	again
π_2	dumpty	fall	all	put
π_3	dumpty	dumpty	horse	humpty
π_4	humpty	humpty	king	humpty

KNOWLEDGE MEDIA INSTITUTE

Surrogate documents form random permutations

	A_2	A_4
π_1	dumpty	again
π_2	fall	put
π_3	dumpty	humpty
π_4	humpty	humpty

Estimate sim(A_2,A_4) = 1/4

(proportion of co-inciding words)

KNOWLEDGE MEDIA KMi INSTITUTE

Exercise: estimate sim(A_4, A_5) with min hash

Mice are dancing in a round,
On a bench a cat is sleeping.
"Hush, you mice, don't make such noise
Or you'll wake up Vaska Cat
Vaska Cat will jump and leap
And will spoil and break your round".

A_1 = {mice, danc, round}

A_2 = {bench, cat, sleep}

A_3 = {hush, mice, nois}

A_4 = {wake, vaska, cat}

A_5 = {vaska, cat, jump, leap}

A_6 = {spoil, break, round}

KNOWLEDGE MEDIA KMi INSTITUTE

Exercise: estimate sim(A_4, A_5) with min hash

A_4 = {wake, vaska, cat}

A_5 = {vaska, cat, jump, leap}

π_1 = (bench, break, cat, danc, hush, jump, leap, mice, nois, round, sleep, spoil, vaska, wake)

π_2 = (cat, vaska, wake, bench, danc, nois, leap, jump, sleep, round, mice, break, spoil, hush)

π_3 = (hush, break, vaska, nois, jump, mice, sleep, spoil, wake, round, leap, bench, danc, cat)

π_4 = (spoil, bench, cat, wake, nois, leap, danc, sleep, jump, round, mice, hush, break, vaska)

π_5 = (vaska, danc, leap, break, round, nois, spoil, hush, wake, jump, sleep, cat, benc,h mice)

π_6 = (round, vaska, danc, wake, spoil, hush, sleep, leap, cat, nois, mice, break, bench, jump)

π_7 = (danc, break, vaska, spoil, sleep, wake, round, bench, leap, hush, jump, mice, nois, cat)

π_8 = (jump, nois, break, danc, round, leap, hush, cat, sleep, vaska, spoil, mice, bench, wake)

SIFT

Scale Invariant Feature Transform

"distinctive invariant image features that can be used to perform reliable matching between different views of an object or scene."

Invariant to image scale and rotation.

Robust to substantial range of affine distortion, changes in 3D viewpoint, addition of noise and change in illumination.

[Lowe, D.G. (2004). Distinctive Image Features from Scale-Invariant Keypoints. International Journal of Computer Vision, 60, 2, pp. 91-110.]

SIFT Implementation

For a given image:

