A Fuzzy Approach to Deal with Uncertainty in Image Forensics

M. Barni, A. Costanzo

Department of Information Engineering
University of Siena, Italy

October 16, 2012

Visual Information Processing and Protection
Department of Information Engineering
University of Siena

The plan

1. Introduction
2. Foundations of Fuzzy Logic
3. Fuzzy Inference Systems
4. Towards Image Forensics scenarios
5. The proposed approach
6. Experimental validation

Introducing: The Boomerang^{TM}

"GUESS WHAT? MY BOOMERANG CAME BACK TODAY!"

Introduction

Introduction: the first problem I

- In the past years many techniques have been developed to identify common image manipulations [1, 2]
  - single and multiple compressions [3, 4]
  - resampling [5, 6]
  - ...
- Some of them have been used to detect common forgeries (and others have been specifically developed)
  - cut & paste [7, 8, 4]
  - copy & move [9, 10]

Introduction: the first problem II

Usually each technique is designed to detect a single type of manipulation

- Large number of specialized algorithms looking for one or more specific footprints under precise setting

Introduction: the first problem II

Usually each technique is designed to detect a single type of manipulation

- Large number of specialized algorithms looking for one or more specific footprints under precise setting

Most of the times an edited (or tampered) image is the result of the application of multiple processing tools

- Even *"non-expert users"* can resort to resampling, cropping, color/contrast enhancement . . .

Introduction: the first problem III

- Suppose that a forensic analyst wants to decide on the authenticity of an image (or of a region of it)
  - occurred processing chain not known beforehand

A single image forensic tool is not enough, use more

- each tool provides an output describing the degree of presence of the specific footprint
- many, heterogeneous, conflicting, mutually exclusive ouputs

- *"How to make a final decision on authenticity by starting from each partial answer provided by the tools?"*

Introduction: the first problem IV

- Classic techniques may not provide satisfactory results
  - Majority: image tampered if the majority of tools say so
    - ▷ fails if there are mutually excluding tools
  - Binary OR: image tampered if at least one tool says so
    - ▷ fails if there is a tool plagued by high false positive rate
- Learning techniques, although quite effective, become rapidly unfeasible
  - SVM, Neural Networks: computational burden of training and testing as the number of tool increases

Introduction: the first problem IV

- Classic techniques may not provide satisfactory results
- Learning techniques, although quite effective, become rapidly unfeasible

Our first goal

To devise a sound strategy to elaborate (i.e. to *fuse*) into a **single global output** the heterogeneous information provided by the different tools

Introduction: the second problem I

- Like all realistic processes and systems, forensic techniques are far from being perfect
- Measurements can be affected by noise, ambiguity or impreciseness, behaviors can be unexpected
- Let us refer to all of this with "*uncertainty*"
  - noisy inputs, incomplete or not fully trustable outputs
- Several causes
  - image characteristics (e.g. color space, compression)
  - wrong tool settings
  - partial presence (or absence) of the feature(s)
  - deviation from the working assumptions

Introduction: the second problem II

- Problem more complicated when using multiple tools
- Each tool brings its contribution to the final decision as well as to the total uncertainty

Introduction: the second problem II

- Problem more complicated when using multiple tools
- Each tool brings its contribution to the final decision as well as to the total uncertainty

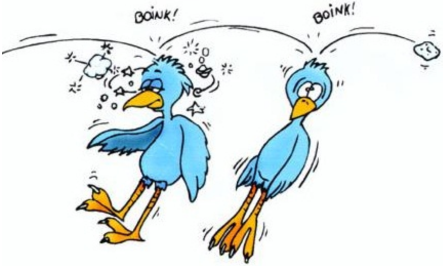
Our second goal

To devise a sound strategy to handle the **uncertainty** introduced by error-prone tools

Introduction: the proposed solution I

We propose a fusion framework based on Fuzzy Logic to decide on authenticity of a given region within an image

- Why Fuzzy Logic?
- Two birds with a stone: success in data fusion (e.g. sensor networks) and noise reduction (e.g. industrial)



Introduction: the proposed solution II

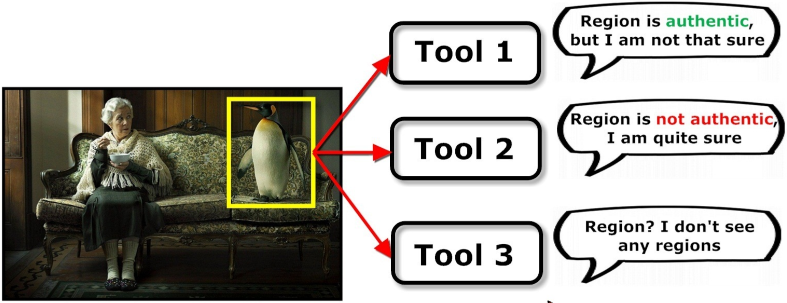
- *How would a forensic analyst face the problems of uncertainty and fusion?*
  - tweak the tools by gathering as much informations as possible
    - ▷ *on what images? how thrustworthy? what interactions?*
  - run all the tools on the image under analysis
  - exploit the gathered knowledge to make a final decision

We build a framework whose task is to mimic the analyst's behavior in the most automated way possible

Introduction: the proposed solution III

- Main strengths
  - general
  - independent from tools
  - easy to extend
  - automatic
  - no mathematical model
- Main achievements
  - we tested the method on 5 different tools looking for *cut&paste* tampering
  - outperforming logical OR-based method on a realistic scenario

Introduction: the proposed solution IV



Tool 1

Region is authentic, but I am not that sure

Tool 2

Region is not authentic, I am quite sure

Tool 3

Region? I don't see any regions

Foundations of Fuzzy Logic

Why would we need Fuzzy Logic?

- It works well on practical applications

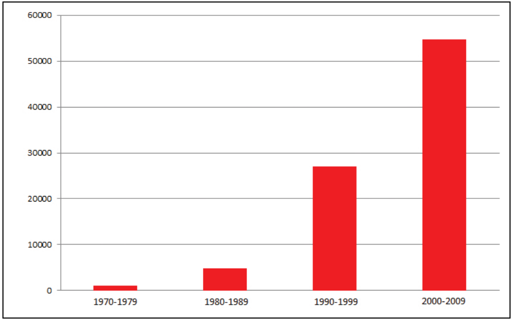
  "It is a tool that enhances our ability to deal with problems that are too complex and too ill-defined to be susceptible to solution by conventional means"

- Effective, although not excessively formal (or not at all)

  "Classical logic is like a person who comes to a party dressed in a black suit [...]. And fuzzy logic is a little bit like a person dressed informally, in jeans, t-shirt and sneakers. In the past, this [...] wouldn't have been acceptable. Today, it's the other way around. Somebody who comes dressed to a party in the way I described earlier would be considered funny.

