CEEDs

The Collective Experience of Empathic Data Systems

15 February 2012

Using Attributed Affect for Implicit Sentiment Image Tagging & Content-based Retrieval

K.C. Apostolakis & P. Daras
Informatics & Telematics Institute
Centre for Research & Technology Hellas

SEVENTH FRAMEWORK PROGRAMME

Implicit Human-Centred Tagging (IHCT)

- Attempts to obtain user behavioural response for tagging purposes
  - Effectively reduces user effort in contrast to explicit (textual) annotation
- Challenges (Pantic & Vinciarelli, 2009)
  - Effort to include observed user reactions & behaviour, as well as implicit tags to the data tagging & retrieval loop
  - Develop behaviour analyzers that can attain accurate and reliable results even on audiovisual sensors built in commercial computers

Psychological Framework (Russell, 2003)

- Core Affect
  - *2-D space defined by two components*
    - ***Valence*** – Amount of pleasure experienced at any given moment
    - ***Arousal*** – Activation level in preparation for action
    - e.g. feeling *delighted*, *bored*, etc.
- Perception of Affective Quality
  - *To perceive stimuli in terms of their emotional properties*
    - e.g. *delicious* meal, *boring* lecture etc.

Psychological Framework (Russell, 2003)
Attributed Affect

- Subconscious attempt to attribute change in Core Affect to its perceived cause
- The stimulus that is identified as the cause becomes **the "Object"**
  - ***Attention*** *is shifted towards* ***the "Object"***
  - ***Behaviour*** *is directed at* ***the "Object"***
- Defines Emotional Awareness
  - *Main route to the affective quality of the stimulus*

Introducing Attributed Affect to IHCT problem

- Obtain user *affective response* (Core Affect)
- Obtain *the "Object"* via gaze information
  - *Identify specific stimulus depicted in the image, where the users have focused their attention on*
- Attribute *affective response* to *the "Object"*
- Image annotated with appropriate *affective tag*
- A new image containing *the "Object"* is automatically annotated with the assoc. label

Introducing Attributed Affect to IHCT problem Advantages (1/4)

- Automatic annotation of large portions of the image database by looking at a single image
  - *User looks at image depicting a spider and experiences a **jittery** reaction*
  - *Spider identified & attributed as the cause → spider considered **jittery** by the user*
  - *Framework annotates all images in the collection depicting spiders with the **jittery** affective label*
    - User most likely to experience the same reaction when presented with the same stimuli

Introducing Attributed Affect to IHCT problem Advantages (2/4)

- Retrieval & Recommendation readily available through annotated stimulus
  - *User looks at images of cars, trying to locate models that spark his interest*
  - *Several cars are identified as causing feelings of satisfaction – others are dismissed*
  - *Annotation of all images in the collection, depicting either dismissing or pleasing stimuli*
  - *Retrieval of images that were annotated as '**pleasing**'*

Introducing Attributed Affect to IHCT problem Advantages (3/4)

- Annotation based on user personal experience
  - Users can annotate content specifically to their preferences
    - Not all spiders are considered *jittery* by all people!
  - Several culture-dependant points addressed
    - Something funny here might be considered offensive somewhere else
  - Personalized recommendation of like-valenced content
    - Horror movies scare me! → don't show me Horror movies!
    - I love Horror movies! → Show me more!

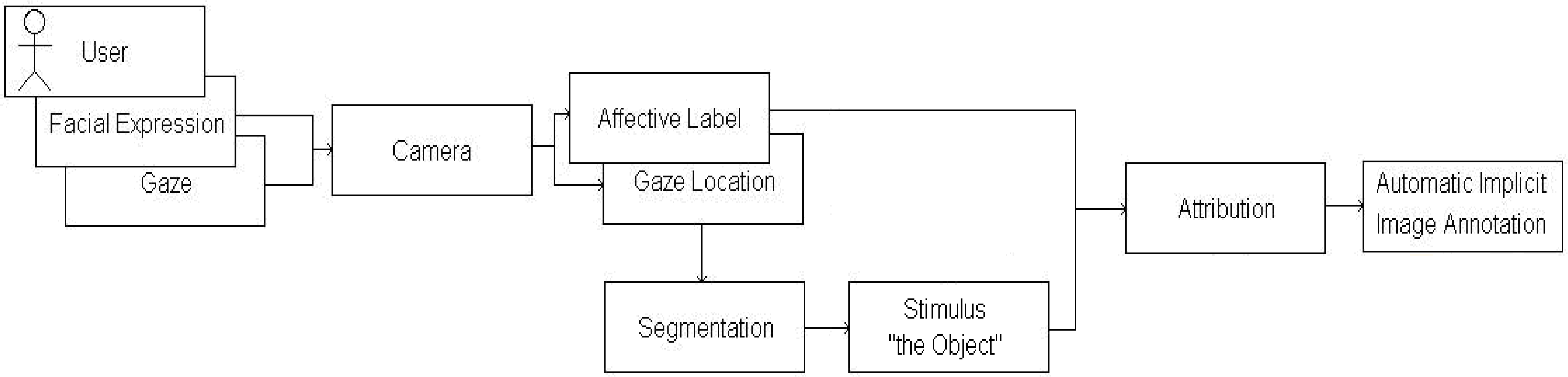
Introducing Attributed Affect to IHCT problem Advantages (4/4)

- IHCT based on Attributed Affect can be applied to multitude of setups, as long as there's a means to obtain gaze and affect information
  - *Many methods for obtaining user affective response*
    - Facial Expressions, Blood Pressure, Body Temperature, EMC, etc.
  - *Many methods for obtaining gaze information*
    - Single Image, Stereo, Special apparatus (eyeglasses)
  - *Most affordable setup: Commercially available computer systems with a single low-res camera*
    - Today's Laptop computers!

The Framework

- Input received via Affect Recognition and Gaze Tracking modules
  - Affect Recognition module identifies affective response and generates affective label → *tag*
  - Gaze Tracking module tracks user's eye gaze and generates gaze point on the image display screen
  - Segmentation module receives gaze point and generates a foreground image of the viewed stimulus → *the "Object"*
- Output contains **the "Object"** (foreground image) and **affective quality** (tag)
  - The "Object" → retrieval
  - Affective Quality → annotation

The Framework



The Framework
Obtaining User Affective Response (1/5)

- Affective Response obtained via Facial Expression Analysis
  - *Available to single low-res webcam setup*
- Facial Action Coding System (FACS) (Ekman & Friesen 1978)
  - *Deconstructs every anatomically possible facial expression into a set of Action Units (AUs)*
  - *AUs describe the movement of individual facial muscle groups*
  - *Different comb. of AUs → different expressions*

The Framework
Obtaining User Affective Response (2/5)

- Identifying AU activation
  - *Track key facial features corresponding to AU muscle groups*
- Active Shape Model (ASM, Cootes & Taylor 1995)
  - *Statistical model describing the shape of an object*
  - *Capable of deforming to fit to a new instance of the object*
- Applications
  - *Face tracking, Hand Tracking, Object Fitting, X-Ray Segmentation, etc…*

