Generic and optimized framework for multi-content analysis based on learning approaches

Q. Besnehard, C. Marchessoux & T. Kimpe

June 15th, 2010
Multitel Spring Workshop 2010

BARCO
Visibly yours

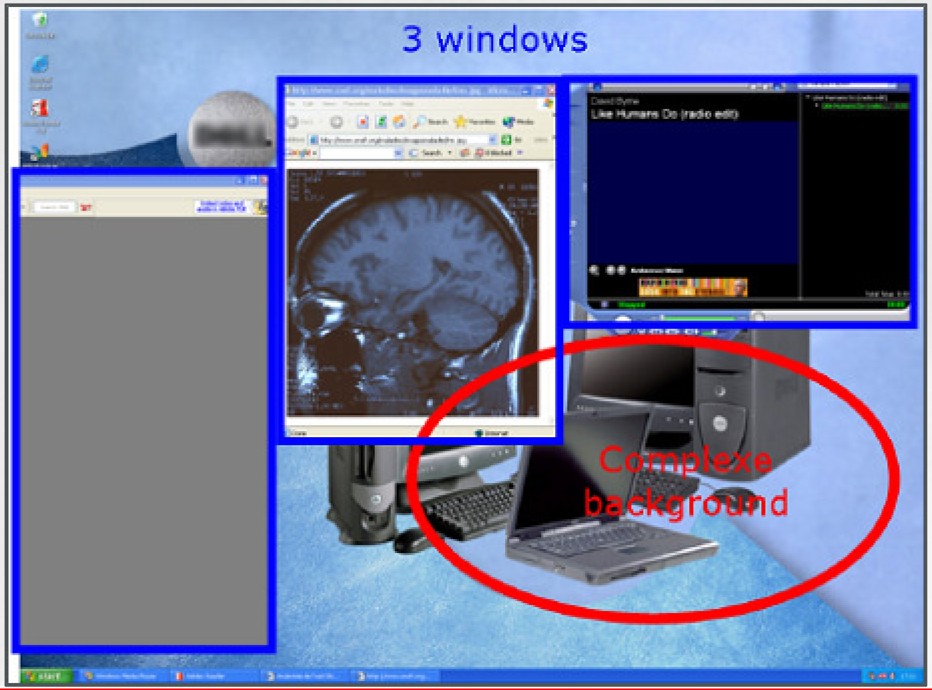
Technology and Innovation Group

- Barco Medical Imaging Division applied research group.
- Development of technical improvements for medical displays.
- New product concepts.
- Involved in projects with private and public partners.

CANTATA – Multi Content Analysis for Networked Medical Applications

- Develop algorithms to automatically classify the content of a screenshot:
  - **Without any a priori knowledge on the content**
  - **Color image, RGB 24 bits**
  - **The algorithm implemented in hardware devices should run in real time**

Example



Visibly yours

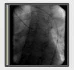
Multi Content Analysis

- Content classification:

– Gui content

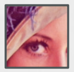
MID – Text content

– Uniform content (one color)



– Medical image content

» Digitized Screen Film Conventional Radiography (Chest, Mammography and General Radiography), Computed Radiography, Digital Radiography, Computed Tomography (CT), MRI, Angiography, ultrasonography, Positron Emission Tomography (PET), and multi modalities fusion.



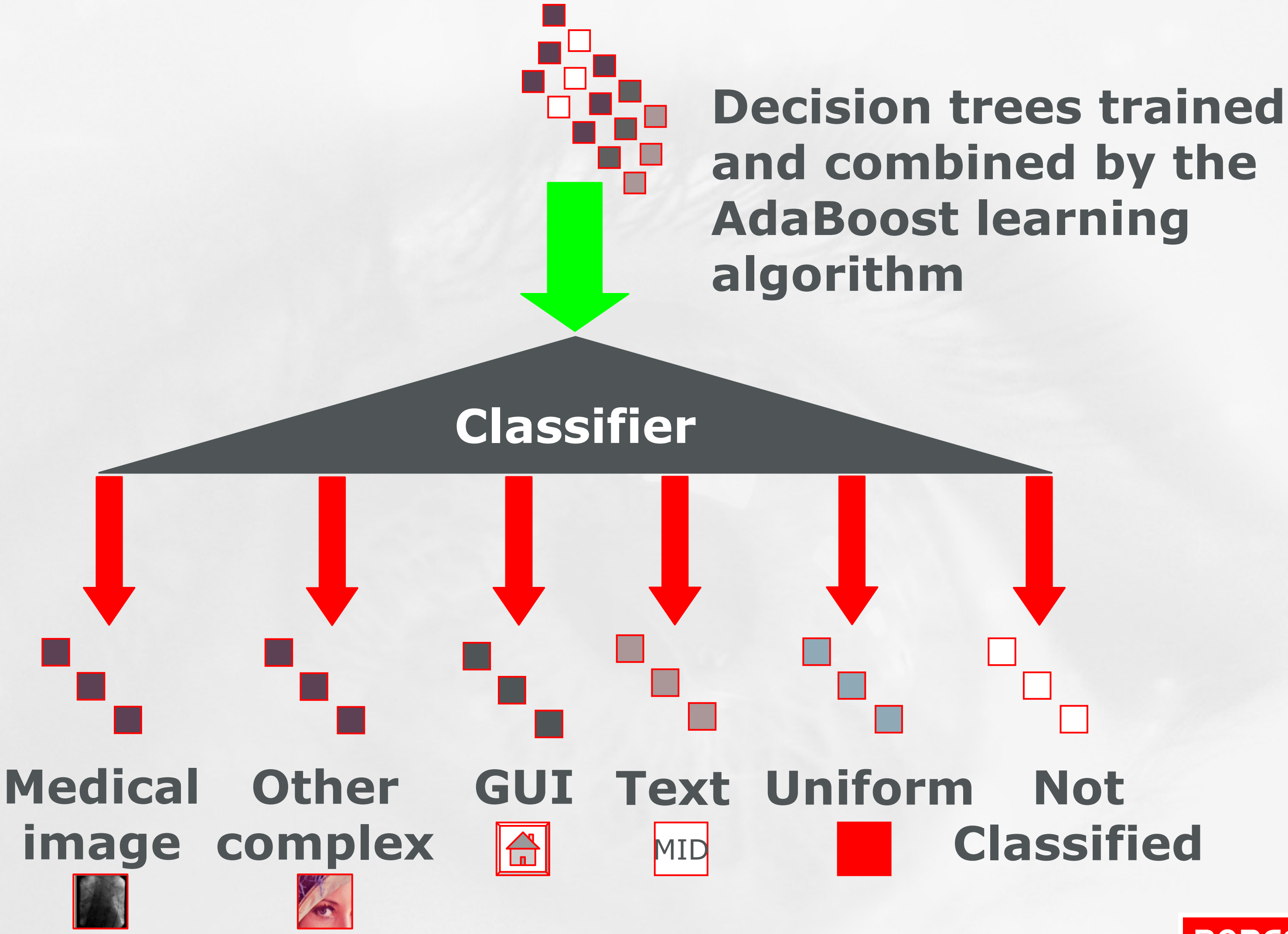
– Other complex content

BARCO
Visibly yours

MCA methodology

- Image decomposed in tiles / thumbnails:

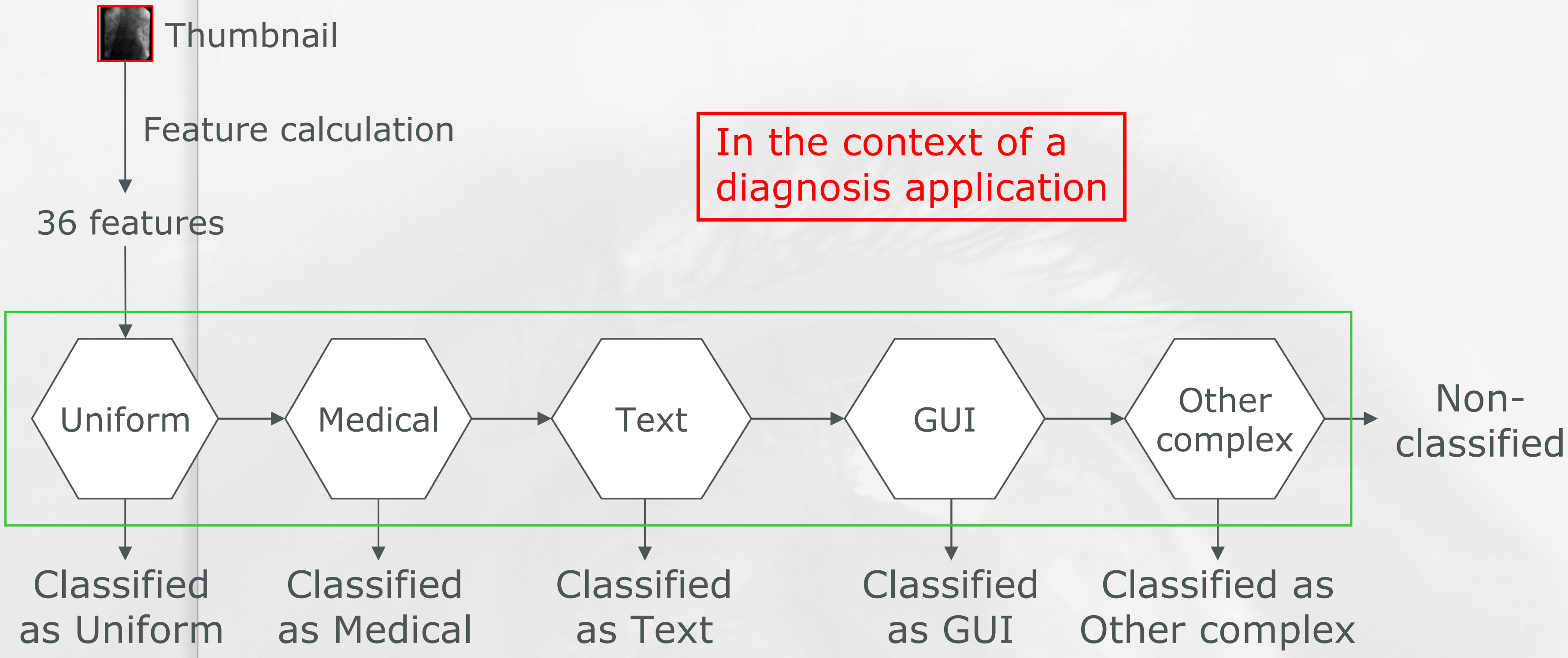
MCA: Thumbnail classifier



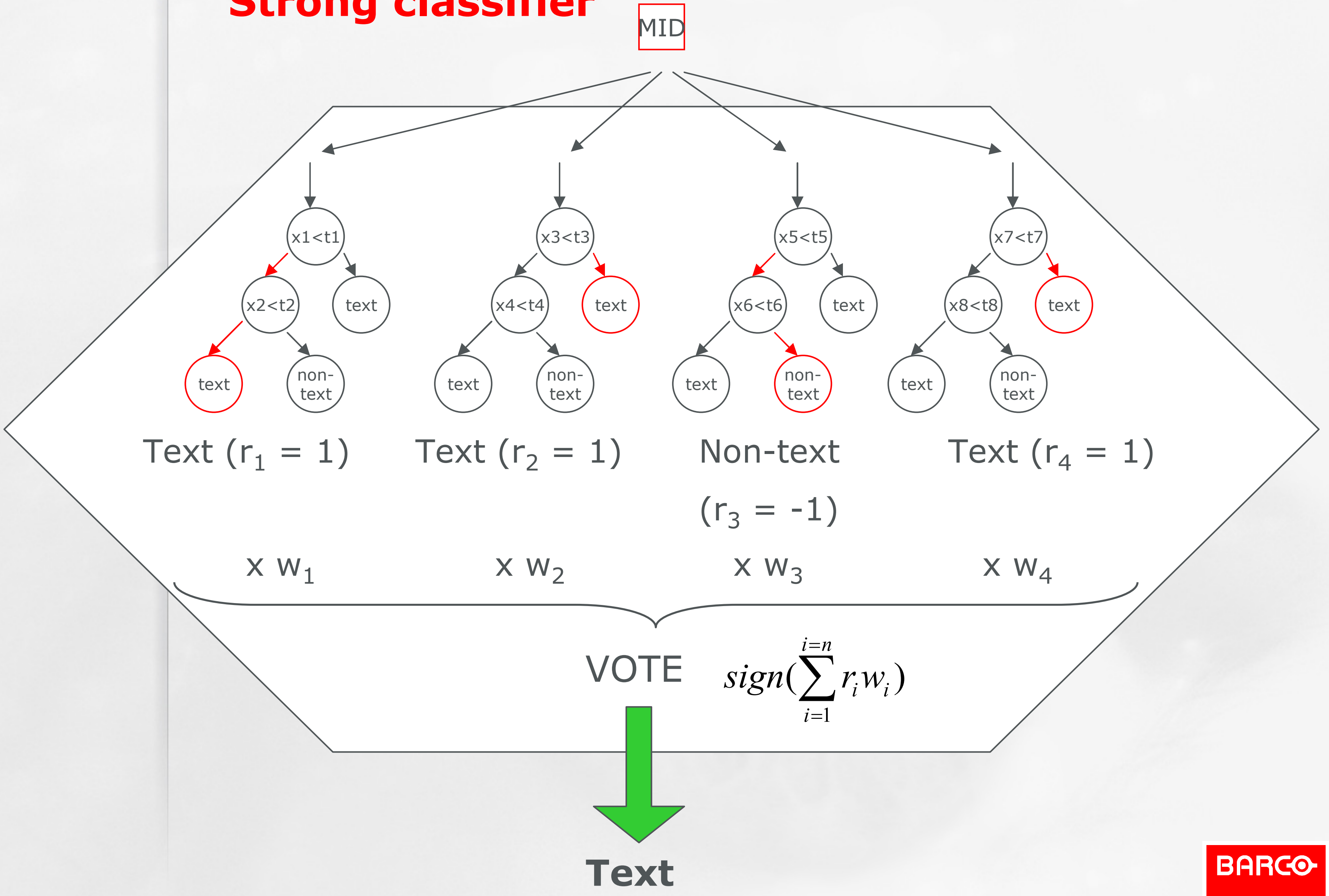
Outlines

- Basic methodology
- Implemented optimizations
- Validation, results and observed improvements

Cascade of classifiers



Strong classifier



Finding the best classifier

- Right combination of classifier training parameters
  - **Number of nodes in the decision trees.**
  - **Number of decision trees per strong classifier.**
  - **Order of strong classifiers in the cascade.**
  - **Especially important to prevent overfitting.**

- Need to run the training algorithm multiple times

- Bottlenecks
  - **Long feature calculation time.**
  - **Long decision tree training time.**
  - **Causes:**
    - Large number of features to compute and process: 81 candidates.
    - Large training dataset: thousands of images.

Feature selection

- List of relevant features to compute from the thumbnails

Feature	Number
Number of colors	1
Number of gray levels	1
Standard deviation of the hue component	1
Standard deviation	4
Derivatives gradient counts	16
Sobel filters gradient counts	16
Laplace operator gradient counts	8
Kirsh operators gradient counts	32
Absolute DCT coefficient sums	2
Total number	**81**

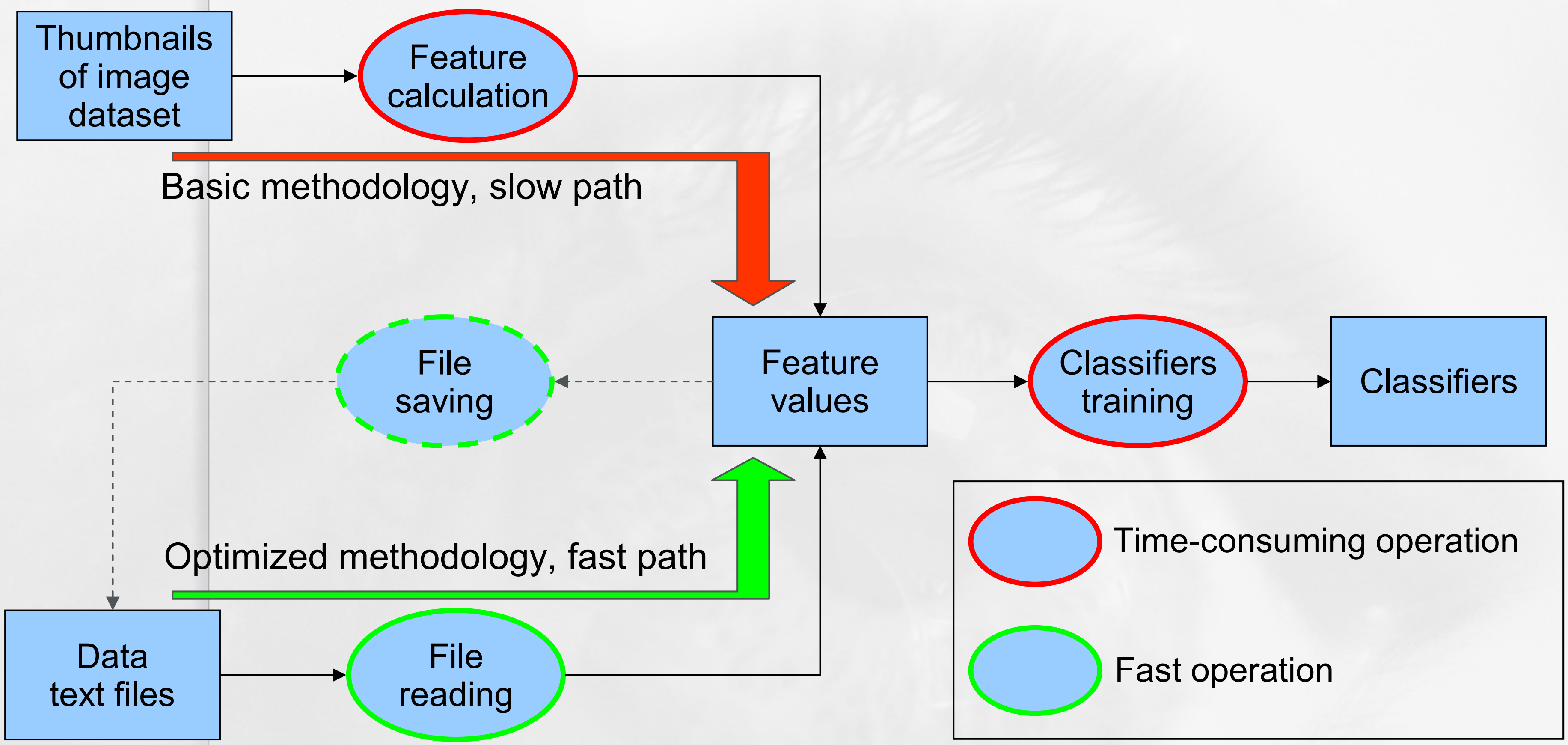
Feature analysis

- Elimination of redundant features thanks to a dimensionality reduction technique (Principal Feature Analysis, Cohen et al. 2002)
- Less features → faster calculation / classification