Quite popular topic

- Fuzzy Logic related patents as of September 2011: Japan **22541**, USA **33022**
- Publications containing the word "*fuzzy*" in the title



- `http://www.cs.berkeley.edu/~zadeh/stimfl.html`

The plan

Understanding the instruments

We will briefly discuss the 3 most important concepts

- Fuzzy Sets (as extension of classical sets)
- Membership functions
- Fuzzy if–then rules

We can move from sets theory to Fuzzy Logic

Using the instruments

We will briefly explore how the above concepts are put into practice by means of Fuzzy Inference Systems

Foundations of Fuzzy Logic: fuzzy sets

- Let $\mathcal{X}$ be the universe set, $\mathcal{C} \subseteq \mathcal{X}$ a classical set and $x \in \mathcal{X}$; $\mathcal{C}$ represented by *characteristic function*:

$$\mu_{\mathcal{C}}(x) = \begin{cases} 1 & \text{if } x \in \mathcal{C} \\ 0 & \text{otherwise} \end{cases}$$

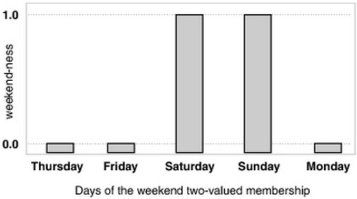
- A fuzzy set $\mathcal{F} \subseteq \mathcal{X}$ is defined through a *generalized characteristic function* [11, 12]:

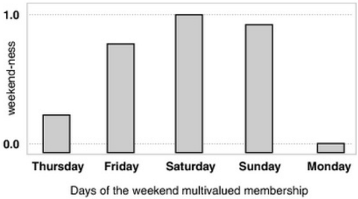
$$\mu_{\mathcal{F}}(x) : \mathcal{X} \to [0, 1]$$

- The function $\mu_{\mathcal{F}}(x)$ is called *membership function* (MF) and associates to each element $x \in \mathcal{X}$ a grade of membership that is a real number in the interval $[0, 1]$

Fuzzy Sets: example

- *Should Friday be considered weekend or not?*
- If we are to to respond with an absolute response: "*Well no, it isn't*"





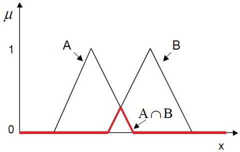
- If we are allowed to respond with fuzzy in-between values: "*Quite yes, but not completely*". So does Sunday.

Fuzzy Sets: Operations I

- Usual operations on crisp sets can be extended
- Let $\mathcal{X}$ be the universe set; let $A, B \subseteq \mathcal{X}$ be two fuzzy sets and $\mu_A(x)$, $\mu_B(x)$ their membership functions

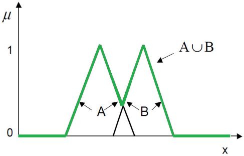
intersection

$$\mu_{A \cap B}(x) = min\ (\ \mu_A(x), \mu_B(x)\ )$$



union

$$\mu_{A \cup B}(x) = max\ (\ \mu_A(x), \mu_B(x)\ )$$

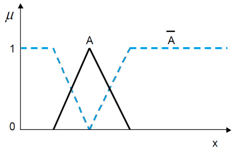


Fuzzy Sets: Operations II

- Let $\mathcal{X}$ be the universe set; let $A, B \subseteq \mathcal{X}$ be two fuzzy sets and $\mu_A(x)$, $\mu_B(x)$ their membership functions

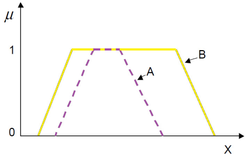
complement

$$\mu_{\bar{A}}(x) = 1 - \mu_A(x)$$



inclusion

$$\mu_{A \subseteq B}(x) \Leftrightarrow \mu_A(x) \leq \mu_B(x)$$

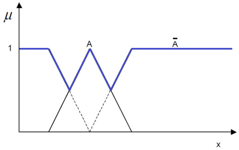


Fuzzy Sets: Operations III

- Let $\mathcal{X}$ be the universe set; let $A, B \subseteq \mathcal{X}$ be two fuzzy sets and $\mu_A(x)$, $\mu_B(x)$ their membership functions

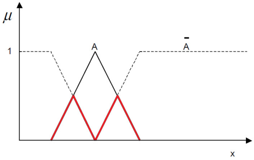
no middle

$\bar{A} \cup A \neq \mathcal{X}$



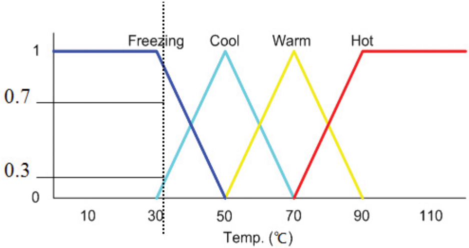
no contradiction

$\bar{A} \cap A \neq \varnothing$



Foundations of Fuzzy Logic: fuzzy variables

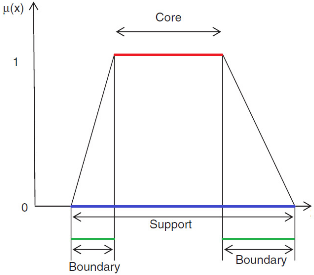
- x used so far is commonly called *fuzzy variable* or *linguistic variable*, i.e. a variable whose values are linguistic terms
- x is defined by the labels (names) of fuzzy sets



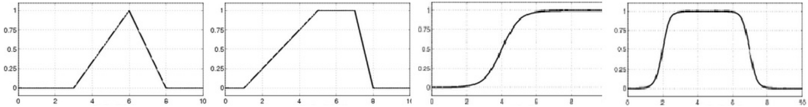
$Temperature$ $x = 36° C$; $Labels = \{Freezing, Cool, Warm, Hot\}$ $\mu_{Freezing}(36) = 0.7$; $\mu_{Cool}(36) = 0.3$

Foundations of Fuzzy Logic: membership functions

- A *membership function (MF)* consists of three parts: `core`, `support`, `boundary`



- The are many possible shapes, the most common being:



Foundations of Fuzzy Logic: from sets to logic

- Sets theory extended to multi–valued fuzzy logic, formally requiring t-norm, t-conorm, residuum . . .
- Classic Boolean logic: proposition true (1) or false (0)

> Fuzzy logic: proposition not always totally false (true) but false (true) to some grade in $[0, 1]$

- Extension of logical operators is quite simple
  - $\mu_{A \wedge B}(x) = min\,(\,\mu_A(x), \mu_B(x)\,)$
  - $\mu_{A \vee B}(x) = max\,(\,\mu_A(x), \mu_B(x)\,)$
  - $not(A) = 1 - \mu_A(x)$

Foundations of Fuzzy Logic: if-then rules I

IF-THEN rules define how *fuzzy sets* and *logic operators* interact with each other by means of *membership functions*.