Obtaining User Affective Response
ASM Fitting (1/2)

- Facial Active Shape Model Built out of 161 frontal face images
  - *Picked out of 5 freely available databases*


Talking Face[1]


IMM[2]


BioID[3]


MUCT[4]


IR Marks[5]

- Manual annotation of 68 landmarks to obtain face shape

---

[1] http://personalpages.manchester.ac.uk/staff/timothy.f.cootes/data/talking_face/talking_face.html

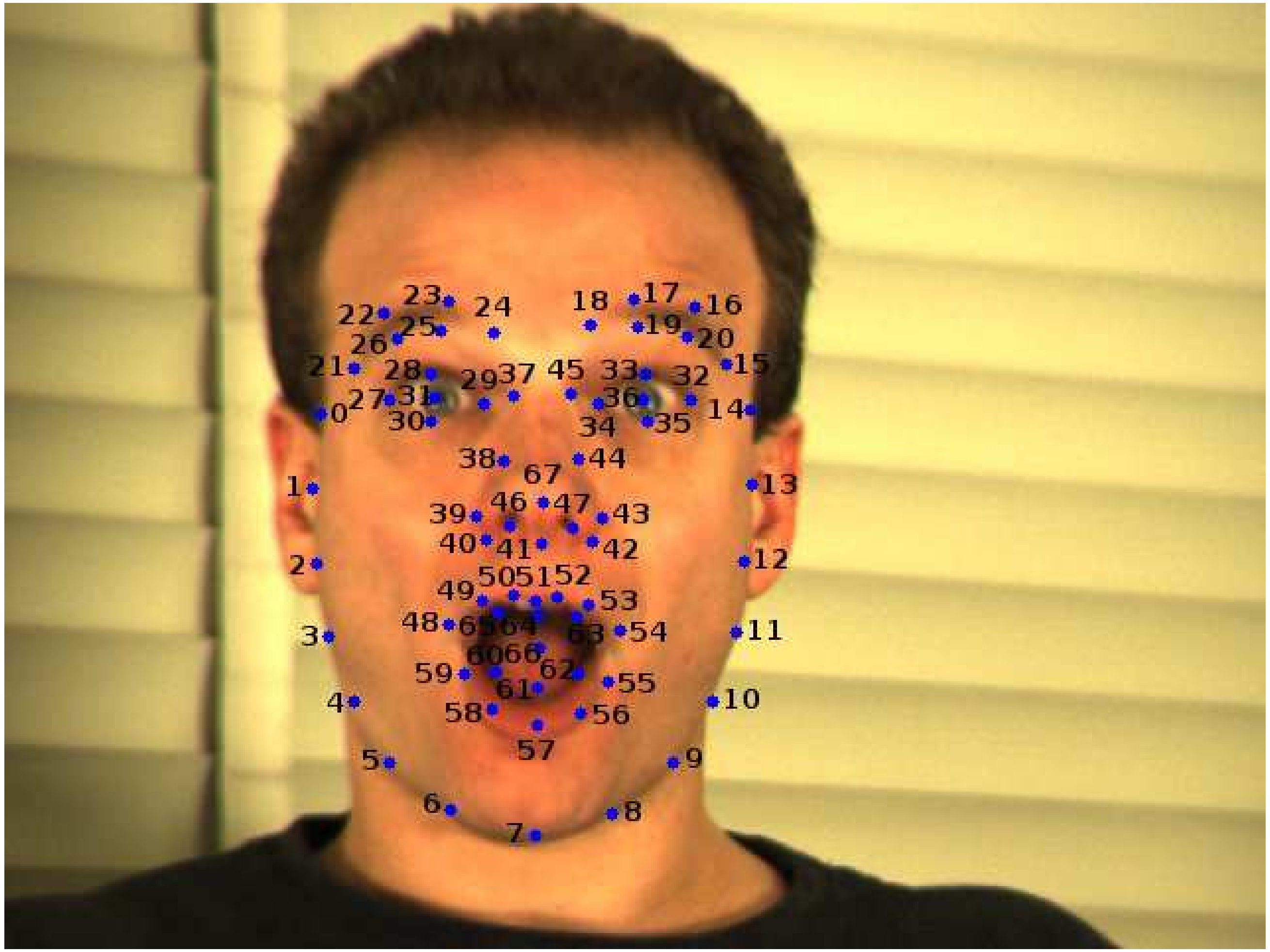
[2] http://www2.imm.dtu.dk/pubdb/views/publication_details.php?id=922

[3] http://support.bioid.com/downloads/facedb/index.php

[4] http://www.milbo.org/muct/

[5] http://mplab.ucsd.edu/wordpress/?page_id=1207

Obtaining User Affective Response
ASM Fitting (2/2)

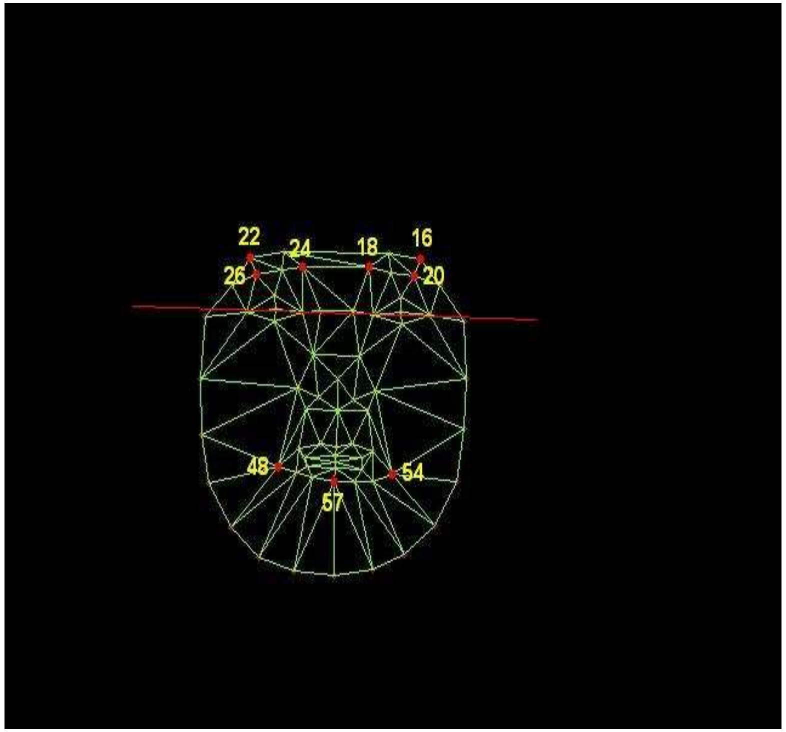


The Framework
Obtaining User Affective Response (3/5)