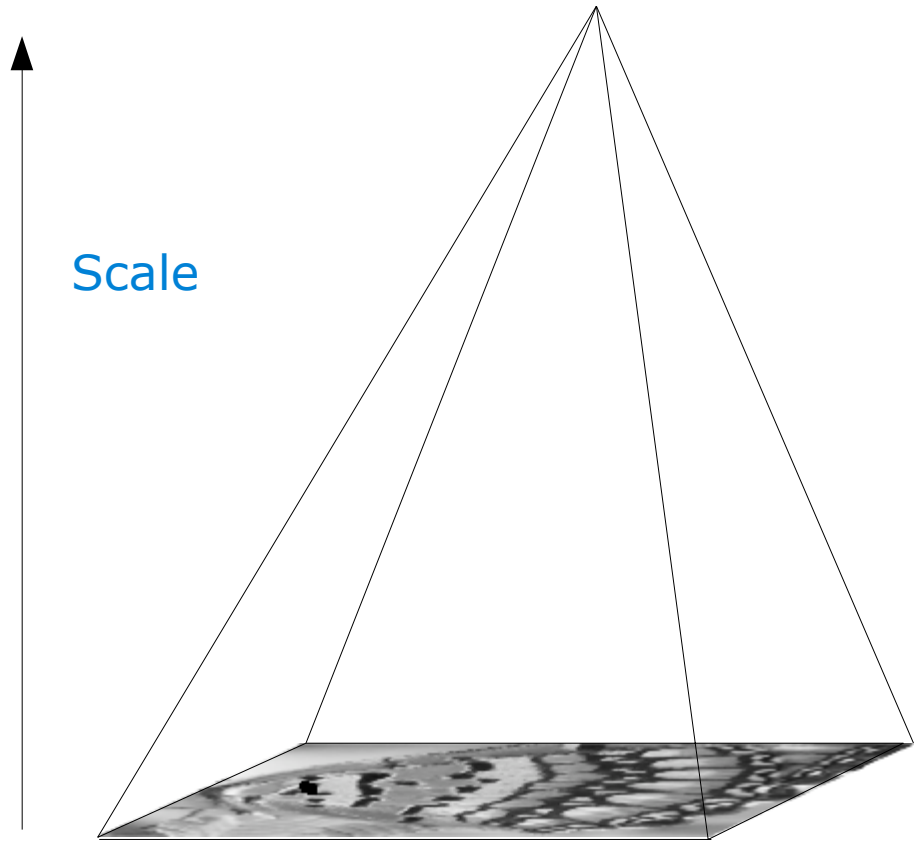
Detect scale space extrema

Localise candidate keypoints

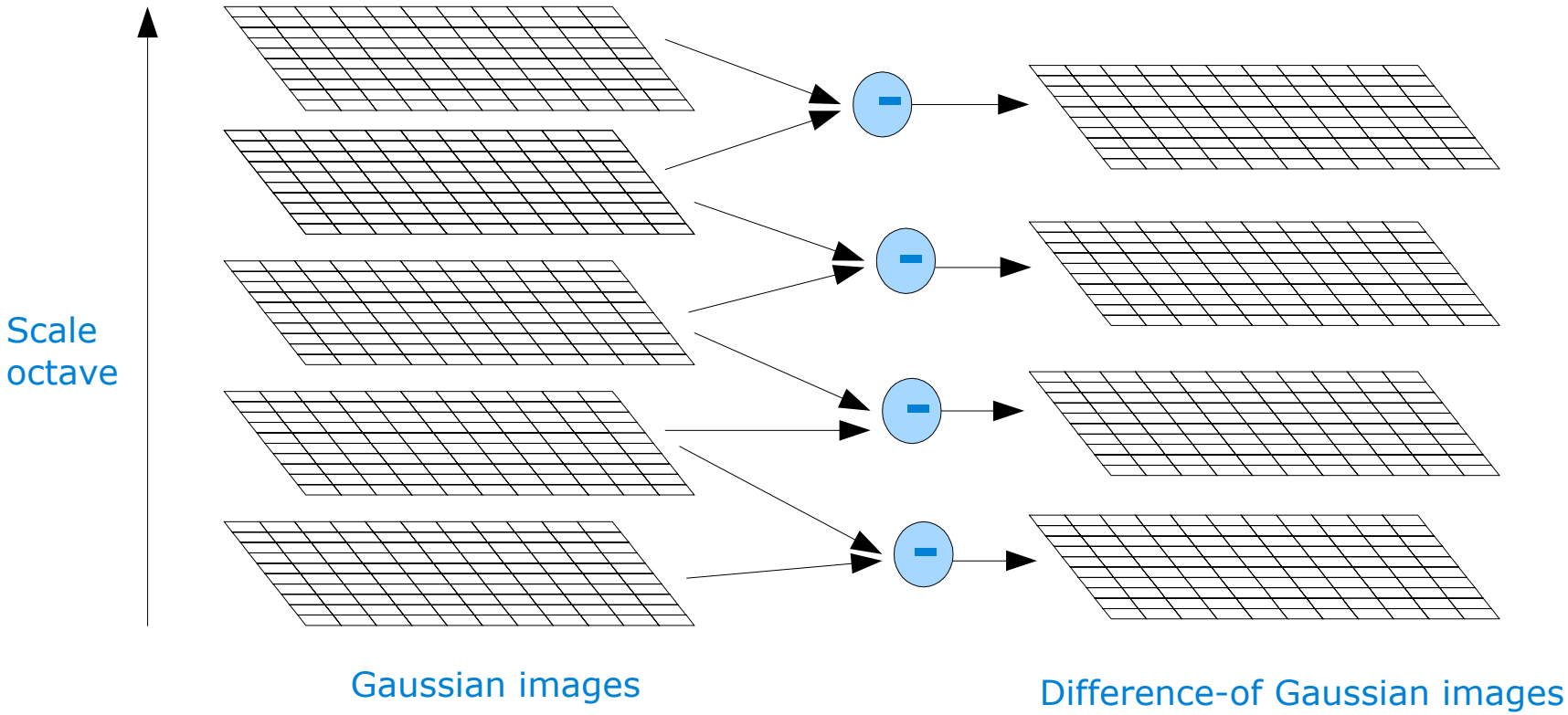
Assign an orientation to each keypoint

Produce keypoint descriptor

A scale space visualisation



Difference of Gaussian image creation



KNOWLEDGE MEDIA
KMi
INSTITUTE

Gaussian blur illustration

The Open University

KNOWLEDGE MEDIA INSTITUTE

Difference of Gaussian illustration

The Open University

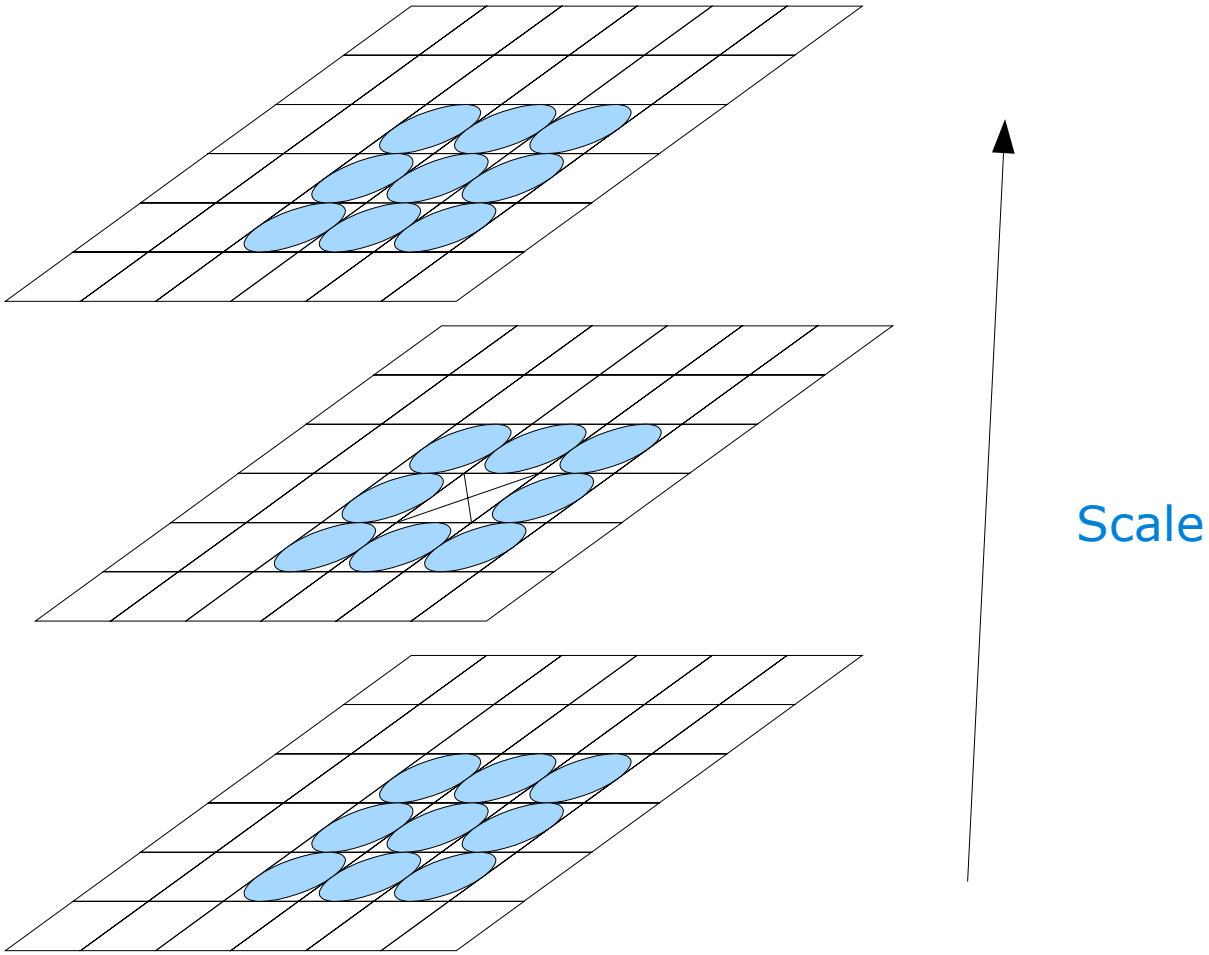
KNOWLEDGE MEDIA KMi INSTITUTE

The SIFT keypoint system

Once the Difference of Gaussian images have been generated:

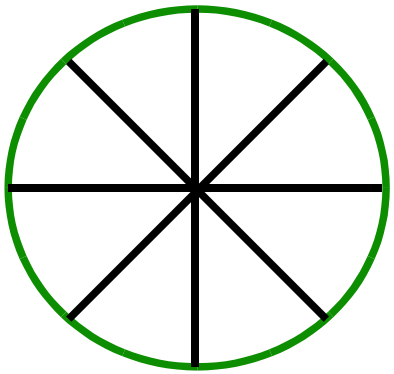
- Each pixel in the images is compared to 8 neighbours at same scale.
- Also compared to 9 corresponding neighbours in scale above and 9 corresponding neighbours in the scale below.
- Each pixel is compared to 26 neighbouring pixels in 3x3 regions across scales, as it is not compared to itself at the current scale.
- A pixel is selected as a SIFT keypoint only either if its intensity value is extreme.