- Simplest and most common form:

  IF *assignment* 1 **AND/OR** . . . **AND/OR** *assignment M*
  THEN *assignment Y*

- Composed by antecedent and consequent
  - description of a situation / action to be performed
  - connected with $\wedge$, $\vee$, $\neg$
- Can be way more complex: nested structures IF-THEN-ELSE

Foundations of Fuzzy Logic: if-then rules II

- Rarely we express ourselves in binary terms
- Often our informations are approximated and imprecise
- IF-THEN rules allow to describe such approximations
  - **IF** you are *fairly hungry* **THEN** put *some* pasta
  - **IF** you are *really hungry* **THEN** put *some more* pasta
  - **IF** you are *really hungry* **AND** *many* friends are coming **THEN** put *a lot of* pasta
  - **IF** you are *not hungry* **OR** you don't feel *too well* **THEN** put *not too much* pasta

If-then rules: assumptions

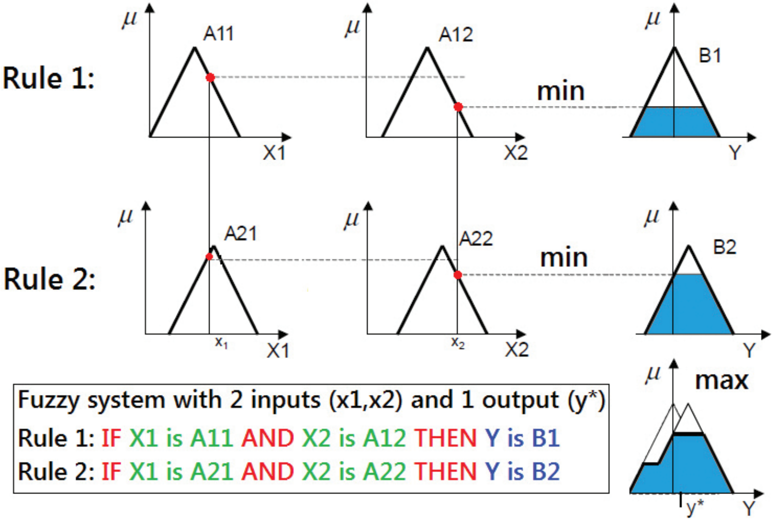
We will work with a set of n rules in MISO (*Multi-Input Single-Output*) form, that is m inputs and 1 output

Rule 1: **IF** X_1 is A_{11} **AND** … **AND** X_m is A_{1m} **THEN** Y is B_1
Rule 2: **IF** X_1 is A_{21} **AND** … **AND** X_m is A_{2m} **THEN** Y is B_2
…
Rule n: **IF** X_1 is A_{n1} **AND** … **AND** X_m is A_{nm} **THEN** Y is B_n

We will resolve the rules with a FITA (*First Infer Then Aggregate*) approach

- implication: $strength_n = min(\ \mu_{An1}(x_1), \dots, \mu_{Anm}(x_m)\ )$
- aggregation: $B' = max(\ strength_1, \dots, strength_n\ )$

If-then rules: example of composition

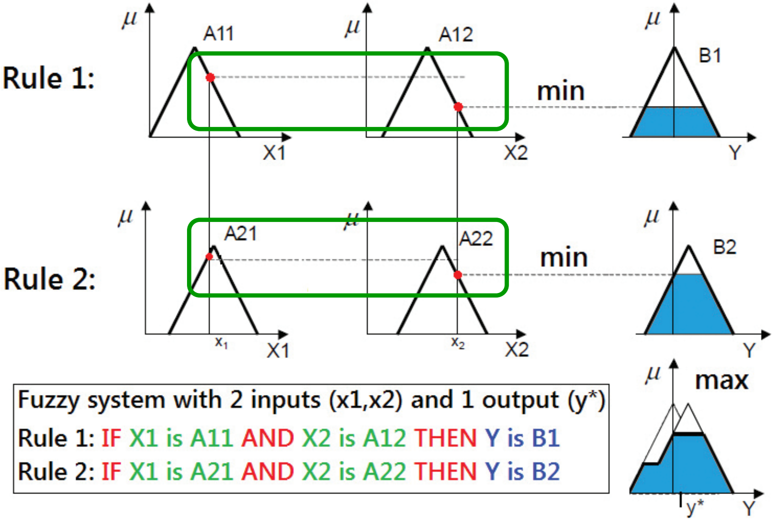


Fuzzy system with 2 inputs (x1,x2) and 1 output (y*)

Rule 1: IF X1 is A11 AND X2 is A12 THEN Y is B1

Rule 2: IF X1 is A21 AND X2 is A22 THEN Y is B2

If-then rules: example of composition

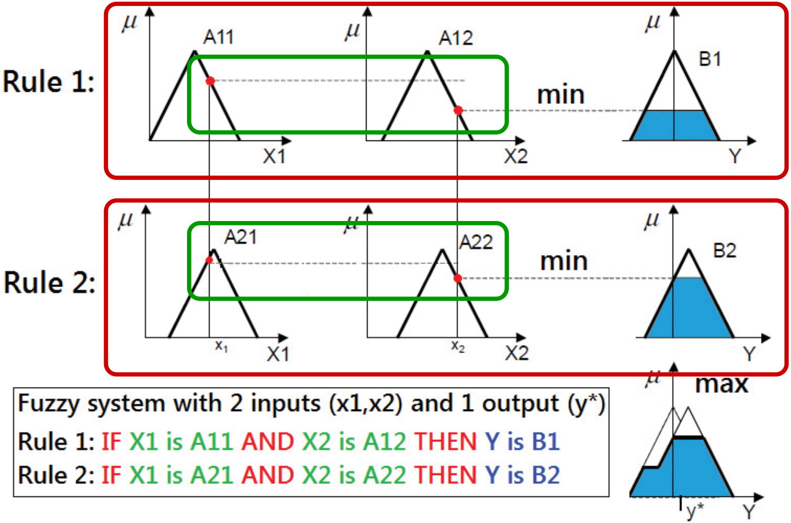


Fuzzy system with 2 inputs (x1,x2) and 1 output (y*)