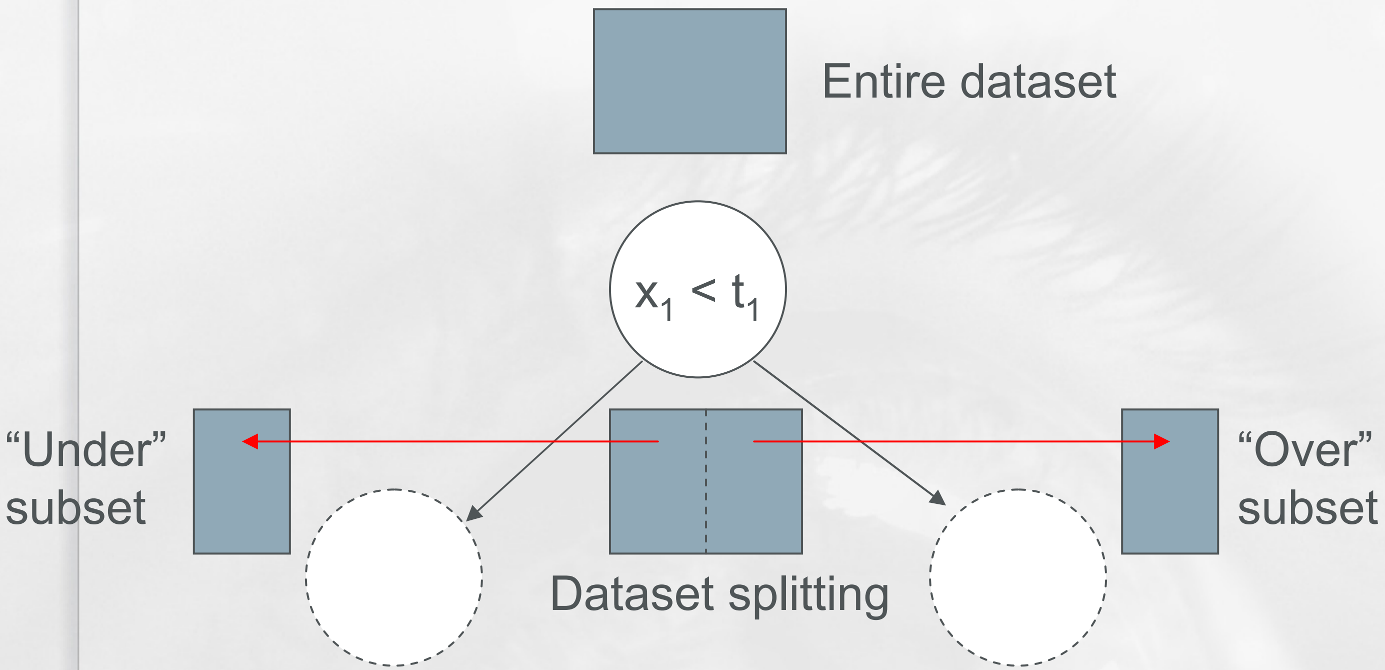
Feature	Number
Number of colors	1
Number of gray levels	1
Standard deviation of the hue component	1
Standard deviation on color channels	3
X Derivative gradient counts on color channels	6
Sobel filters gradient counts on color channels	12
Laplace operator gradient counts on color channels	6
NW Kirsh operator gradient counts on color channels	6
Absolute DCT coefficient sums	2
Total number	**36**