Pixel neighbourhood comparison



KNOWLEDGE MEDIA INSTITUTE

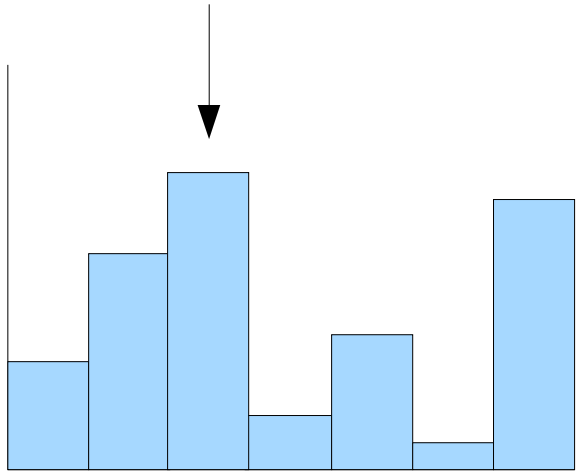
Orientation assignment

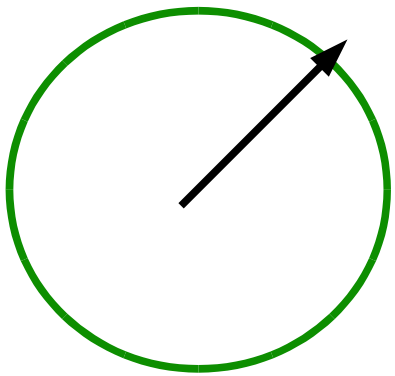


Orientation histogram with 36 bins – one pe
10 degrees.

Each sample weighted by gradient magnitude and Gaussian window.

Canonical orientation at peak of Smoothed histogram.





Where two or more orientations are detected
keypoints created for each orientation.

The Open University

The SIFT keypoint descriptor

We now have location, scale and orientation for each SIFT keypoint ("keypoint frame").

→ descriptor for local image region is required.

Must be as invariant as possible to changes in illumination and 3D viewpoint.

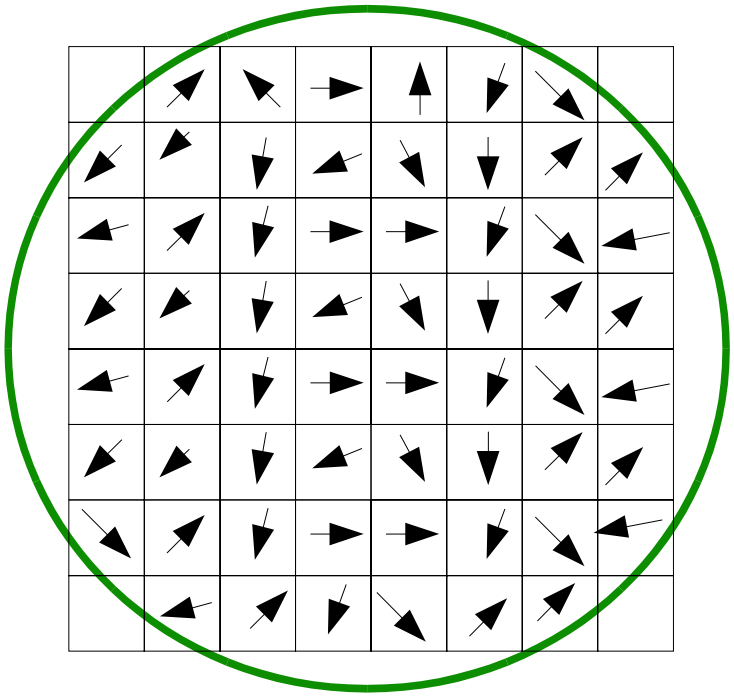
Set of orientation histograms are computed on 4x4 pixel areas.