Rule 1: IF X1 is A11 AND X2 is A12 THEN Y is B1

Rule 2: IF X1 is A21 AND X2 is A22 THEN Y is B2

If-then rules: example of composition

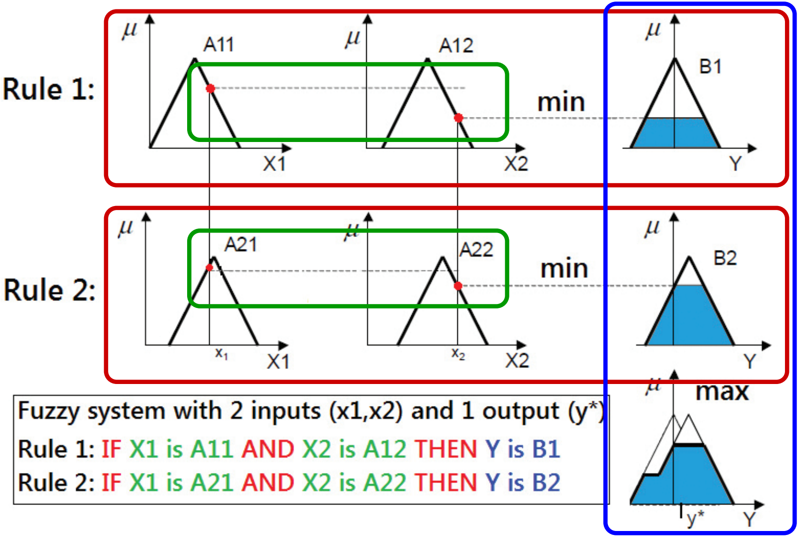


Fuzzy system with 2 inputs (x1,x2) and 1 output (y*)

Rule 1: IF X1 is A11 AND X2 is A12 THEN Y is B1

Rule 2: IF X1 is A21 AND X2 is A22 THEN Y is B2

If-then rules: example of composition



Fuzzy system with 2 inputs (x1,x2) and 1 output (y*)

Rule 1: IF X1 is A11 AND X2 is A12 THEN Y is B1

Rule 2: IF X1 is A21 AND X2 is A22 THEN Y is B2

If-then rules: another example of composition

Toy example: controller deciding speed (S) according to hour of day ($H \in$ [00:00,23:59]) and intensity of fog ($F \in [0, 100]\%$)

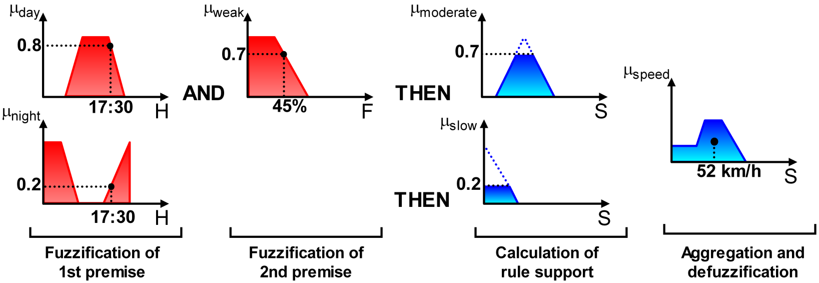
Desired behavior: "Speed should be *moderate* during *day* when fog is *weak* and *slow* during *night* regardless of fog"

Rules:

- **IF** H is *day* **AND** F is *weak* **THEN** S is *moderate*
- **IF** H is *night* **THEN** S is *slow*

Inputs: H = 17:30 and $F = 45\%$. **Output**: $S = ?$

If-then rules: another example of composition



μday
0.8
17:30
H
μnight
0.2
17:30
H
AND
μweak
0.7
45%
F
THEN
μmoderate
0.7
S
THEN
μslow
0.2
S
μspeed
52 km/h
S
Fuzzification of 1st premise
Fuzzification of 2nd premise
Calculation of rule support
Aggregation and defuzzification

$$\mu_{day}(17{:}30) \wedge \mu_{weak}(45) = \mathtt{min}(0.7, 0.8) = 0.7$$
$$\mu_{night}(17{:}30) = 0.2$$
$$support = \mathtt{max}(\mu_{moderate}, \mu_{slow})$$
$$S = \mathtt{centroid}(support)$$

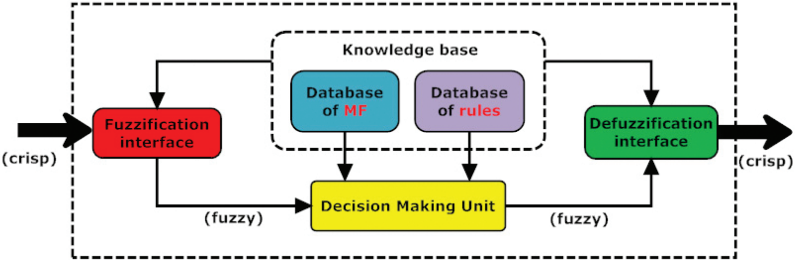
Fuzzy Inference Systems

Fuzzy inference systems (FIS): Why so popular?

- Allow robust reasoning against noise, approximate or imprecise inputs
- Address problems whose mathematical or statistical models are hard to define
- Resort to the experience and the knowledge of human operators to mimic their behavior
- Very intuitive building, similar to natural language

Fuzzy inference systems: what

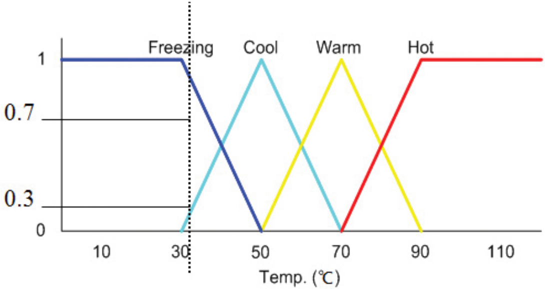
- Intuitively: *a set of fuzzy rules that converts inputs to outputs*
- Specifically: a system consisting on the following parts



Fuzzy inference systems: Fuzzification Interface

Crisp (numerical) inputs are converted into fuzzy quantities. A degree of membership is assigned by means of membership functions.

$$t = 36^\circ C \rightarrow \mu_{freezing}(36) = 0.7,\ \mu_{cold}(36) = 0.3,\ \mu_{hot}(36) = 0$$



Fuzzy inference systems: Knowledge Base

A database which contains all the membership functions and all the fuzzy rules that the system can use

- Contains all the informations (experience) that a human operator has gathered
- A single rule is generally not enough. There is need of more than one rule playing off each other
- A system can easily feature several hundreds rules

Fuzzy inference systems: Decision Making Unit

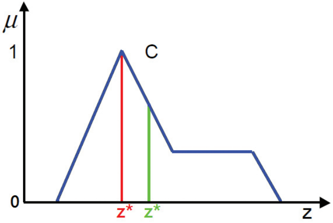
The heart of a FIS, performs the reasoning by interpreting each fuzzy rule and then aggregating the results.