BARCO
Visibly yours

Improvement on methodology

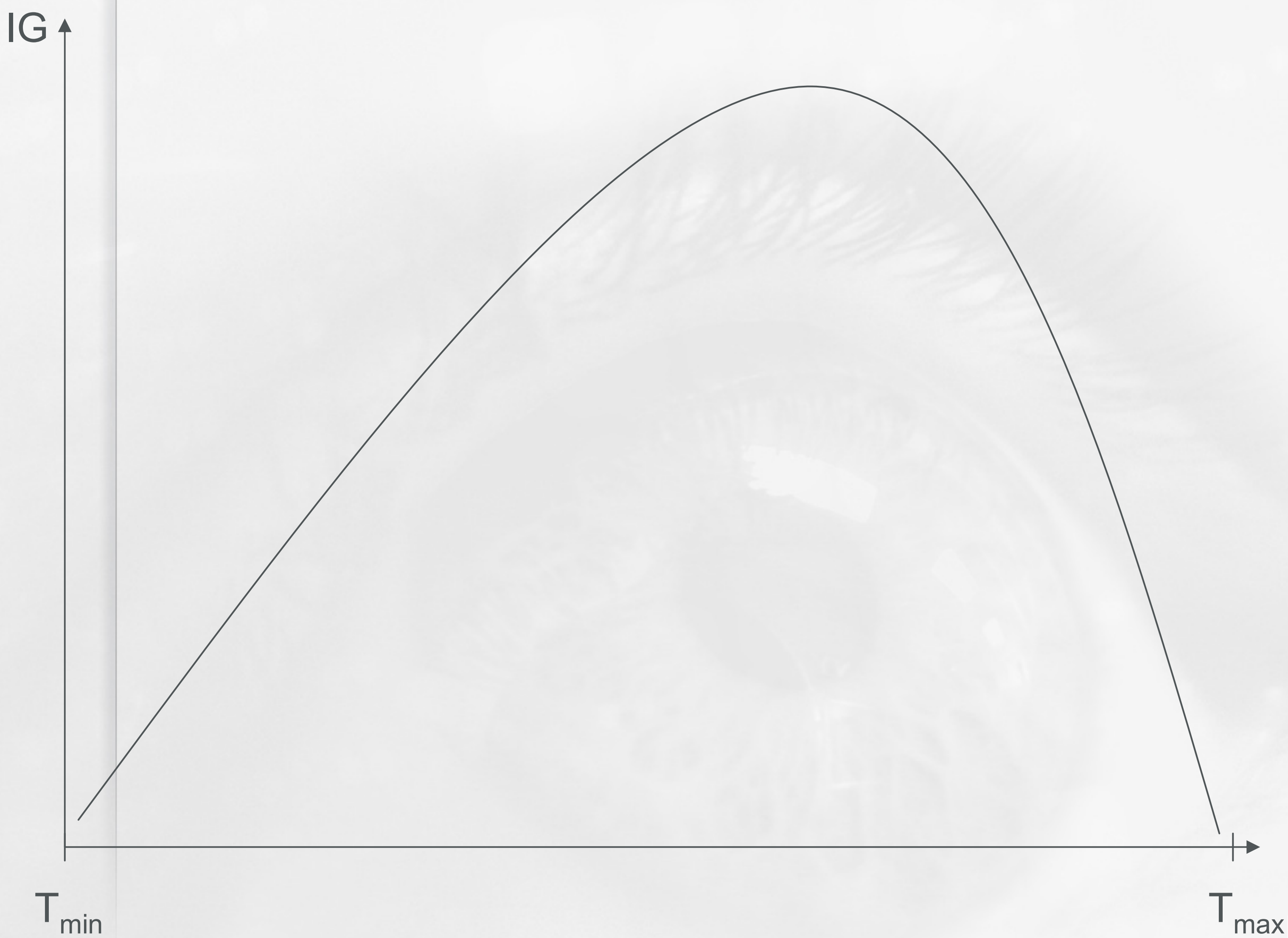


Faster decision tree training

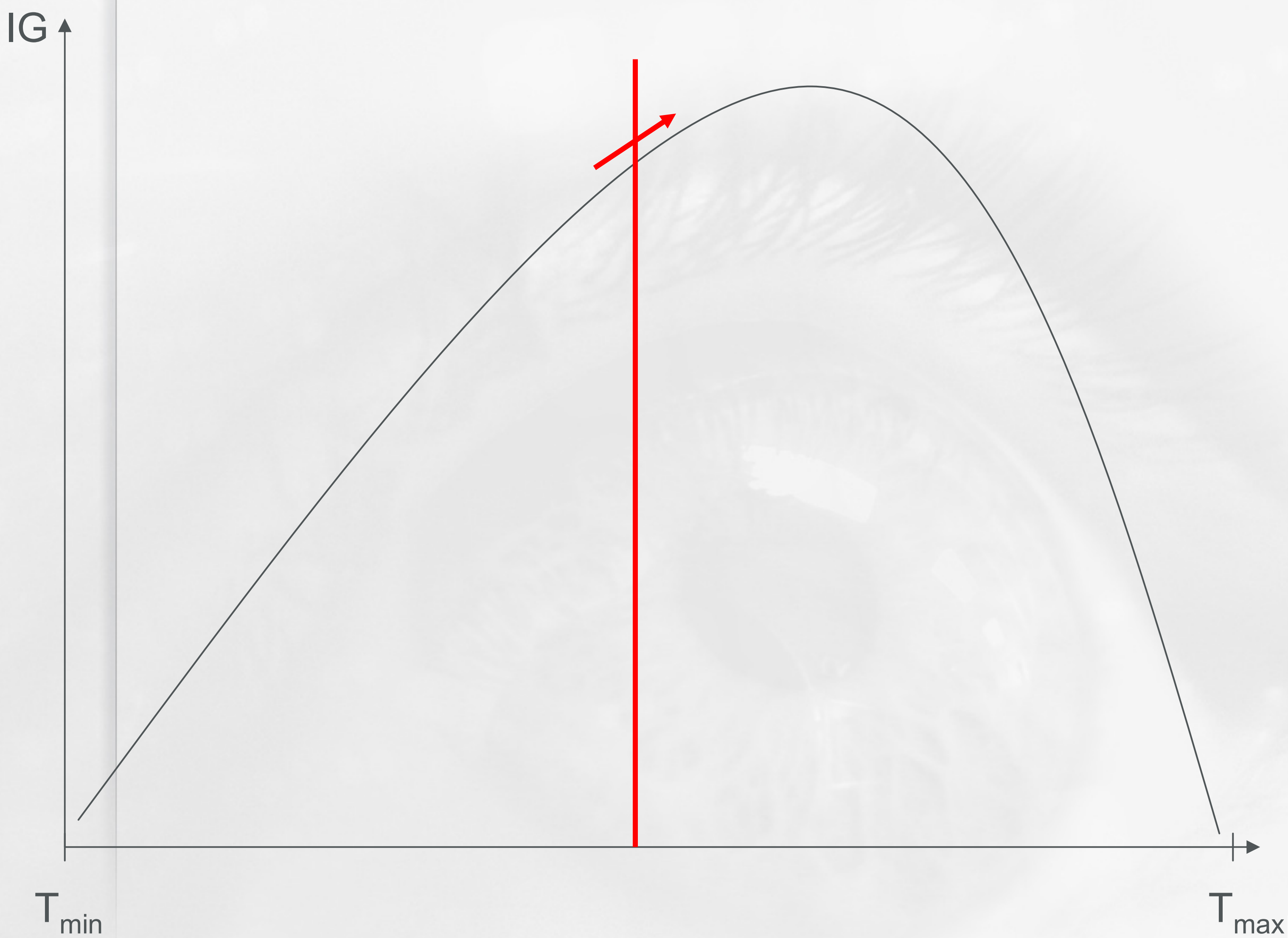


- Threshold searching requires
  - **Sorting the dataset for each node and for each feature.**
  - **Testing the relevance of each potential threshold.**