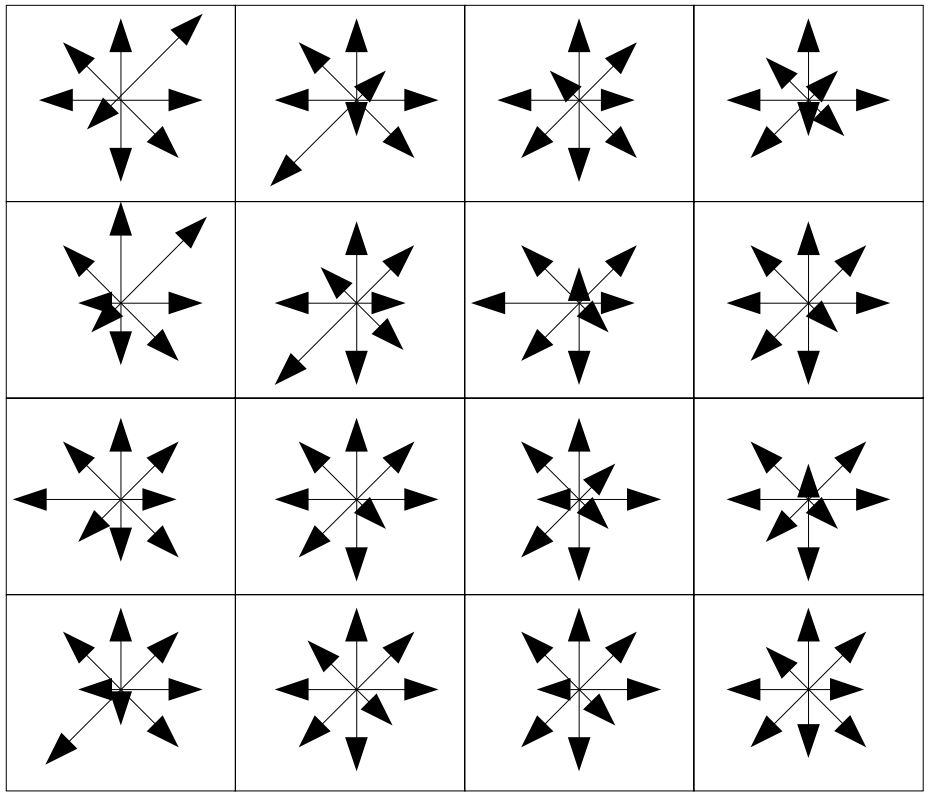
Each gradient histogram contains 8 bins and each descriptor contains an array of 4 histograms.

→ SIFT descriptor as 128 (4 x 4 x 8) element histogram

KNOWLEDGE MEDIA INSTITUTE

Visualising the keypoint descriptor





KNOWLEDGE MEDIA
KMi
INSTITUTE

Example SIFT keypoints

KNOWLEDGE MEDIA INSTITUTE

SURF

- Alternative to SIFT - “Speeded Up Robust Features”
- High dimensionality of SIFT descriptor makes it costly to compute and slow to match.
- Goal is to speed up the detection and description process for image features.
- Similar to SIFT but the authors claim better and more robust performance.