- Reasoning works as follows:
    1. Evaluating the rules (applying $\wedge$, $\vee$, $\neg$ operators)
    2. Applying the result of (1) by truncating the consequent (result is a fuzzy set called *strength of the rule*)
    3. Aggregating all consequents (result is a fuzzy set)

Fuzzy inference systems: Defuzzification Interface

Conversion from fuzzy quantities to a numerical value.

- The result of aggregation is still a fuzzy set. However, one needs a numerical output to make a decision.



(1) Max membership

(2) **Centroid**

(3) Weighted average

(4) First (last) of maxima

(5) Center of sums (of largest area)

- Choice may depend on MF symmetry, on computational burden or on specific application

Fuzzy Inference Systems: countless applications

- Industrial
  - power plants, water treatment, incineration plants . . .
- Automatic control
  - vehicle controllers, traffic monitoring, robot navigation . . .
- Biomedics
  - anesthetic depth control, disease diagnostics . . .
- Image processing
  - edge detection, denoising, contrast enhancement, smoothing, segmentation, image forensics . . .
- Signal processing and data mining
  - clustering, feature selection, partitioning, pattern recognition, sensor networks . . .

Towards Image Forensics scenarios

Towards image forensics scenarios: a reminder I

First problem

To devise a sound strategy to elaborate (i.e. to *fuse*) into a **single global output** the heterogeneous information provided by the different tools

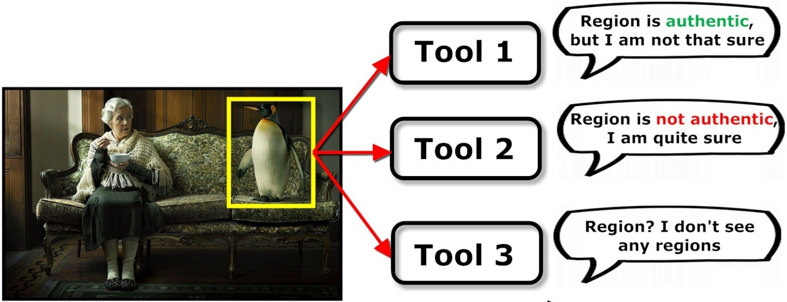
Second problem

To devise a sound strategy to handle the **uncertainty** introduced by error-prone tools

Our solution

We propose a fusion framework based on Fuzzy Logic to decide on authenticity of a given region within an image

Towards image forensics scenarios: a reminder II



Tool 1

Region is authentic, but I am not that sure

Tool 2

Region is not authentic, I am quite sure

Tool 3

Region? I don't see any regions

Towards image forensics scenarios: literature

- **Fusion** categorized in steganalysis: 3 main approaches to merge the outputs of several tools [13]
    1. `Feature level`: aggregation of all the features before actually taking a final decision (e.g. with SVM)
    2. `Measurement level`: each tool makes a partial decision by relying only on its features, all the partial detection scores are aggregated into a global score
    3. `Abstract level`: threshold to all partial scores separately, aggregate binary values into a global value
- **Uncertainty** largely unexplored in image forensics
    - just one technique [14] relying on fuzzy integrals

Towards image forensics scenarios: literature

- **Fusion** categorized in steganalysis: 3 main approaches to merge the outputs of several tools [13]

2. Measurement level: each tool makes a partial decision by relying only on its features, all the partial detection scores are aggregated into a global score ← our choice!

- **Uncertainty** largely unexplored in image forensics
  - just one technique [14] relying on fuzzy integrals ← direct comparison unfeasible!

The proposed approach

Formalization: variables

- K tools, each analyzing a set of features in a specified region of an image I looking for tampering traces
  - K tools $= K$ outputs
  - questions: *is the trace present? Is the tool sure about it?*
- We chose to answer with a pair (D, R)
  1. Detection $D \in [0,1]$: measure of presence of the tampering
  2. Reliability $R \in [0,1]$: measure of confidence of the tool on D

Formalization: variables

- K tools, each analyzing a set of features in a specified region of an image I looking for tampering traces
    - K tools $=$ K outputs
    - questions: *is the trace present? Is the tool sure about it?*

- We chose to answer with a pair (D, R)
    1. Detection $D \in [0, 1]$: measure of presence of the tampering
    2. Reliability $R \in [0, 1]$: measure of confidence of the tool on D

- Framework is general with respect to (D, R)

Each tool is free to choose how to calculate (D, R)!

Formalization: tampering tables I

- *"If all the tools at our disposal work as intended, what kind of output do we expect from them?"*
- Depending on the nature of the manipulation, a tool may or may not be able to detect a region as tampered
  - Y = capability of detecting a tampering trace
  - N = incapability
- If K tools, manipulation identified by K-dimensional sequences of Y and N
- Organize the sequences into tables
  - T_{true}: sequences of presence of tampering
  - T_{false}: sequences of absence of tampering
  - T_{doubt}: all the other unknown sequences

Formalization: tampering tables II

- Toy example: t_1 (t_2) considers a region with aligned (misaligned) double compression as tampered
- We expect that
  - aligned double compression: (Y,N)
  - misaligned double compression: (N,Y)
  - no double compression: (N,N)
  - *something strange*: (Y,Y)
- Tables will then be:

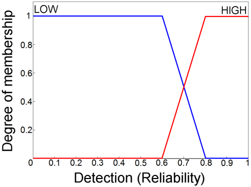
Tool	T_{true}	T_{false}	T_{doubt}
t_1	Y N	N	Y
t_2	N Y	N	Y

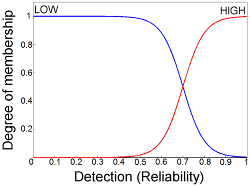
Formalization: membership functions

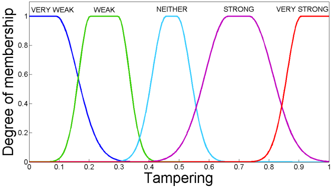
- “*We know how ideally the tools behave. But what does really happen when they are used on the field?*”
- A tool is not perfectly secure about the presence (absence) of a manipulation
  - noisy value of D high (low) but not necessarily near 1 (0)
  - same goes for R
- Fuzzy comes to the rescue. Define MF’s of fuzzy sets:
  - D and R: `low`, `high`
  - *tampering*: `very weak`,`weak`,`neither`,`strong`,`very strong`