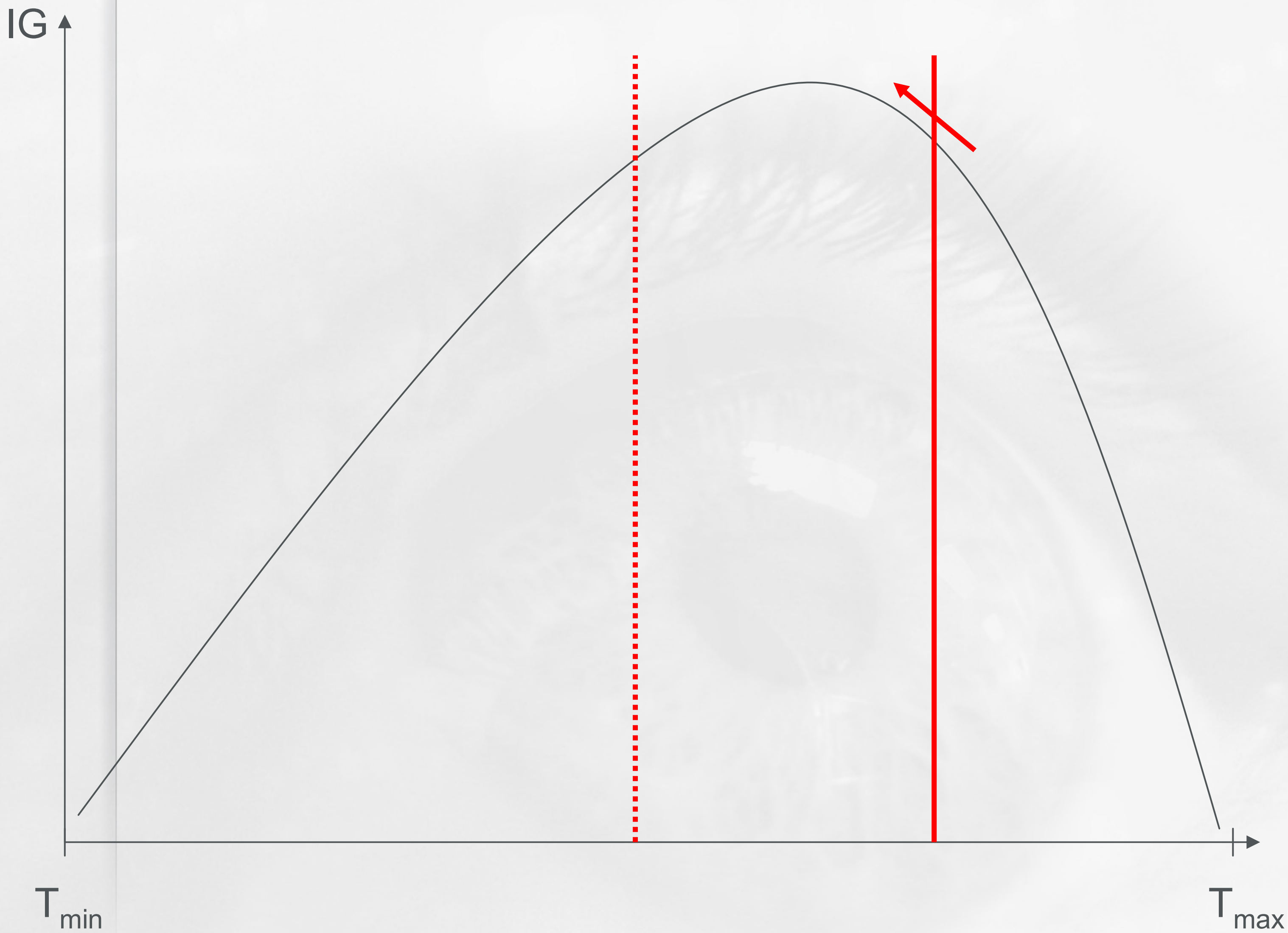
Binary threshold search



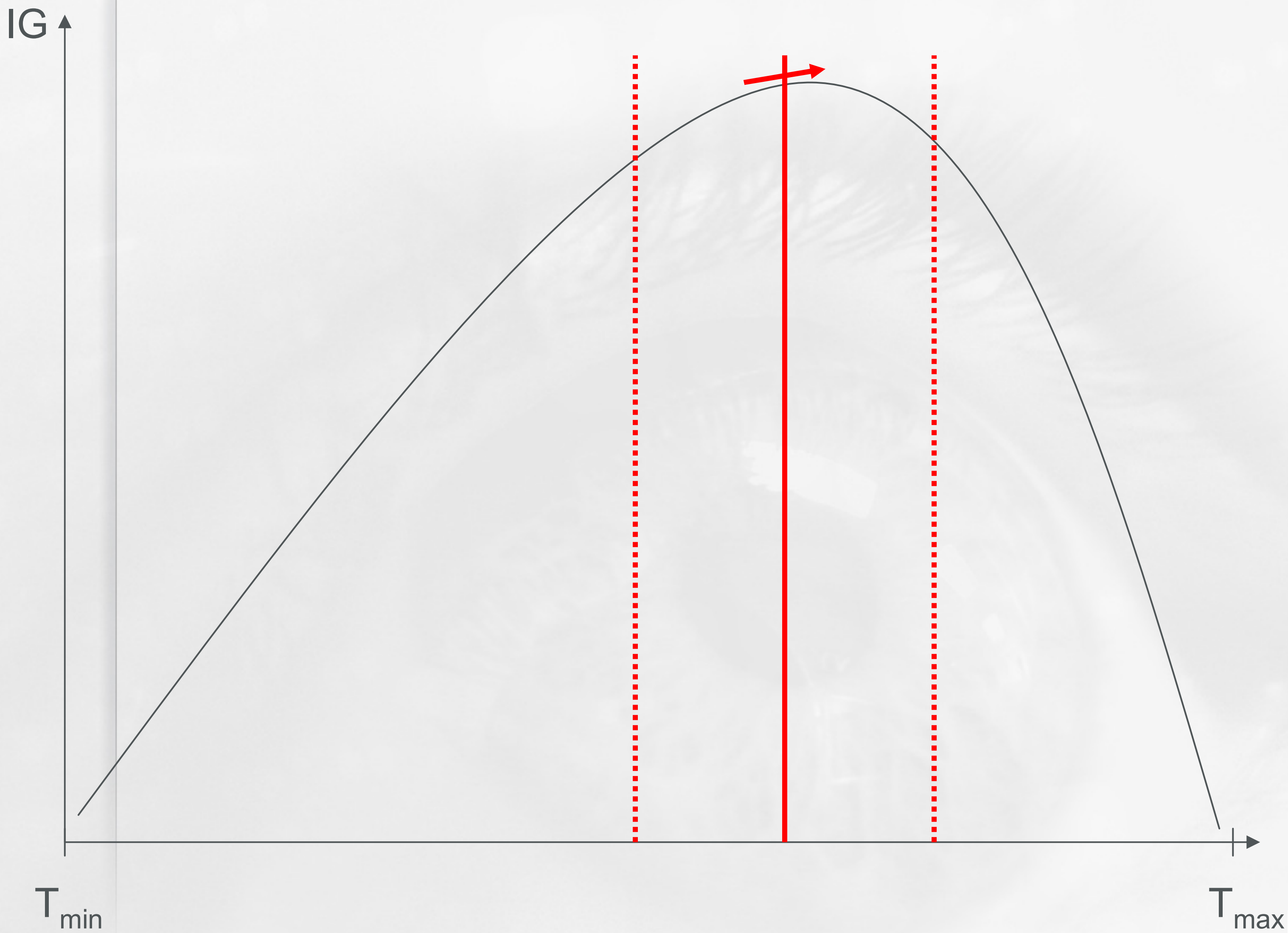
Binary threshold search



Binary threshold search

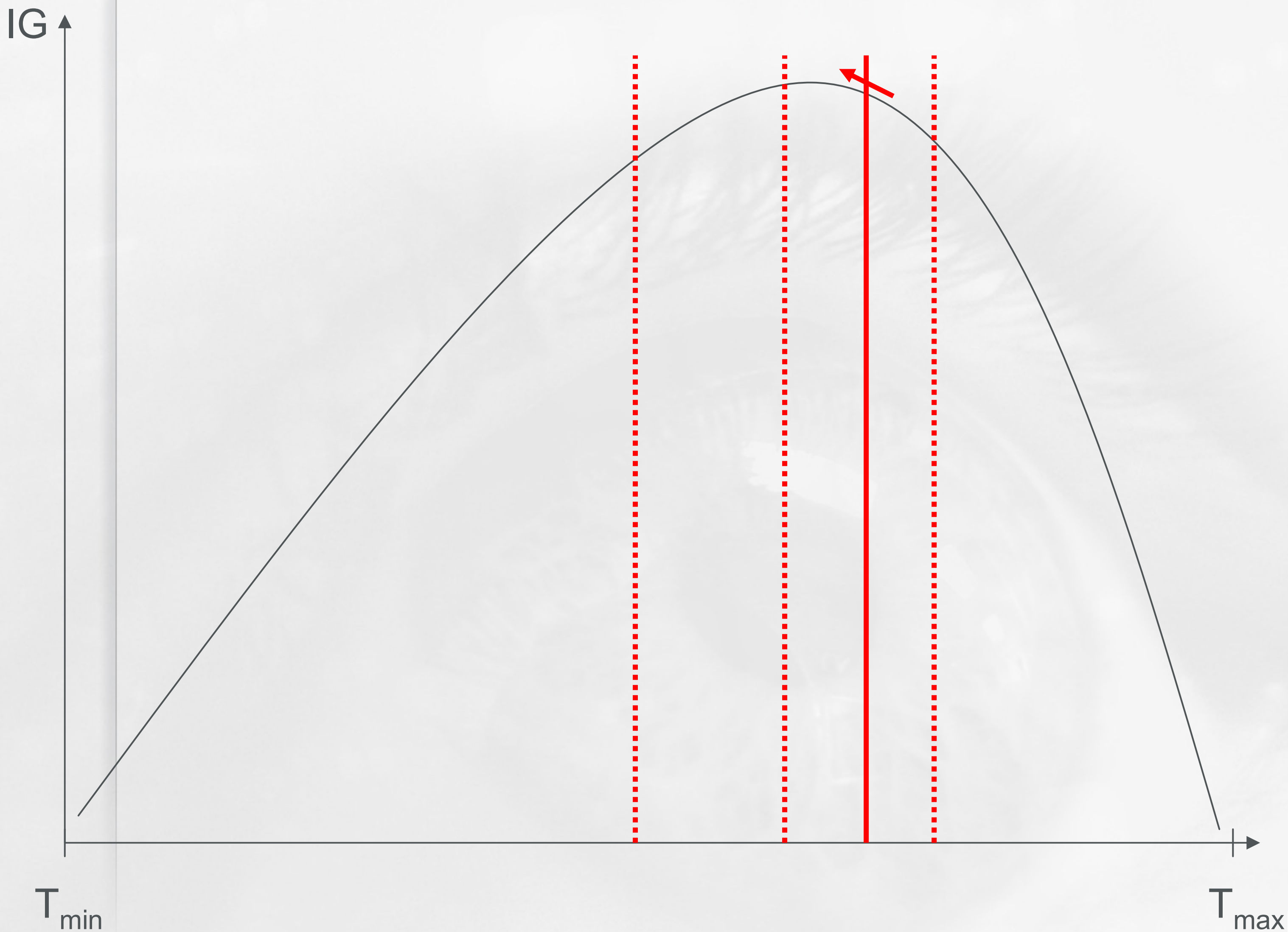


Binary threshold search



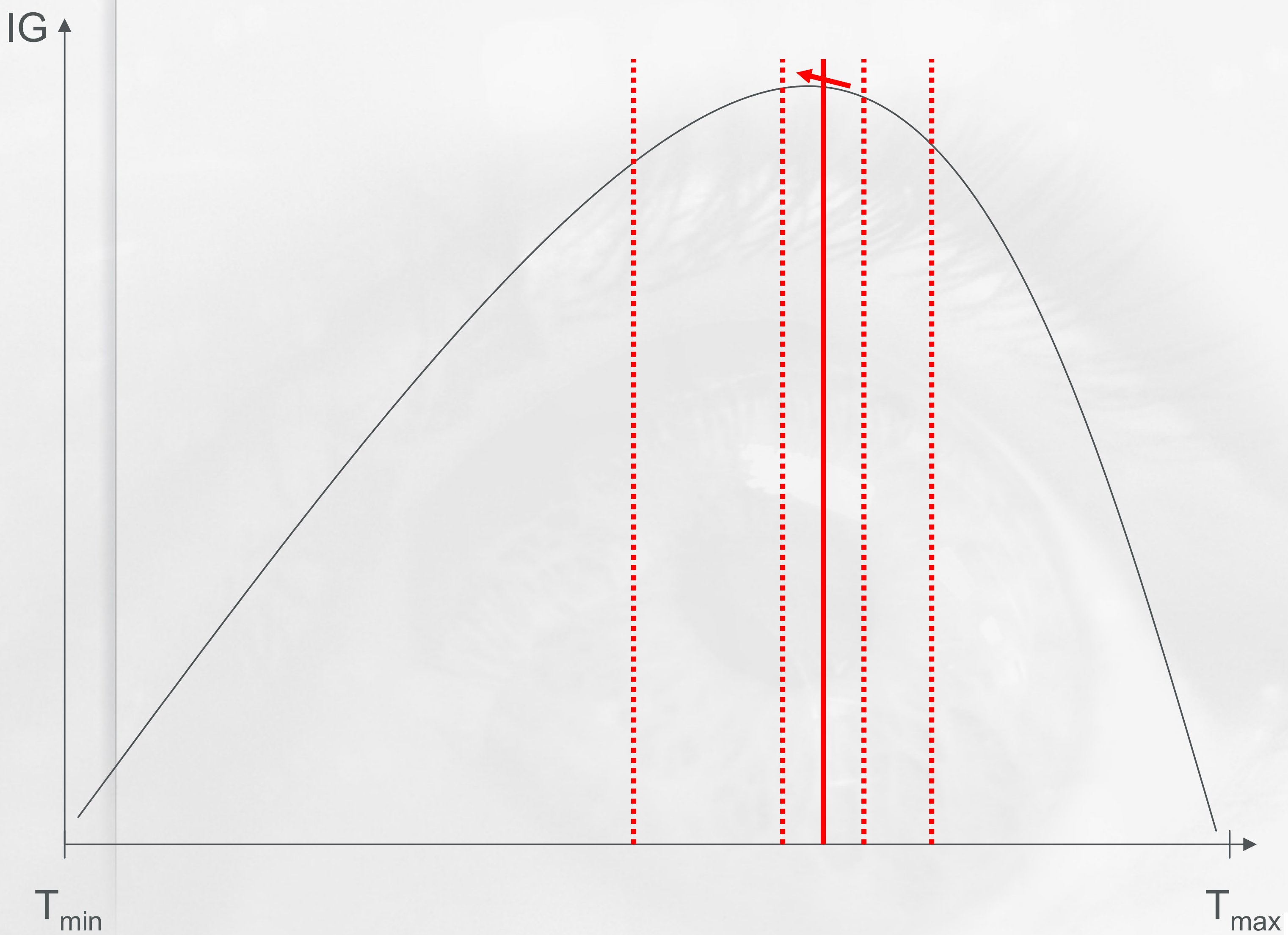