TABLE I
ESTIMATED VALENCE – AROUSAL MAPPING TO AU ACTIVATION DURING POSED DISPLAY OF CERTAIN FACIAL EXPRESSIONS

Facial Expression	Corresponding Action Units (AU)	Mean Valence Estimate for both sexes	Mean Arousal Estimate (for both sexes)
Happiness	6 + **12**	+ 2.990	+ 2.140
Anger	**4** + 7 + 23	− 1.685	+ 1.240
Fear	**1** + **4** + 5 + 25	− 2.215	+ 1.475
Surprise	**1** + **2** + **26**	− 0.010	+ 1.515
Sadness	**1** + **4** + **15**	− 2.190	− 0.605
Neutral	-	+ 0.025	− 1.205

Action Unit	Action Unit Name	Corresponding Landmark(s)	Effect on Valence	Effect on Arousal
12	Lip Corner Puller	48, 54	+	↑
15	Lip Corner Depressor	48, 54	−	↓
1	Inner Brow Raiser	18, 24	None	↑
2	Outer Brow Raiser	16, 22	None	↑
4	Brow Lower	(18 + 20), (24 + 26)	−	↑
26	Jaw Drop	57	None	↑

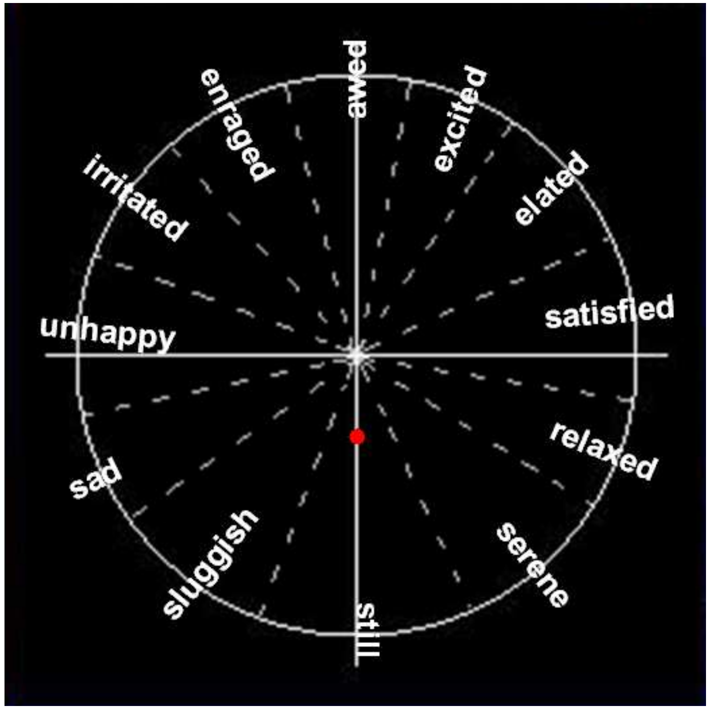


The Framework
Obtaining User Affective Response (4/5)

- Procedure:
  - *Take snapshot of "neutral" expression*
    - Fit ASM and save "neutral" landmark positions
    - Calculate landmark distances from the eye line
  - *For every consequent frame:*
    - Fit ASM
    - Calculate landmark distances from the eye line
    - Calculate AU intensity from the distance differences
    - Calculate valence – arousal according to Eqs:
      - $Valence = AU\ 12 - \left(\frac{AU\ 15 + AU\ 4}{2}\right)$
      - $Arousal = -0.30125 + \frac{1.30125}{5}(AU1 + AU2 + AU4 + AU12 + AU26) - AU15$

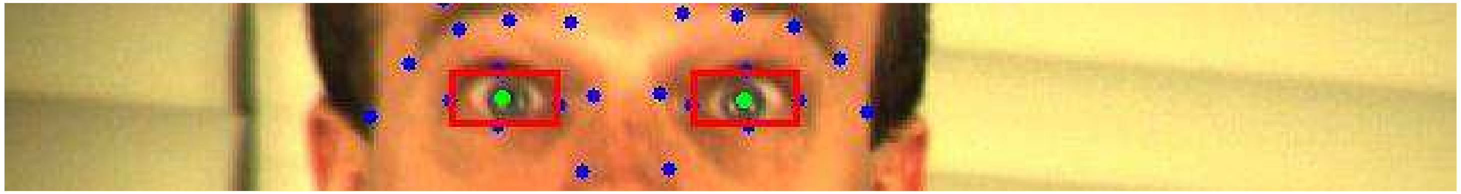
The Framework
Obtaining User Affective Response (5/5)

- Core Affect value normalized and placed inside 2D Affective Circumplex
- Extraction of affective label via Yik et al (2011)



The Framework
Gaze Tracking (1/3)

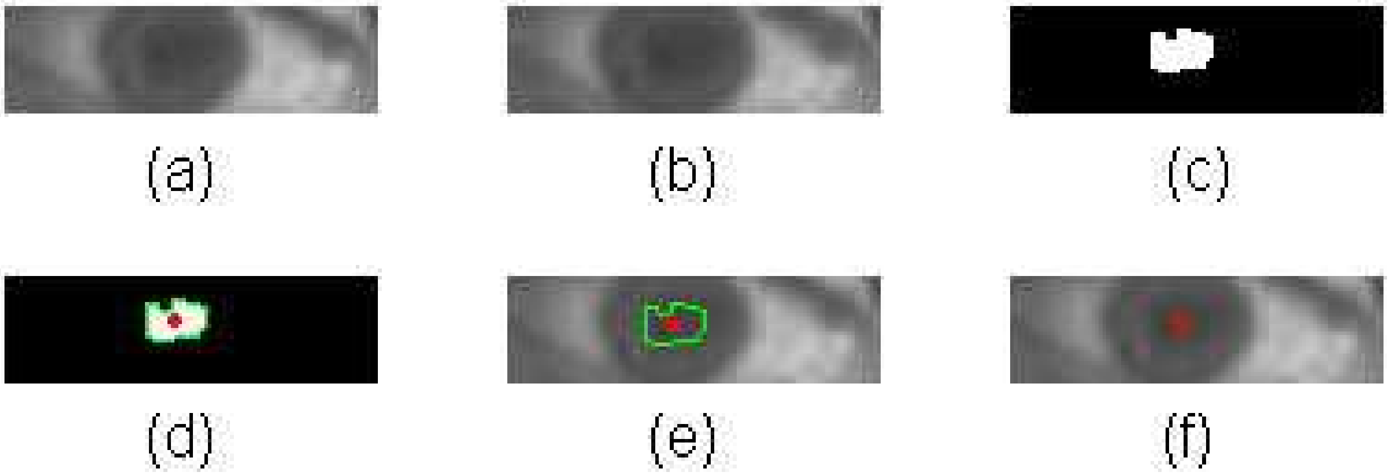
- Single Image Gaze Tracking & Gaze Point Est.
  - *Locate the iris centre (pupil) P & eye corners E1, E2*
  - *Map current information on P, E1, E2 to 2D screen coordinates*
    - Requires calibration step
- Eye corners located via ASM (27, 29, 32, 34)
  - *Generation of Pupil Search Area (PSA)*
  - *Pupil is certain to be contained within PSA*



The Framework
Gaze Tracking (2/3)