Formalization: membership functions

- *"We know how ideally the tools behave. But what does really happen when they are used on the field?"*
- A tool is not perfectly secure about the presence (absence) of a manipulation
    - noisy value of D high (low) but not necessarily near 1 (0)
    - same goes for R
- Fuzzy comes to the rescue. Define MF's of fuzzy sets:
    - D and R: `low`, `high`
    - *tampering*: `very weak`,`weak`,`neither`,`strong`,`very strong`







Formalization: naming convention of if-then rules

- We will consider 2 categories of if–then rules: *standard* and *non standard*
- From the perspective of fuzzy Logic, conceptually are not different
- The difference resides in the fact that:
  - *standard* → derived from known behaviors (i.e. T_{true}, T_{false})
  - *non standard* → derived from unknown behaviors (i.e. T_{doubt})

Fuzzy variables and membership functions will vary according to rule's origins

Formalization: standard if-then rules – detection

- We assign to T_{true} and T_{false} a linguistic meaning
- Consider a tool capable (incapable) of detecting a manipulation if it provides a high (low) value of detection

$$\begin{aligned} \text{Y} &= \text{detection is } \textit{high} \\ \text{N} &= \text{detection is } \textit{low}. \end{aligned}$$

- D fuzzy variable, `high` and `low` fuzzy sets
- e.g. in a 4-tool scenario, $\mathbf{s}$=(Y,Y,N,N) becomes

$$D_1 \ \texttt{high} \wedge D_2 \ \texttt{high} \wedge D_3 \ \texttt{low} \wedge D_4 \ \texttt{low}$$

Formalization: standard if-then rules – reliability

- The trustworthiness of a tool (hence R) impacts the nature of the consequent
- *Do we fully trust a tool?* → most intense fuzzy set
  - `very strong` if $\mathbf{s} \in T_{true}$
  - `very weak` if $\mathbf{s} \in T_{false}$
- *We do not fully trust a tool?* → less intense fuzzy set
  - `strong` if $\mathbf{s} \in T_{true}$
  - `weak` if $\mathbf{s} \in T_{false}$
- In rules:

$$
\begin{array}{ll}
\text{IF} & \big(\, D\ \texttt{high}\big) \\
\text{THEN} & \Big[\ \text{IF } (R\ \texttt{high}) \text{ THEN tampering is } \texttt{very strong} \\
 & \quad \text{ELSE tampering is } \texttt{strong}\ \Big]
\end{array}
$$

Formalization: standard if-then rules – example I

- Exemplify first, generalize then
- Case (Y,N)$\in T_{true}$. Resulting fuzzy rule:

IF $\big(D_1\ \texttt{high} \wedge D_2\ \texttt{low} \big)$

THEN $\big[$ IF $(R_1\ \texttt{high} \wedge R_2\ \texttt{high})$ THEN tampering is $\texttt{very strong}$

ELSE tampering is $\texttt{strong}\ \big]$

- Correct, yet not standard. Split the contributions [15]:

IF $\big(D_1\ \texttt{high} \wedge D_2\ \texttt{low} \big)$

THEN $\big[$ IF $(R_1\ \texttt{high} \wedge R_2\ \texttt{high})$ THEN tampering is $\texttt{very strong}\ \big]$

IF $\big(D_1\ \texttt{high} \wedge D_2\ \texttt{low} \big)$

THEN$\big[$ IF $(\overline{R_1\ \texttt{high} \wedge R_2\ \texttt{high}})$ THEN tampering is $\texttt{strong}\ \big]$

Formalization: standard if-then rules – example II

- One last conversion step [15]:

  IF $\left(\, D_1\ \mathtt{high} \wedge D_2\ \mathtt{low} \,\right) \wedge \left(R_1\ \mathtt{high} \wedge R_2\ \mathtt{high}\right)$

  THEN tampering is $\mathtt{very\ \ strong}$

  IF $\left(\, D_1\ \mathtt{high} \wedge D_2\ \mathtt{low} \,\right) \wedge \left(\overline{R_1\ \mathtt{high} \wedge R_2\ \mathtt{high}}\right)$

  THEN tampering is $\mathtt{strong}$

- Read rule 1: "*if D_1, D_2 have a high membership and both tools are reliable, then assign most intense tampering*"
- Read rule 2: "*if one of the tools (or both) is not reliable* (recall De Morgan's law), *then assign less intense tampering*"

Formalization: generalization to K

- Generalization to K tools fairly simple: same compound structure, same reduction steps
- Main difference in the way the consequents are chosen: use majority
  - $\geq 1/2$ of the tools reliable → most intense consequent
  - $< 1/2$ of the tools reliable → less intense consequent
- Other possibilities exist
  - known subset of most reliable tools, . . .
  - majority simple yet effective

Formalization: non standard if-then rules I

- Construction similar to standard cases. However, no support from theory/experiments means further reasoning
- Idea: map T_{doubt} into something that we know
  - *what do we know?* → standard cases
  - *how do we map?* → taking into account reliability
- Set Y=1,N=0 and compute weighted Hamming distance of non standard case (**ns**) from all standard cases (**s**)

$$d(\mathbf{ns},\mathbf{s}) = \sum_{i=1}^{K} R_i \cdot \mathrm{XOR}\big(\mathbf{ns}(i),\mathbf{s}(i)\big); \;\; \mathbf{s}_{min} = \arg\min_{n=1,..,M}\big[d(\mathbf{ns},\mathbf{s}_n)\big]$$

Formalization: non standard if-then rules II

- Antecedent will be the one of **ns** constructed in the same way of standard cases
- Mapping is an experimental approximation: not wise to lean towards presence or absence of tampering
- Consequent of $\mathbf{s}_{\mathbf{min}}$ but mitigated regardless of reliability

  if $\mathbf{s}_{min} \in T_{true}$ consequent is: THEN tampering is `strong`
  if $\mathbf{s}_{min} \in T_{false}$ consequent is: THEN tampering is `weak`