Visibly yours

Binary threshold search

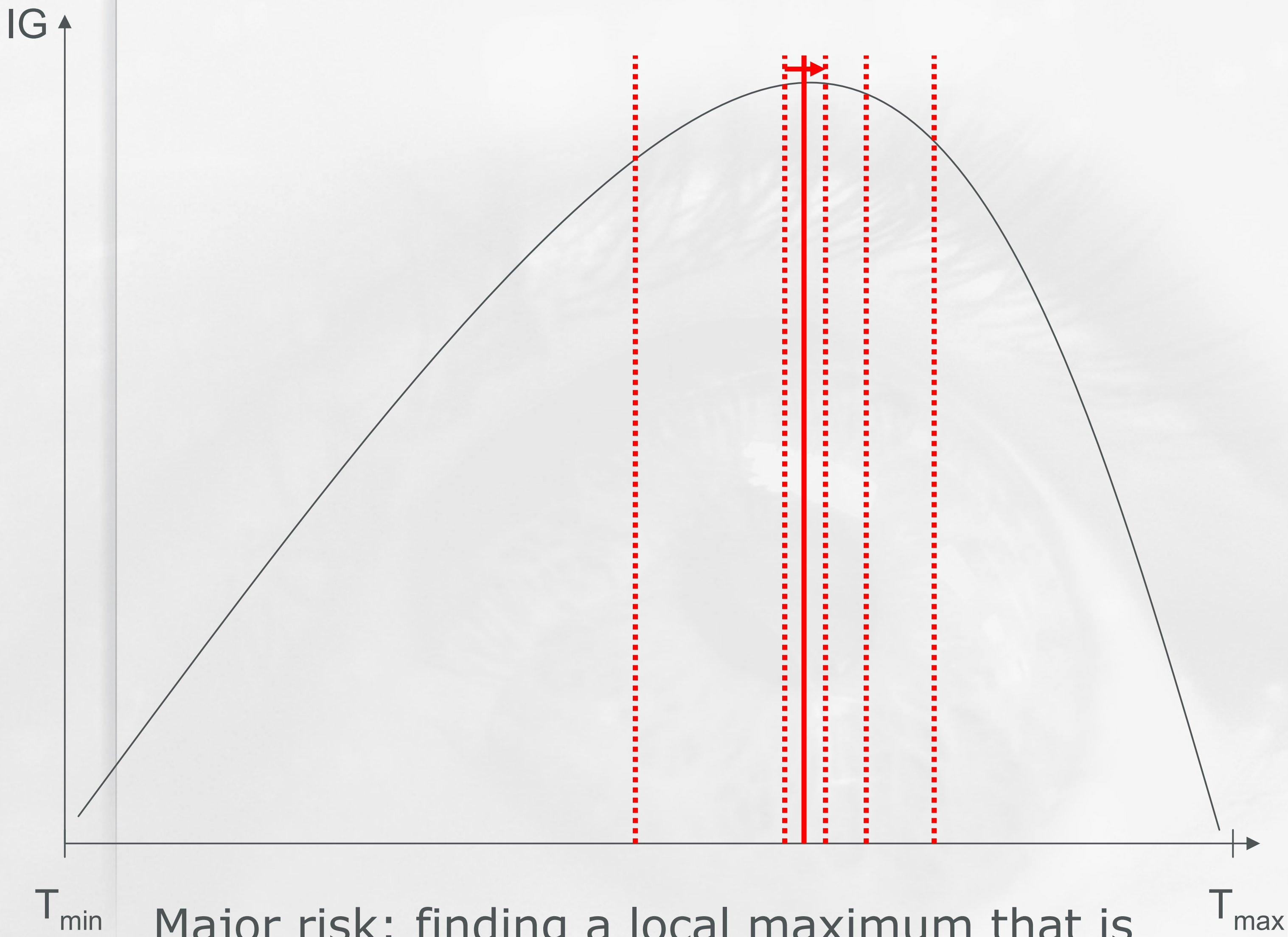


Visibly yours

Binary threshold search

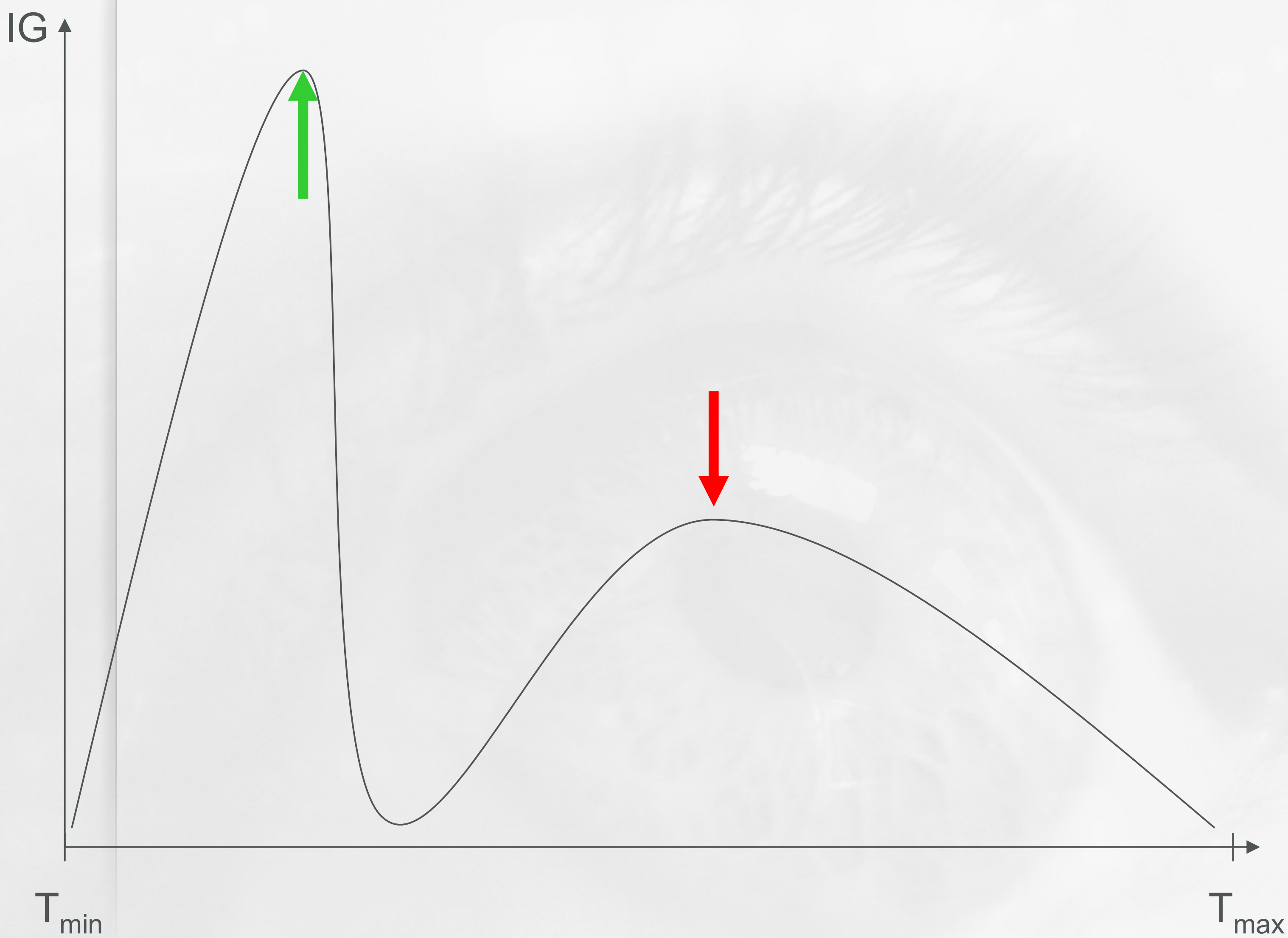


Binary threshold search



Major risk: finding a local maximum that is not the absolute maximum.