- Locate Pupil via Automatic Adaptive Thresholding
  - *Thresholding*
    - Convert greyscale PSA image to binary image using threshold
    - Threshold value determines which pixels are painted white (1) and are part of the object
  - *Adaptive*
    - Threshold value specific to each frame
    - Adaptation to lighting, position changes
  - *Automatic*
    - Thresholds in [0, 255] applied iteratively until one is chosen

Gaze Tracking
Automatic Adaptive Thresholding



The Framework
Gaze Tracking (3/3)

- Gaze Point Estimation via Linear 2D Mapping
  - *Calibration*
    - Calibration points displayed on the screen to collect info
    - Users fixate their gaze on each calibration point
  - *Linear 2D Mapping*
    - Pupil centre positions P_i (x_i, y_i) stored for each calibration point K_i (α_i, β_i) during calibration
    - Minimum of 2 calibration points K_1 (α_1, β_1), K_2 (α_2, β_2)
    - Every subsequent P' (x', y') mapped to screen coordinates (α', β') via Eqs:

$$\alpha ' = a_1 + \frac{x' - x_1}{x_2 - x_1}(a_2 - a_1) \qquad \beta ' = \beta_1 + \frac{y' - y_1}{y_2 - y_1}(\beta_2 - \beta_1)$$

The Framework
Identifying the "Object" (1/3)

- "Object" identified via image segmentation
- Segmentation algorithms
  - *Require explicit designation of foreground – background pixel seeds*
  - *Even more difficult to unobtrusively implement using input obtained via eye tracker (Sadeghi et al, 2009)*
    - User shouldn't need to bother with explicit fg/bg designation
    - User should look at the object depicted in the image
    - The segmentation algorithm should take over the rest
  - *GrabCut Segmentation*

Identifying the "Object"
GrabCut Segmentation Algorithm (1/3)

- GrabCut
  - *Interactive foreground object extraction algorithm*
  - *Demonstrates exceptional extraction quality*
  - *Requires minimal user effort*
- Input
  - *A rectangular area around the object*
    - Pixels inside → certain foreground
    - Pixels outside → certain background
  - *More elaborate interactions available*
    - Explicit fg/bg designation supported

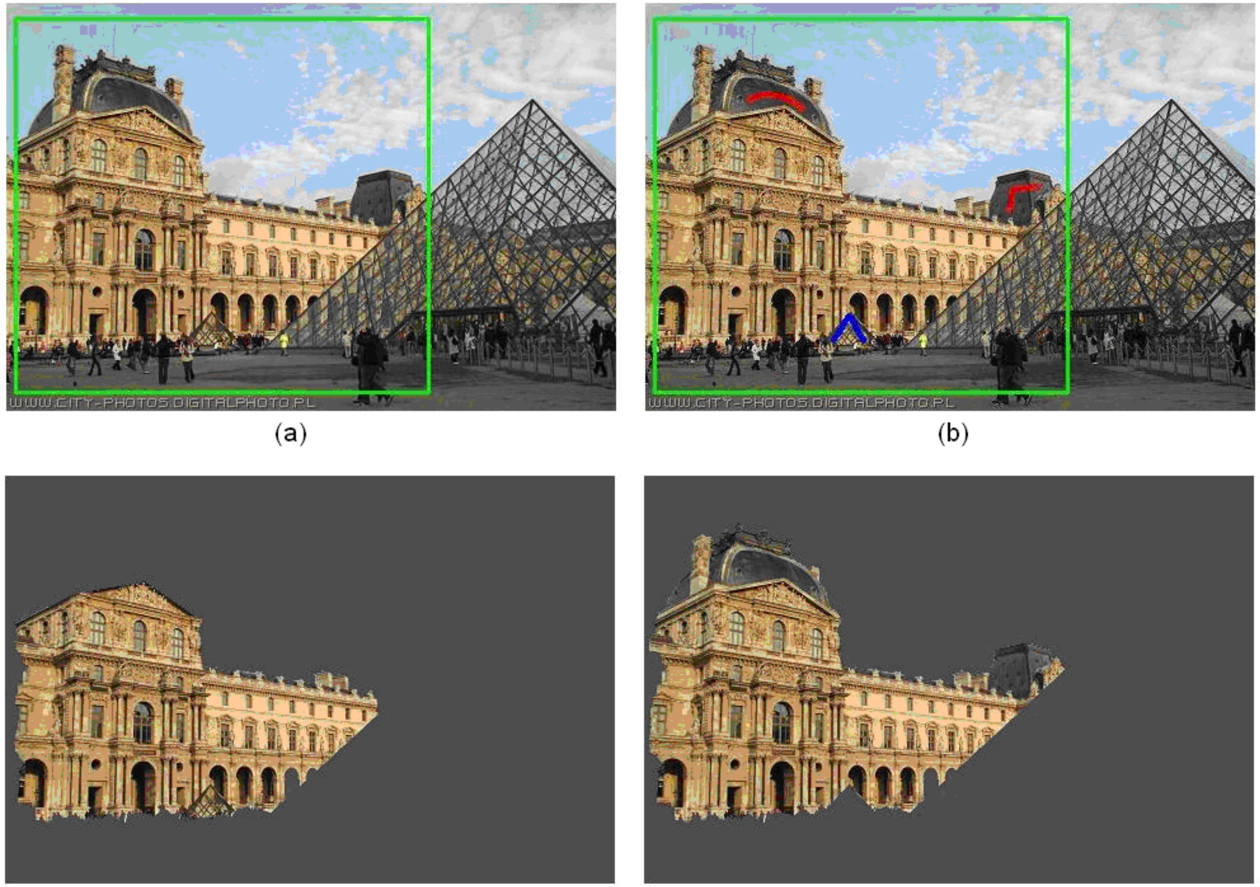
Identifying the "Object"
GrabCut Segmentation Algorithm (2/3)

- GrabCut Algorithm
  - *Image pixels outside rectangle assigned to bg class*
    - Construct Gaussian Mixture Model (GMM)*
  - *Image pixels inside rectangle assigned to fg class*
    - Construct GMM
  - *Iterate until convergence:*
    - Reassign fg pixels according to fg/bg GMMs
  - *Optional: account for user designated fg/bg pixels*

---

*Parametric probability density function represented as a weighted sum of Gaussian component densities. Among the most statistically mature methods for clustering.