- Reliability accounted in mapping, no need to exploit it again

Formalization: non standard if-then rules – example

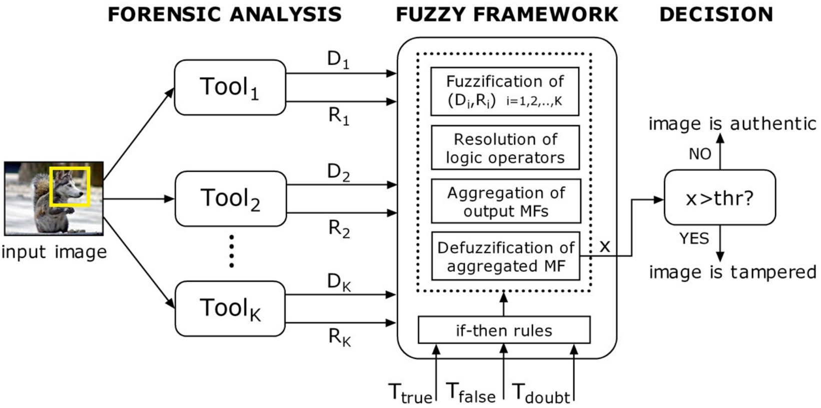
- Toy example: t_1 (t_2) considers a region with aligned (misaligned) double compression as tampered
- Doubtful case (Y,Y), suppose $s_{min} = (Y, N)$:

  IF $\left(D_1 \; \texttt{high} \wedge D_2 \; \texttt{high} \right)$

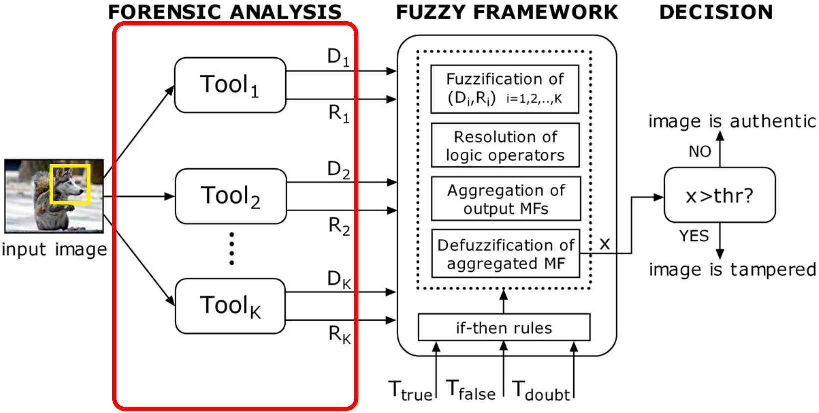
  THEN $\Big[$ *regardless of reliabilities* tampering is ~~`very`~~ `strong` $\Big]$

- Read: "**IF** (t_1 is more or less capable of detecting) **AND** (t_2 is more or less capable of detecting) **THEN** tampering is present but not as much as (Y,N)"

The proposed approach: framework



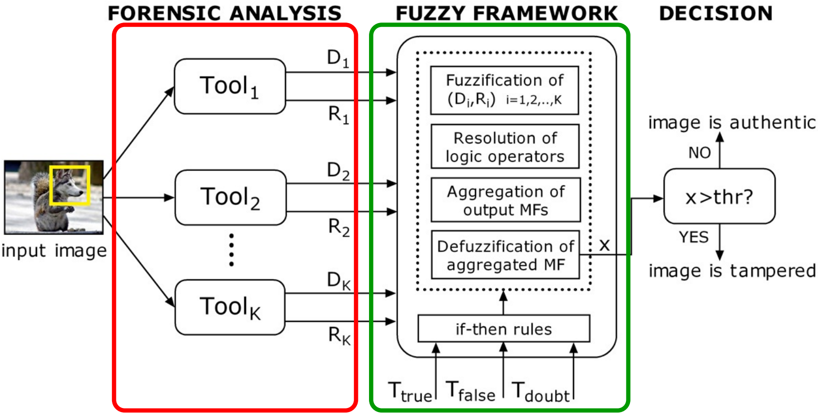
The proposed approach: framework



FORENSIC ANALYSIS OF INPUT IMAGE

K forensic tools analyze the image providing *K* (D,R) pairs

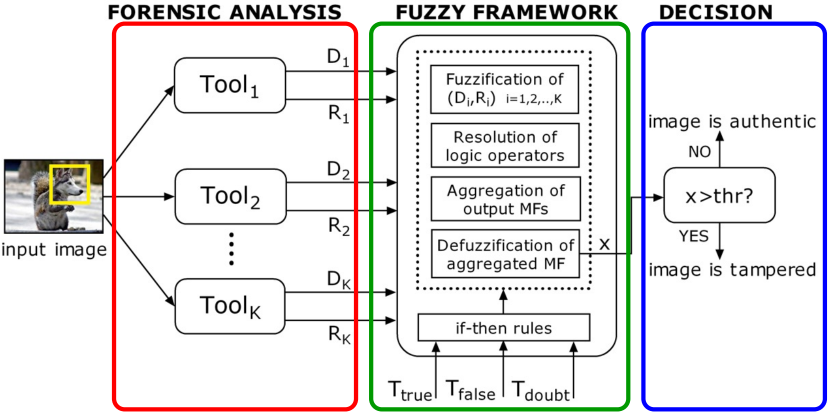
The proposed approach: framework



CONSTRUCTION OF THE INFERENCE SYSTEM

if-then rules are built accordingly with T_{true}, T_{false} and T_{doubt}

The proposed approach: framework



DECISION ON IMAGE'S AUTHENTICITY

crisp value *x* of tampering presence vs a threshold

Experimental validation

Experimental validation: employed tools I

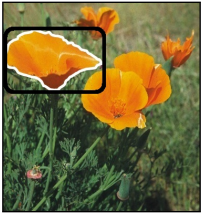
- 5 methods exploiting JPEG compression characteristics to discriminate between single and double compression

Tool	Investigated feature
t_A [7]	Statistical analysis of image blockiness
t_B [8]	Double Quantization (DQ) effect
t_C [4]	Ghost effect (coefficients previously compressed with a higher quantization step)
t_D [16]	Integer periodicity of the DCT coefficients
t_E [17]	Probability models for DCT coefficients

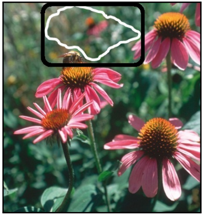
- t_D improves t_A, t_E improves t_B

Experimental validation: employed tools II

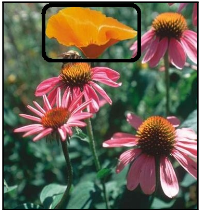
- JPEG artifacts and number of compressions can be used to detect *cut & paste* tampering
- Common forgery whereby a portion of a source image is cut and pasted into another target image



source



target



tampering

Experimental validation: employed tools III

- *How do the tools detect cut & paste?*

Tool	Region cropped from...	Tampering if ...
t_A	JPEG image and pasted **without preserving grid alignment** on another JPEG image	region with misaligned grids ($QF_2 > QF_1$)
t_B	JPEG or uncompressed image and pasted **preserving grid alignment**	region without double quantization effect
t_C	JPEG image and pasted **preserving grid alignment**	region with JPEG ghost effect
t_D	JPEG image and pasted **without preserving grid alignment** on another JPEG image	region with DCT periodicity above threshold
t_E	JPEG or uncompressed image and pasted **preserving grid alignment**	region compressed twice (probability model)