Binary threshold search



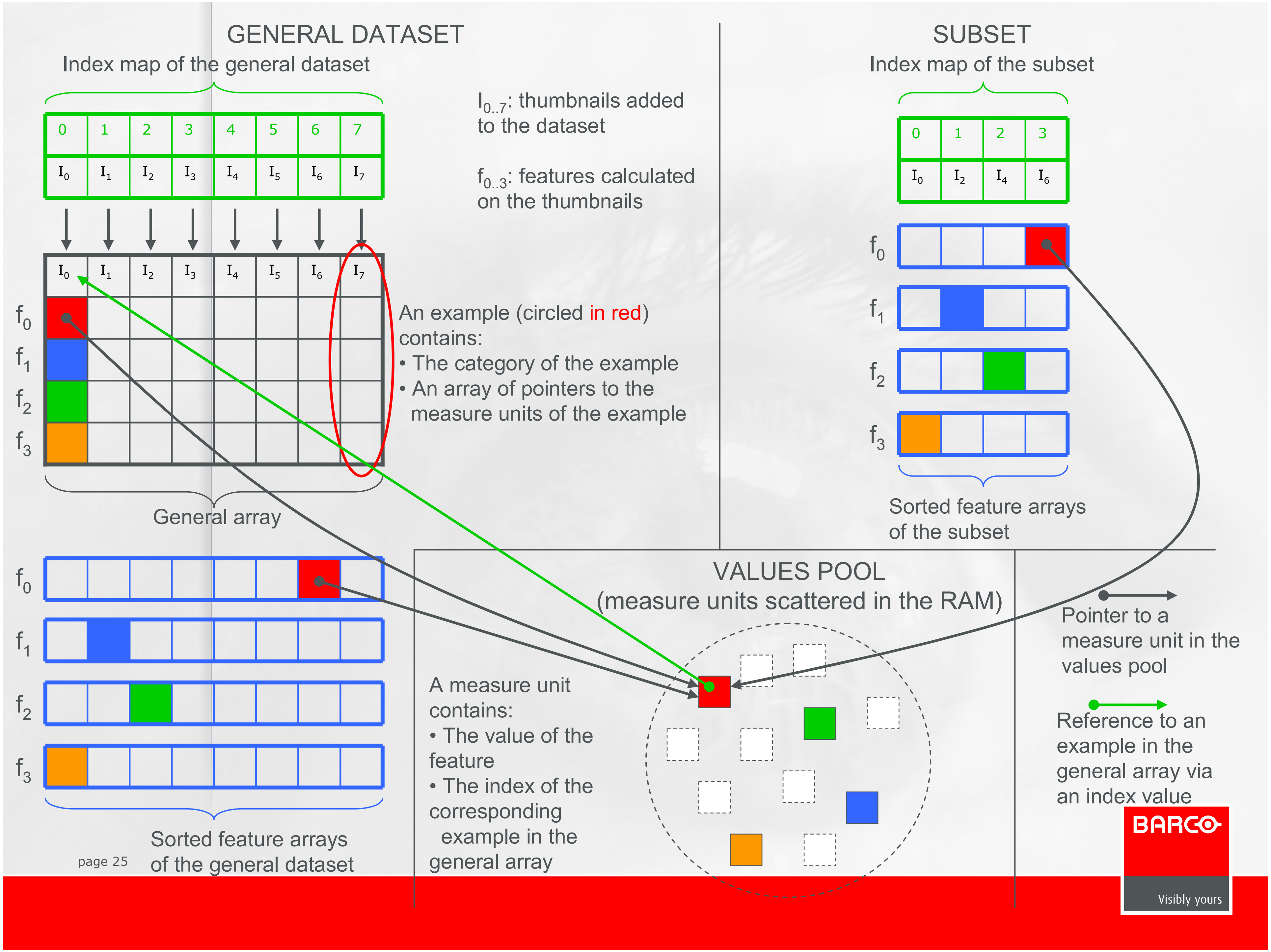
Using an optimized implementation of the training dataset

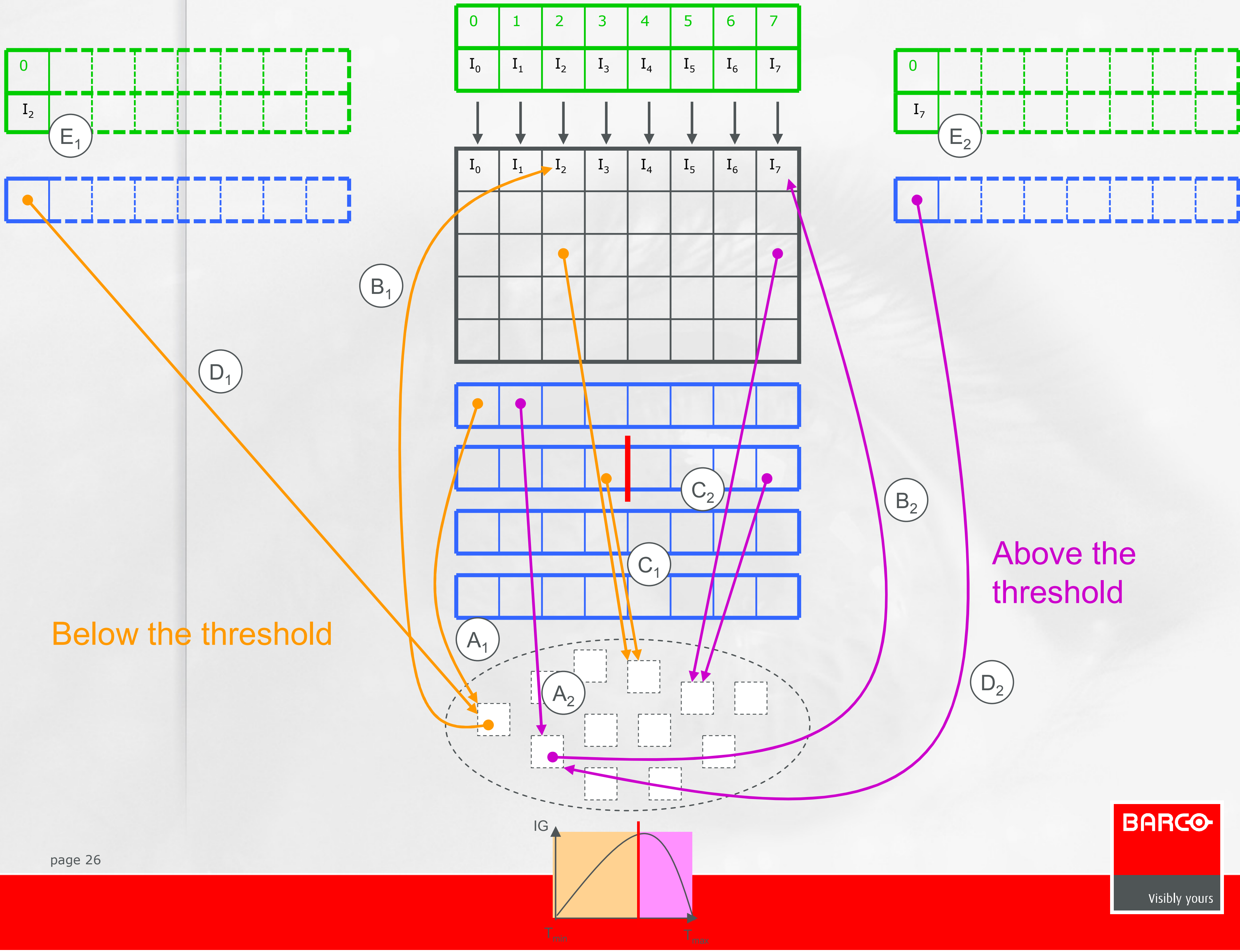
- Sorting the features only once.

- Maintain the features sorted at each dataset split

Straightforward dataset implementation

	I_0	I_1	I_2	I_3	I_4	I_5	I_6	I_7
f_0								
f_1								
f_2								
f_3								


GENERAL DATASET
Index map of the general dataset
0 1 2 3 4 5 6 7
I0 I1 I2 I3 I4 I5 I6 I7
I0..7: thumbnails added to the dataset
f0..3: features calculated on the thumbnails
f0 f1 f2 f3
An example (circled in red) contains:
• The category of the example
• An array of pointers to the measure units of the example
General array
Sorted feature arrays of the general dataset
SUBSET
Index map of the subset
0 1 2 3
I0 I2 I4 I6
Sorted feature arrays of the subset
VALUES POOL
(measure units scattered in the RAM)
A measure unit contains:
• The value of the feature
• The index of the corresponding example in the general array
Pointer to a measure unit in the values pool
Reference to an example in the general array via an index value
BARCO
Visibly yours


0 1 2 3 4 5 6 7
I0 I1 I2 I3 I4 I5 I6 I7
E1
E2
B1
B2
C1
C2
D1
D2
A1
A2
Below the threshold
Above the threshold
IG

Validation

- Training conditions
  - **Cascade of four categories: medical, text, GUI, other complex.**
  - **Each categories is trained through 30 iterations of AdaBoost.**
  - **Maximum depth level of decision trees: 6 for text, 4 for the others.**
- Four different configurations
  - **No optimization (0p0b)**
  - **Binary threshold search only (0p1b)**
  - **Pre-sorted dataset implementation (1p0b)**
  - **Both optimizations (1p1b)**
- Test dataset of 3000 images