Identifying the "Object"
GrabCut Segmentation Algorithm (3/3)



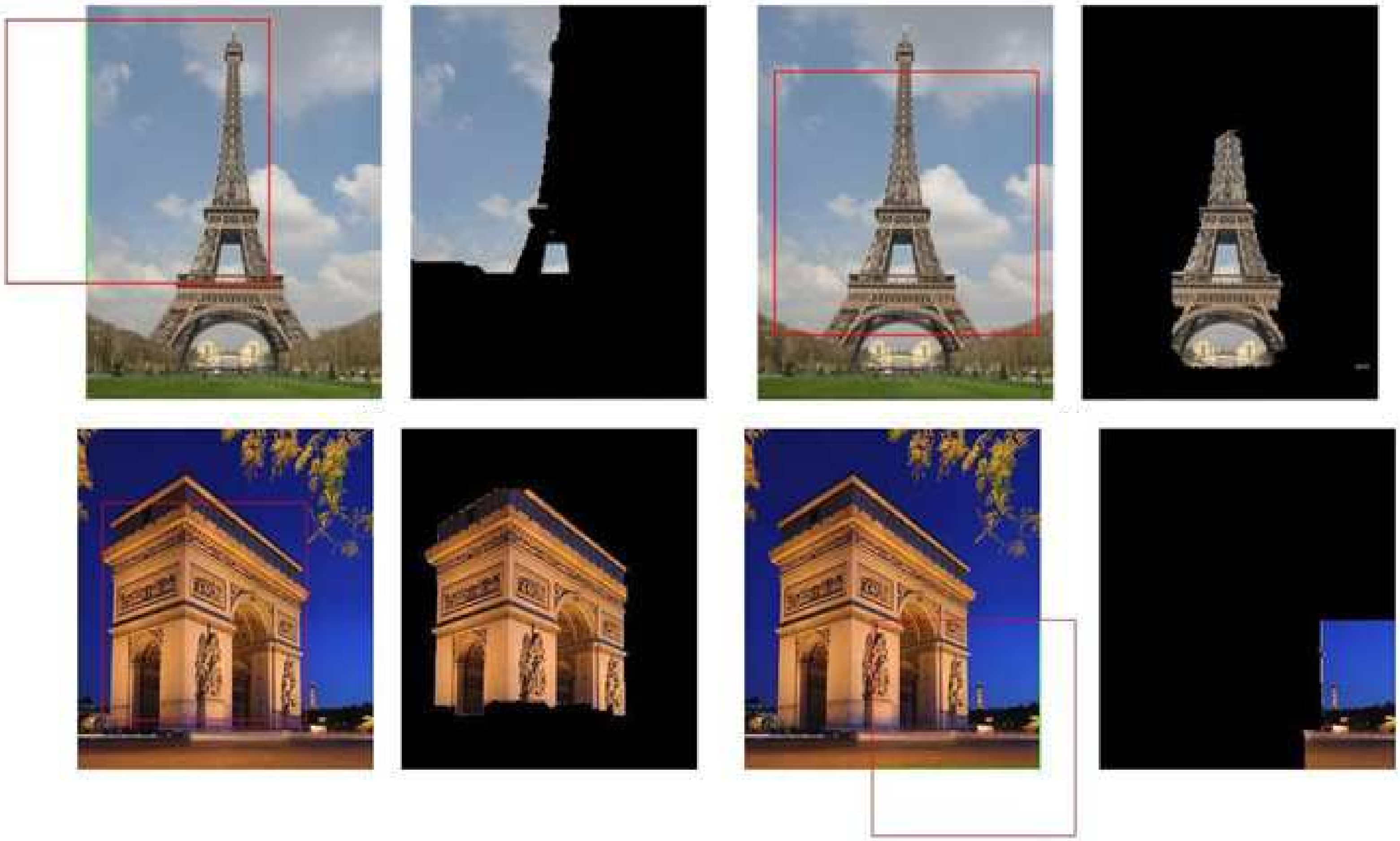
(a) (b)

(c) (d)

The Framework
Identifying the "Object" (2/3)

- Application to Framework
  - *When the user's gaze point on screen is found to intersect one of the images displayed, a rectangular ROI is automatically generated around it*
    - Ensures the unhindered process of image annotation
- Shortcomings & Improvements
  - *Excessive or incomplete segmentations when object is non-convex or not entirely contained within ROI*
    - Solution: modify ROI width & height (mouse wheel)
    - Modified GrabCut versions (Chen et al, 2008)

The Framework
Identifying the "Object" (3/3)

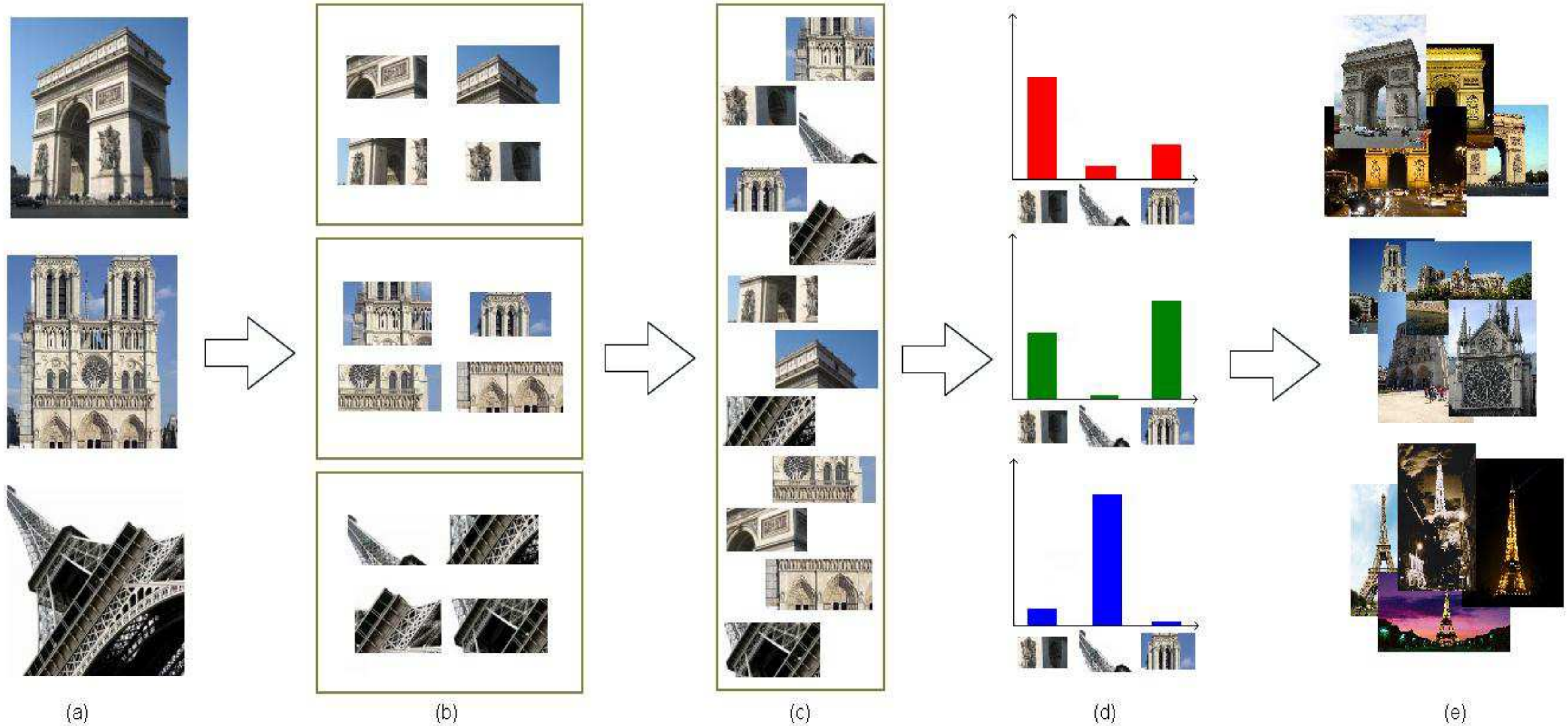


(a) (b)