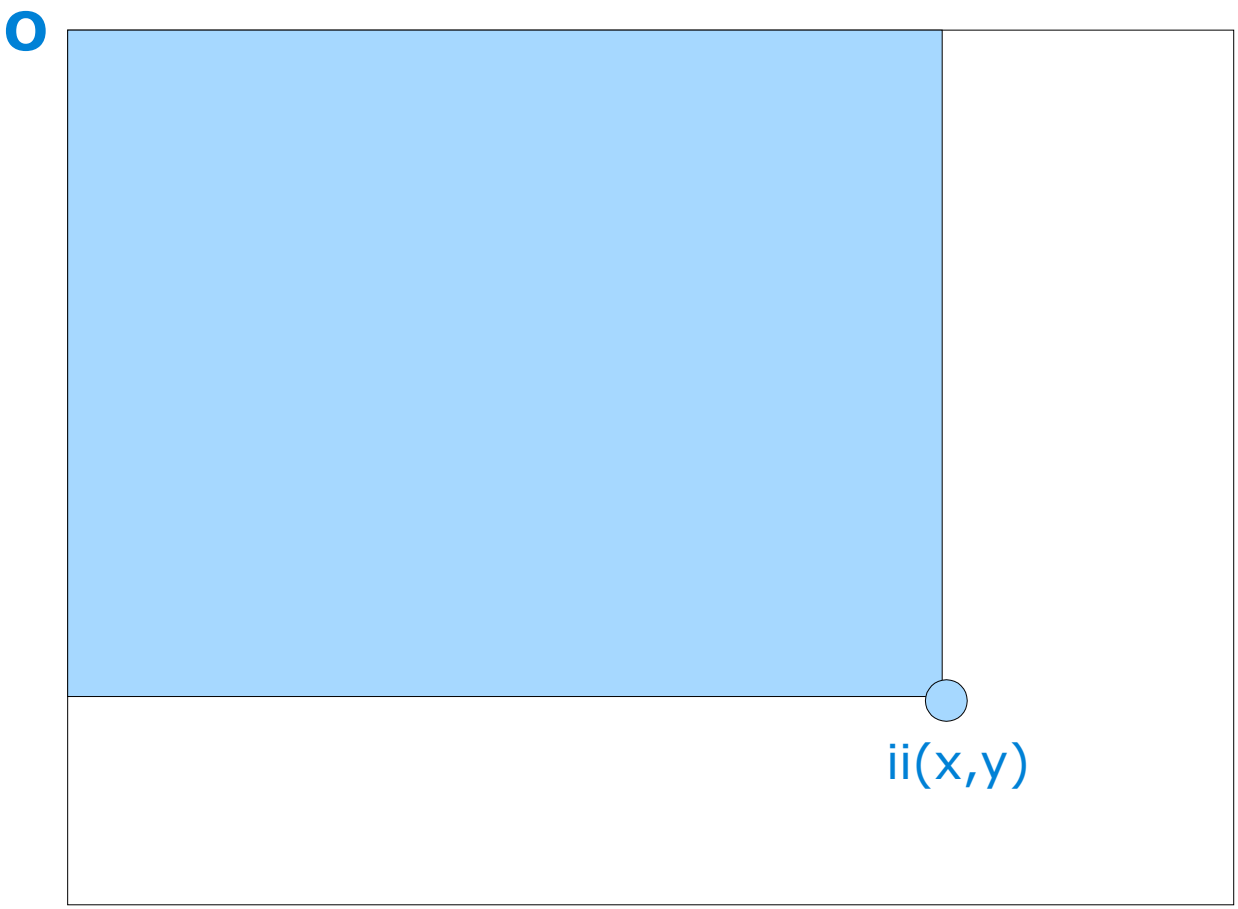
[Bay and Tuytelaars and Van Gool, H and T and L. Speeded Up Robust Features. In ECCV 2006, pp. 404-417.]

The Open University

SURF

- Uses integral images (similar to summed area tables) to quickly compute box-type convolution filters.
- Integral image = the sum of the intensities of all pixels contained in the rectangle defined by the pixel of interest and the origin.

Integral image theory



The value of the integral image at point (x,y) = the sum of all pixels above
and to the left.

Integral image theory

$$ii(x,y) = \sum_{x' \leq x, y' \leq y} i(x',y')$$

Using the following pair of recurrences:

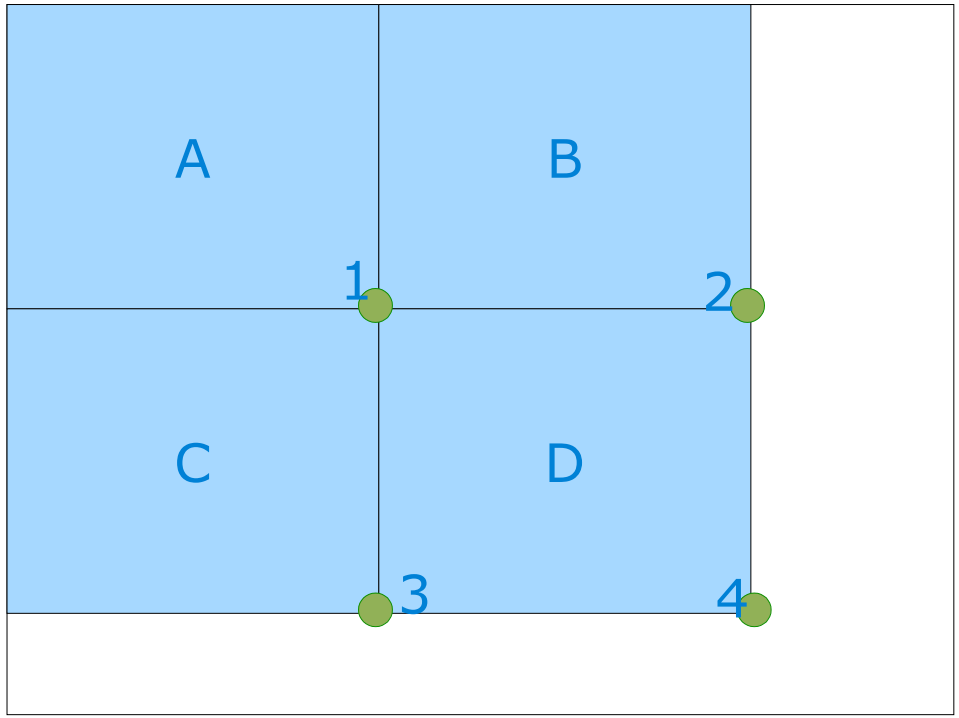
$$s(x,y) = s(x,y-1) + i(x,y)$$

$$ii(x,y) = i(x-1,y) + s(x,y)$$

Where **s(x,y)** is the cumulative row sum
s(x, -1) = 0 and
ii(-1,y) = 0

the integral image can be computed in one pass over the original image

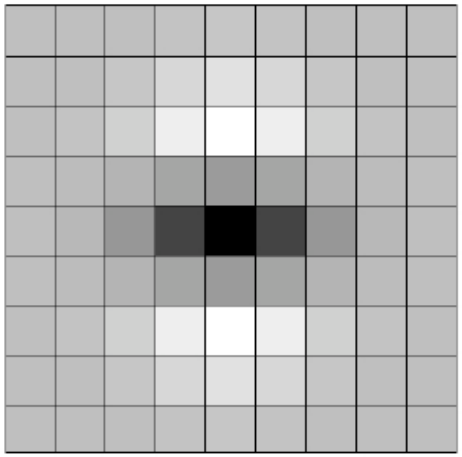
Integral image theory



Integral image at point 1 = **sum of pixels in A.**
Value at point 2 = **A+B.**
Value at point 3 = **A+C.**
Value at point 4 = **A+B+C+D.**
Sum within D can be calculated as 4 + 1 – (2 + 3).