Results

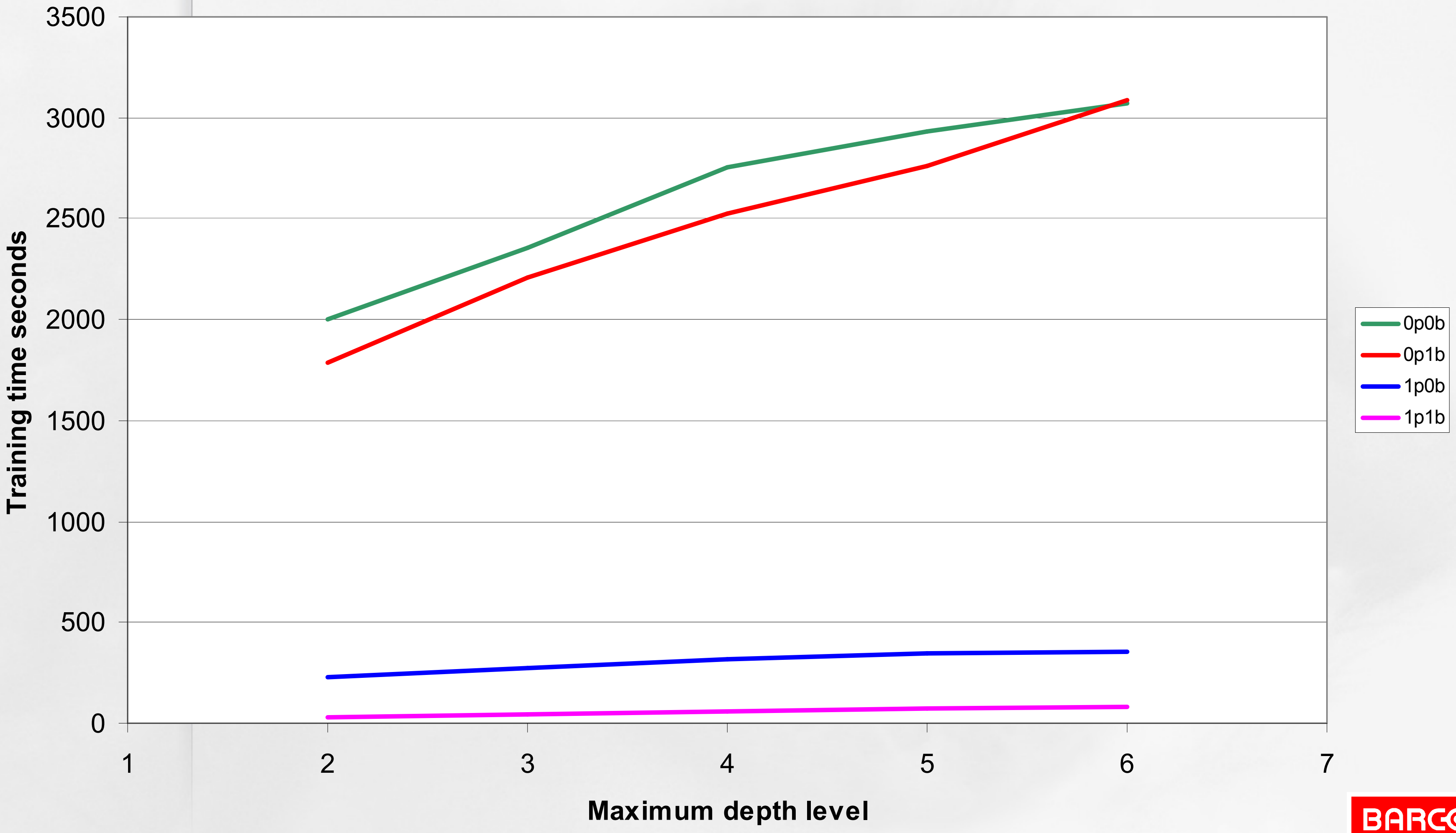
	True Positives	True Negatives	False Positives	False Negatives	Not Classified
Medical	76.4%	99%	1%	23.6%	10.6%
Text	52.6%	95.6%	4.4%	48.4%	19.3%
GUI	52.8%	84.9%	15.1%	47.2 %	26.1%
Other complex	81.6%	92.2%	7.8%	18.4 %	11.2%

- These results are the same for the four configurations.
- Especially interesting for binary threshold searching.

Training time depending on maximum decision tree depth

- Text classifier
- 10 decision trees
- Training dataset: 40000 images
- Computer
  - **2GHz CPU**
  - **1GB RAM**

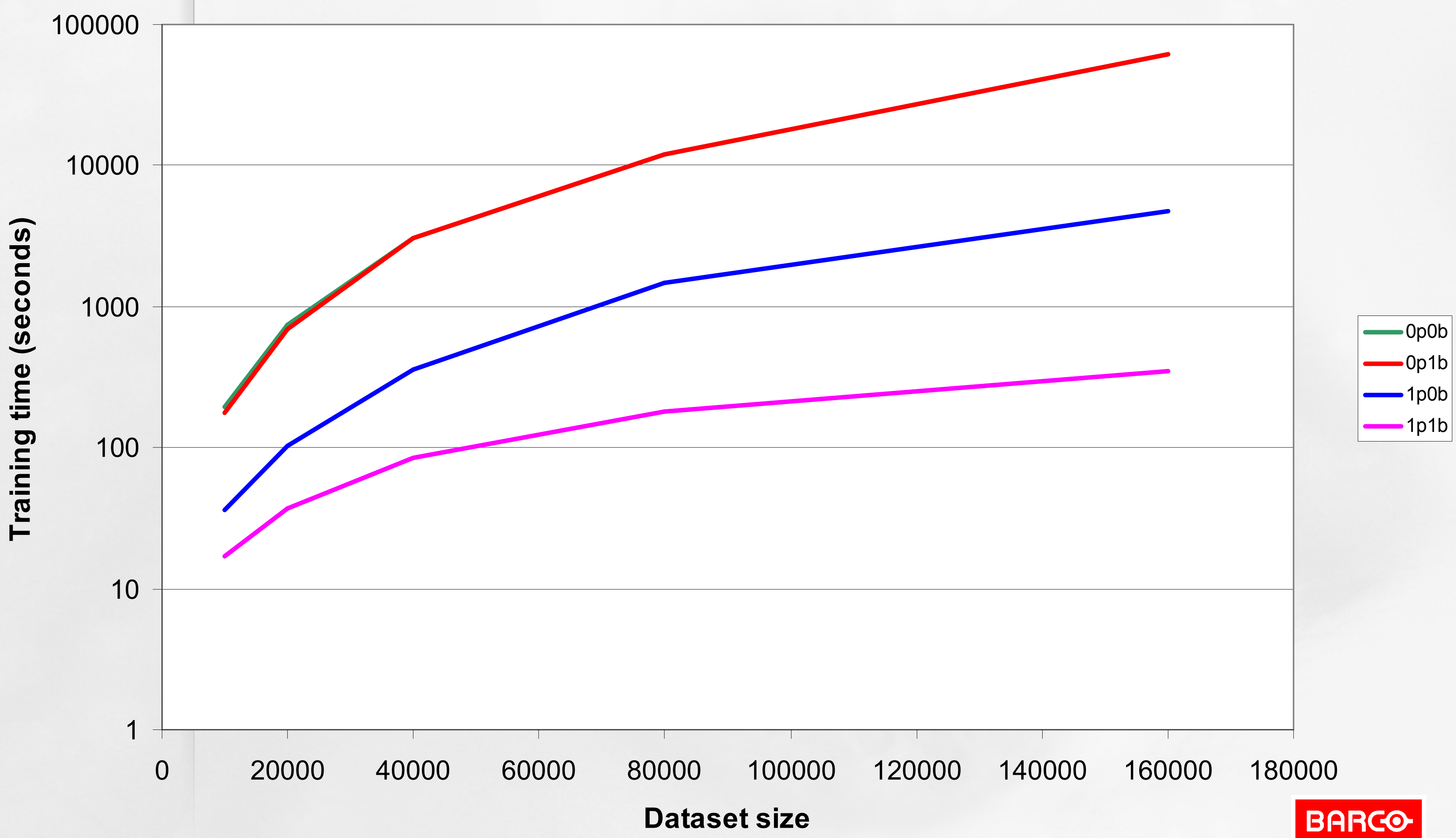
Configuration	Maximum decision tree depth				
	2	3	4	5	6
0p0b	2004s	2354s	2756s	2931s	3075s
0p1b	1784s	2208s	2526s	2762s	3087s
1p0b	226s	273s	320s	350s	354s
1p1b	32s	47s	60s	73s	84s


3500
3000
2500
2000
1500
1000
500
0
Training time seconds
1
2
3
4
5
6
7
Maximum depth level
0p0b
0p1b
1p0b
1p1b

Training time depending on dataset size

- Text classifier
- 10 decision trees
- Maximum depth: 6 nodes
- Computer
  - **2GHz CPU**
  - **1GB RAM**

Configuration	Training dataset size				
	10000	20000	40000	80000	160000
0p0b	195s	744s	3075s	11995s	61681s
0p1b	174s	698s	3087s	11913s	61889s
1p0b	36s	104s	354s	1455s	4716s
1p1b	17s	37s	84s	179s	352s


Training time (seconds)
100000
10000
1000
100
10
1
0
20000
40000
60000
80000
100000
120000
140000
160000
180000
Dataset size
0p0b
0p1b
1p0b
1p1b

Observed complexities

Configuration	Approximate complexity
0p0b	$O(n^2)$
0p1b	$O(n^2)$
1p0b	$O(n^{3/2})$
1p1b	$O(n)$

Conclusions

- A multi-content analysis framework for compound images was optimized for faster training time without noticeable performance loss.
- The time gain provided by these optimization allowed us to test a wide range of parameters during the training phase of classifiers in a very short time → minimization of overfitting.
- The developed optimizations are not specific to image analysis.

Visibly yours

Aknowledgements

- CANTATA project
  - **ITEA program**
- IWT

Thank you for your attention

Quentin Besnehard
Barco nv
Medical Imaging Division
President Kennedypark 35
8500 Kortrijk
Belgium

Quentin.Besnehard@barco.com

Visibly yours