The Framework
Recognition & Retrieval

- In order to annotate images containing the “Object”, the latter needs to be recognized
- Standard **Bag of Words** pipeline (BoW)
  - *Bag of Features*
  - *Bag of Keypoints*
- Pipeline consists of 3 stages
  - *Region descriptors of the image are obtained*
  - *Descriptors projected onto vocabulary → codebook frequency histograms*
  - *Classification of histograms*

Recognition & Retrieval
Bag of Words Pipeline (1/4)



Recognition & Retrieval
Bag of Words Pipeline (2/4)

- Obtaining Image Region Descriptors
  - *Features describe extracted local image patches called image descriptors*
  - *SIFT – Scale Invariant Feature Transform*
    - Spatial descriptor constructed out of 4x4 image sub-regions
    - Responses are Gaussian derivatives
    - Achieves best performance (matching / recognition)
  - *SURF – Speeded Up Robust Features*
    - Based on SIFT
    - Responses are simple operations (sums / subs)
    - Faster feature detection & descriptor extraction

Recognition & Retrieval
Bag of Words Pipeline (3/4)

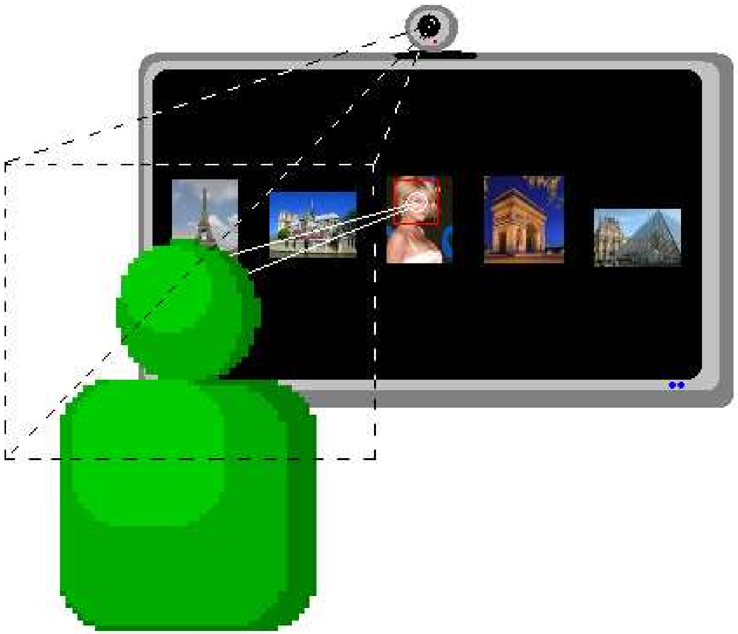
- Visual Vocabulary
  - *Trained from a set of descriptors (SURF) extracted in a previous step*
  - *Once all train descriptors have been added → clustering via **kmeans** produces cluster centres*
- Descriptor projection onto Visual Vocabulary
  - *Each descriptor matched to the nearest visual word (cluster centre) in the vocabulary*
  - *Result is a frequency histogram*
    - i-th bin of histogram → frequency of i-th vocabulary word in the image

Recognition & Retrieval
Bag of Words Pipeline (4/4)

- Histogram Classification
  - *Choice of classifier*
    - Naïve Bayes Classifier
      - *Benchmark for both accuracy & performance*
    - Support Vector Machine (SVM)
      - *Based on x^2 kernel*
        - Most accurate results
        - Not the fastest option
      - *Based on Radial Basis Function (RBF) kernel*
        - Nearly achieve real-time performance
        - Accuracy loss of approx. 10%
- BoW → Best results on large scale benchmarks

Experimental Results

- Application Development
  - *Obtaining **Paris** Database*
  - *OpenCV*[1]
  - *ASMLibrary SDK*[2]
- Framework Evaluation
  - *Implicit Tagging*
  - *Object Retrieval*
- Available via FTP[3]