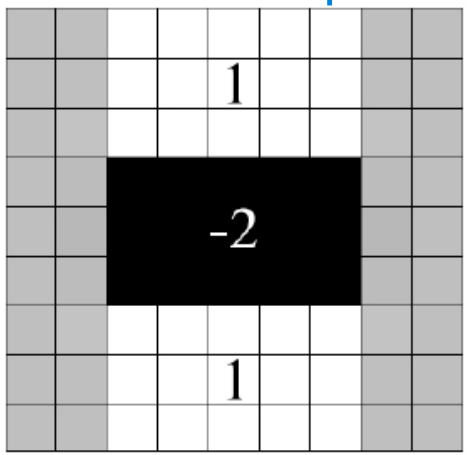
SURF detector

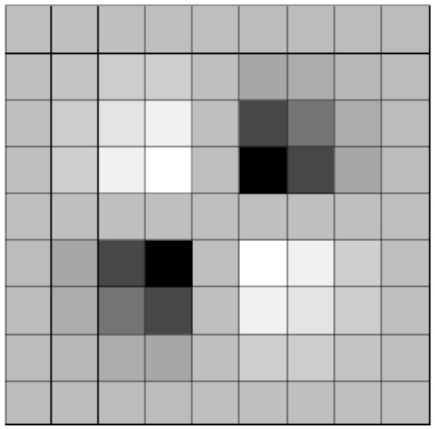
Gaussians



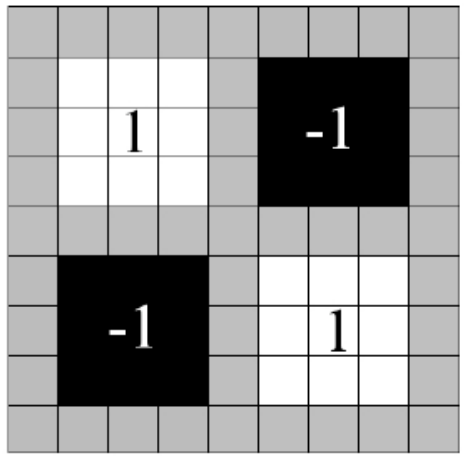
Y direction

Box filter equivalent





XY direction

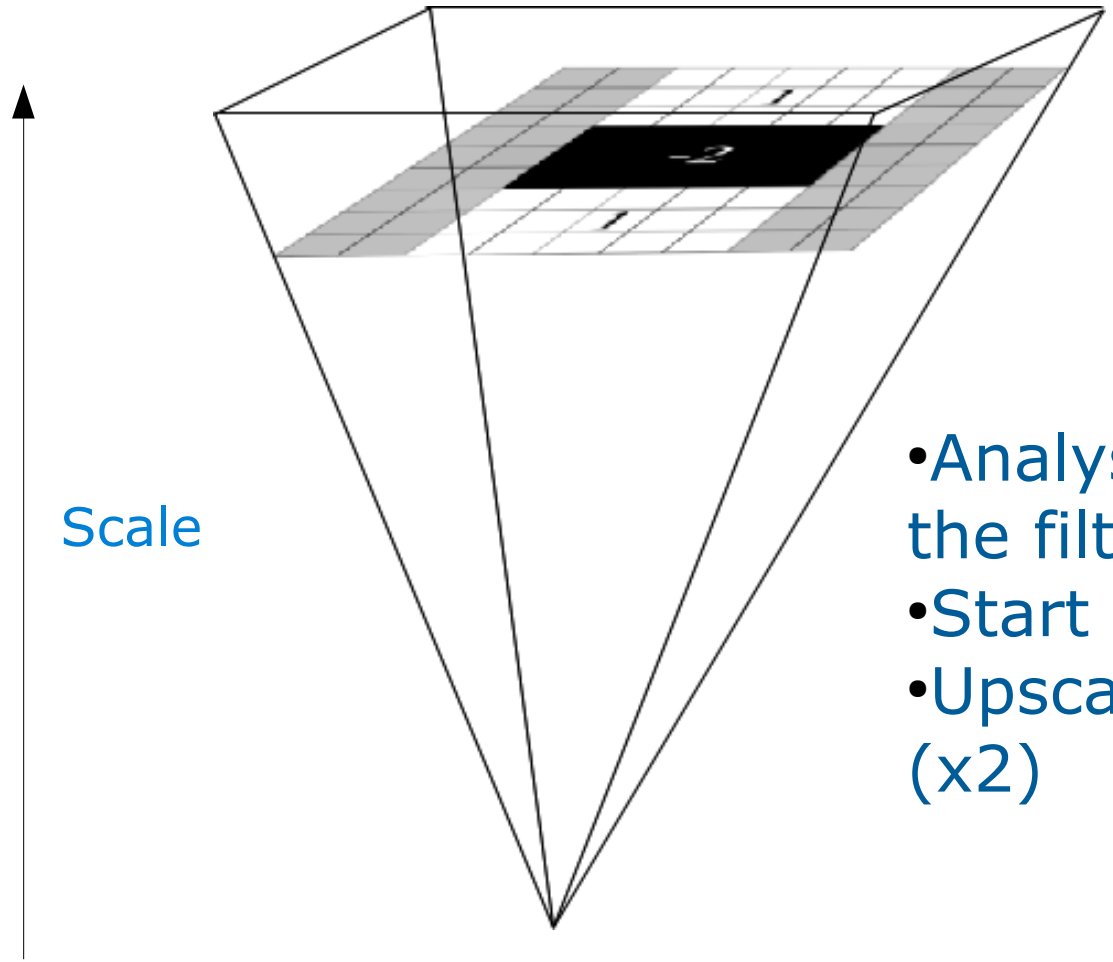


Computation time increases with filter size.

Computation time constant and Independent of filter size.

KNOWLEDGE MEDIA KMi INSTITUTE

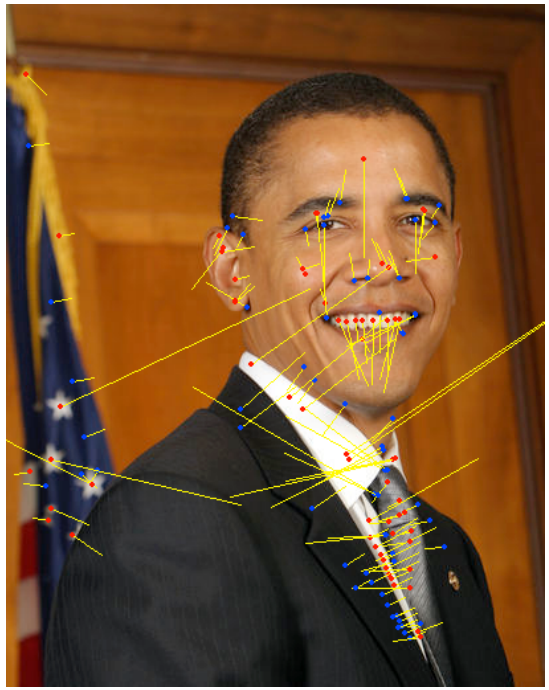
SURF Scale space



- Analyse by upscaling the filter size
- Start with 9x9
- Upscale by octaves (x2)

KNOWLEDGE MEDIA KMi INSTITUTE

SURF: some example images



KNOWLEDGE MEDIA INSTITUTE

SURF and SIFT: differentiation

SURF and SIFT both focus on the spatial distribution of gradient information.

SURF

- Is three times faster than SIFT
- Is less susceptible to noise (claimed to be!)
- Is good at handling serious image blur
- Is good at handling image rotation
- Does not handle viewpoint change or illumination change well

NB: SURF does not always outperform the original SIFT