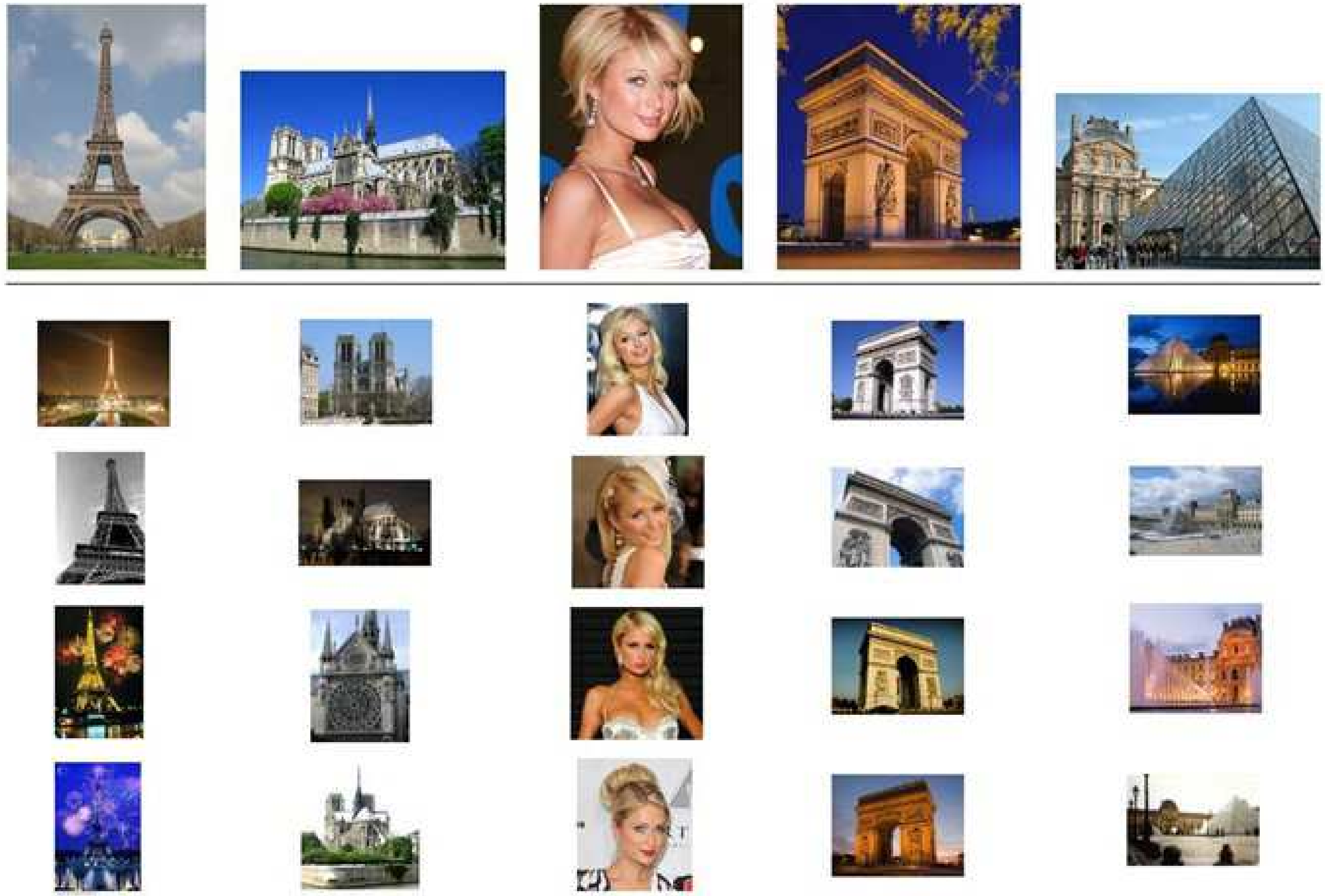


[1]Current version (2.3.1.) available from http://opencv.willowgarage.com/wiki/ under a BSD license.
[2]Current version (6) available under the MIT license from http://code.google.com/p/asmlibrary
[3]http://ftp.iti.gr/pub/incoming/sentiment.zip

Experimental Results
Application Development (1/3)

- *Paris* Database
  - *In-house*
  - *Self-obtained*
  - *Most frequently appearing distinct image categories returned by Google Images when "**Paris**" is typed in*
  - *1125 images split into 5 categories (225 images)*
    - Eiffel Tower
    - Paris Hilton
    - Notre Dame
    - Louvre
    - Arc de Triomphe

Experimental Results
Application Development (2/3)



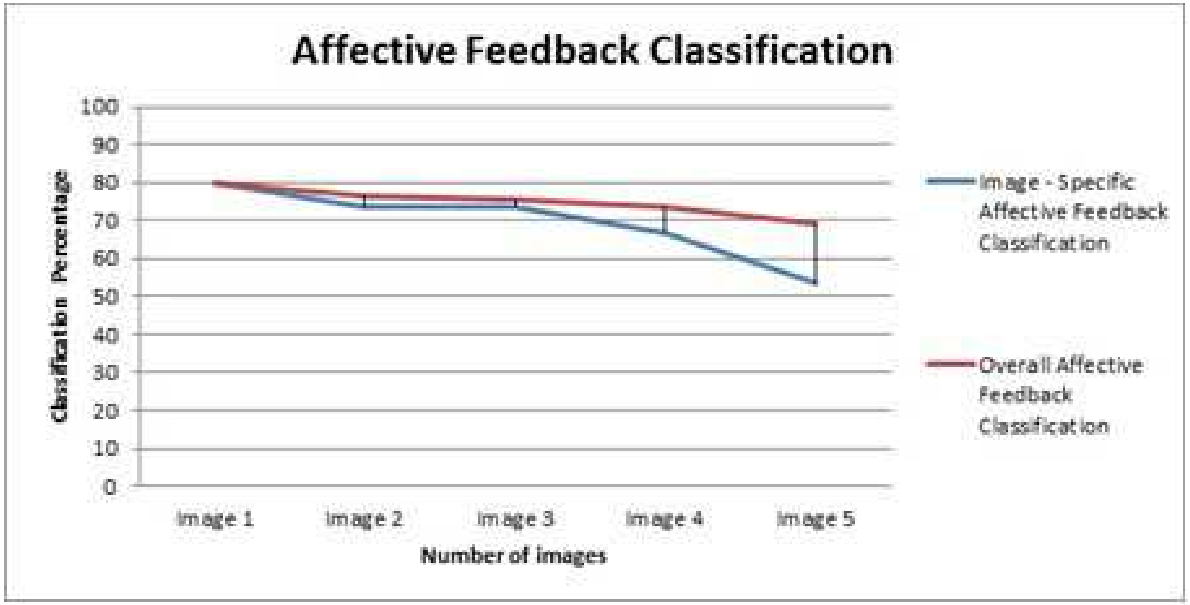
(a) (b) (c) (d) (e)

Experimental Results
Application Development (3/3)

- Implementation
  - Affective Response recognition
    - Face detection via OpenCV Haar cascades
    - ASM Fitting via ASMLibrary on detected sub-image
  - Single Image Gaze Tracking
    - 8-point calibration
  - Segmentation
    - Automatic ROI generation
  - BoW
    - 4096-word dictionary
    - RBF-kernel SVM classifier

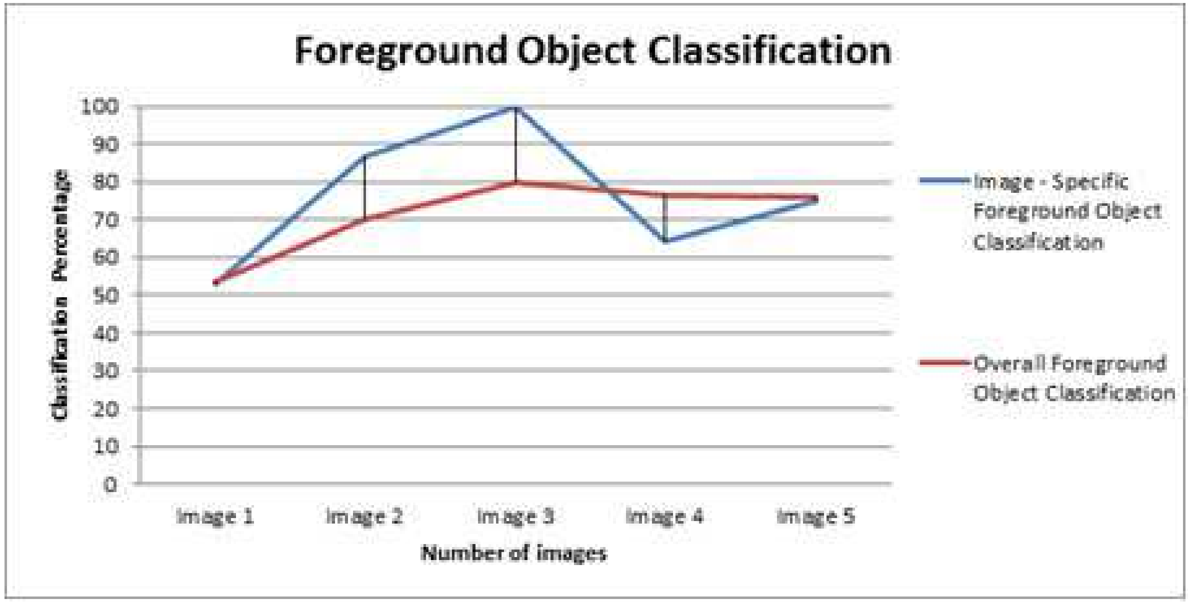
Experimental Results
Affective Feedback Classification

- 15 participants
- Results show the framework achieves an approximate 70% correct affective feedback classification performance

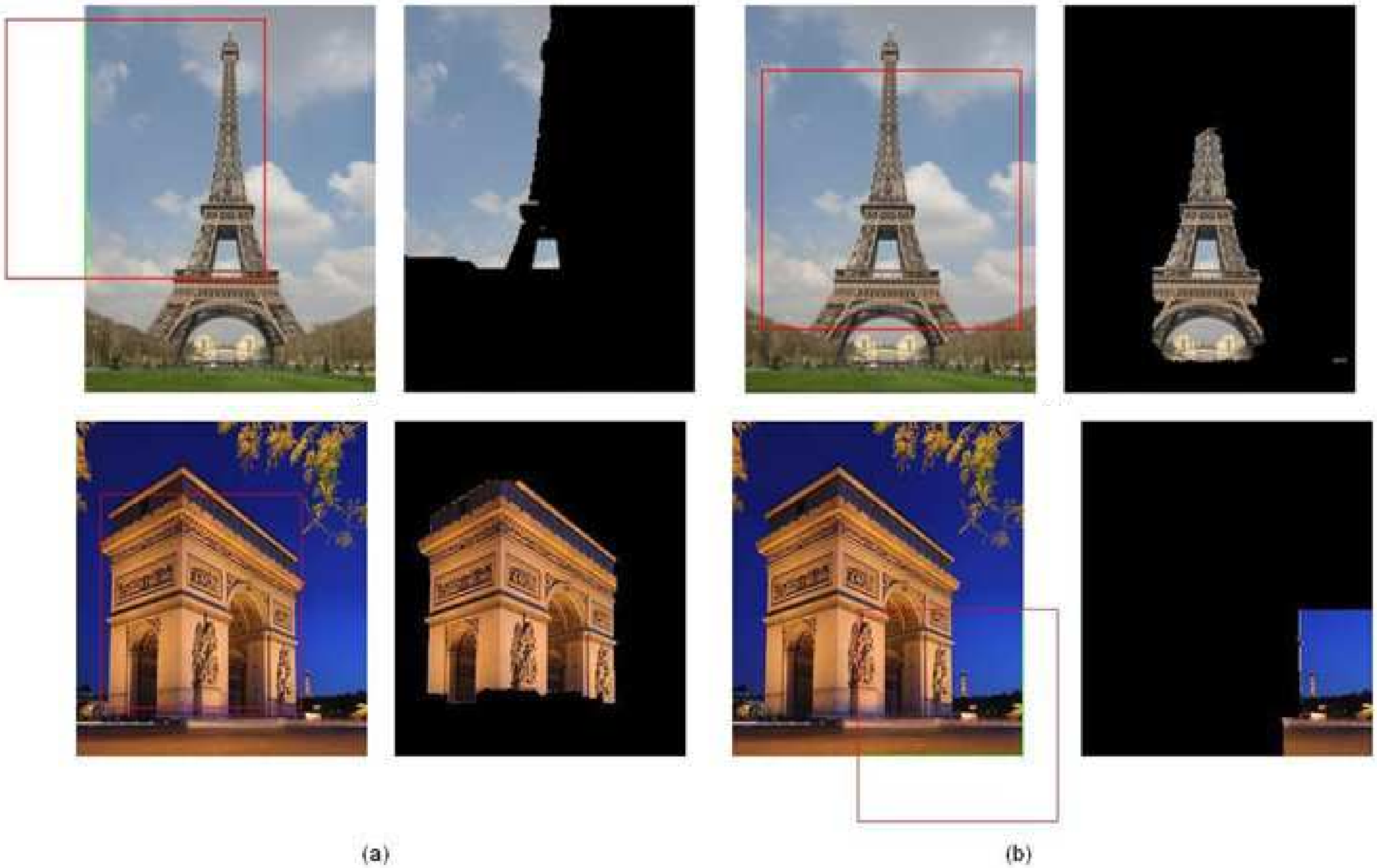


Experimental Results
Foreground Object Classification

- 95% of the images undergone segmentation were classified to one of the 5 available categories
- Overall classification performance approximately reaches 76%

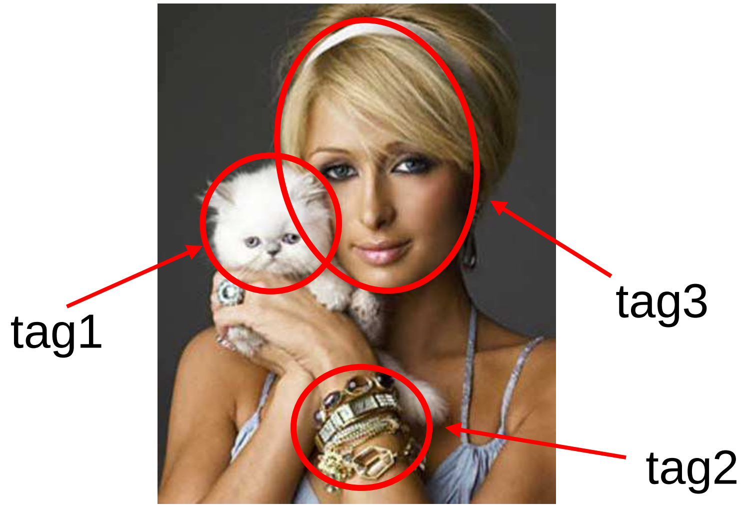


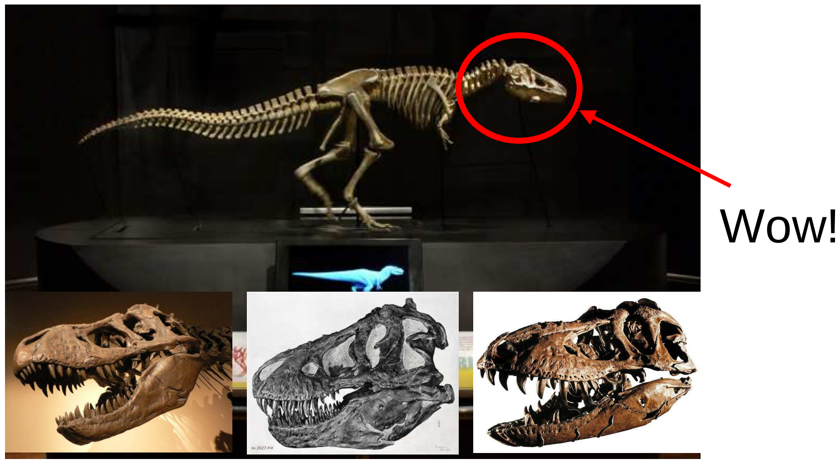
Experimental Results
Why the significant drop?



Future Endeavours…

- Improvements
- Framework applicability
  - *Content-based Recommender Systems*
  - *Tagging & Retrieval on Complex Image Scenes*
  - *Object recognition and display in immersive 3D environments*





Thank You!
Questions?