Experimental validation: tools' interactions I

- 4 classes of tampered images for which tools ideally provide different 5-uples of answers
- From principles underlying the tools and preliminary experimental analysis

Tool	Class 1	Class 2	Class 3	Class 4	Class 5
t_A	Y	Y	N	N	N
t_B	N	Y	N	Y	N
t_C	N	Y	Y	Y	N
t_D	Y	Y	N	N	N
t_E	N	Y	N	Y	N

- Columns 1–4: T_{true}; column 5: T_{false}; not listed: T_{doubt}

Experimental validation: tools' interactions II

Class	Tampering procedure
Class 1	Outer region is compressed once. Inner region is compressed twice with misaligned grids
Class 2	Outer region is compressed twice with aligned grids. Inner region is compressed twice with misaligned grids
Class 3	Outer region is compressed once. Inner region is compressed twice with aligned grids
Class 4	Outer region is compressed twice with aligned grids. Inner region is compressed once
Class 5	Non-tampered images. The image is compressed once with a random but fairly high quality factor: $QF \in \{70, 75, 80, 90\}$

Tampering classes. Each class has been created by varying the number of compression steps with aligned or non-aligned grids. The fifth class corresponds to non-tampered images.

Experimental validation: 3 data sets

- 3 data sets: 2 synthetic and 1 natural
- General procedure common to both synthetic data sets
  - *Cut & paste* of the central 256×256 region
  - 4 classes obtained by slightly variating the procedure
    - ▷ region single/double compressed
    - ▷ region's JPEG grids aligned/misaligned
    - ▷ different quality factors (QF_1, QF_2)
  - Tests on central 256×256 region with 2 different data sets

Experimental validation: first data set

- Starting from 100 uncompressed TIFF images with different visual content (landscapes, people, macros)
- Each original image has been used to create 2 tampered images according to the procedure described above
  - 200 fakes per class, 800 fakes total
  - adding 800 non-tampered images that have been simply compressed once
  - 1600 total images

Experimental validation: second data set

- Derives from the observation of a peculiar behavior of t_B
  - tends to claim as tampered images with textures and regular geometric shapes compressed once with very high quality factor
  - e.g. buildings, walls, squares
- Common subjects in real-world, introducing doubtful cases
- Starting from 50 natural images whose central region has textured / geometric content
  - creating 200 original and 200 fakes (50 per class)
  - according to previously defined classes

Experimental validation: second data set

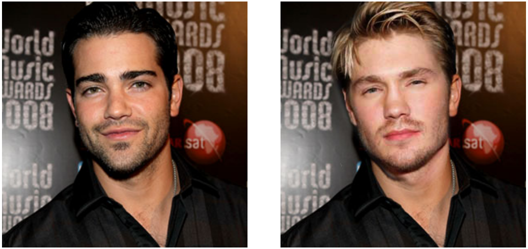
Example of textured images composing to second data set

- Starting from 50 natural images whose central region has textured / geometric content
  - creating 200 original and 200 fakes (50 per class)
  - according to previously defined classes

Experimental validation: third data set

- Rarely in real-world tampering is obtained by playing around only with JPEG on well defined square regions
- "Typical image user" will usually resort to several tools to:
  - cut&paste regions of irregular shape and variable size
  - correct inconsistencies of color, size and region edges
  - save partial/final result in JPEG format (often)
- Set of images of convincing visual quality by using several popular processing
  - starting from 30 original images of faces
  - creating 30 fakes by substituting the original faces

Experimental validation: third data set



Left: original image; right: tampered image obtained by pasting a new face

- Set of images of convincing visual quality by using several popular processing
  - starting from 30 original images of faces
  - creating 30 fakes by substituting the original faces

Experimental validation: settings I – general

- Mamdani's model for the if-then rules
  - THEN y_1 is B_1 AND…AND y_n is B_n AND…AND y_m is B_m
  - $n = 10$ inputs: $D_{A,B,C,D,E}$, $R_{A,B,C,D,E}$
  - $m = 1$ output: *tampering*
- We implemented the AND operator by means of `min` function
- We aggregated if-then rules by means of `max` function
- We performed defuzzification by means of the `centroid` method

Experimental validation: settings II – detection

- D normalized in $[0, 1]$
  - t_A: probabilistic SVM [18]; t_B, t_E: median of the probability map; t_C KS statistic; t_D: proposed statistic normalized $[0, 1]$
- Final value of D comes from two separate analysis:
  - on the region itself: D_{inner}; on the rest of the image: D_{outer}

$$D = |D_{outer} - D_{inner}|$$

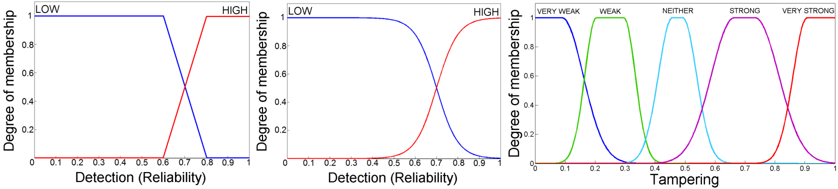
- Achieving more robustness to false positives
  - no tampering? difference should be small (ideally 0)
  - tampering? difference should be large enough (ideally 1)

Experimental validation: settings III – reliability

- R_A, R_D, R_E depend on last JPEG QF_2 (higher $=$ better)
- R increases linearly with QF_2, coefficients from the accuracy curves of articles $+$ interpolation
  - R_A from 0.73 when $QF_2 = 60$ to 0.96 when $QF_2 = 100$
  - R_D from 0.65 when $QF_2 = 60$ to 1.0 when $QF_2 = 100$
  - R_E from 0.659 when $QF_2 = 60$ to 0.91 when $QF_2 = 100$
- R_B and R_C do not seem to be affected
  - $R_B = 0.4$
  - $R_C = 0.85$
  - from tests on separated data sets

Experimental validation: settings IV – MFs

- Membership functions
  - input: `low` and `high`
  - output: `very weak`, `weak`, `neither`, `strong`, `very strong`
- Tests conducted with both piecewise and smooth MFs



Smooth MFs for system variables: (left)–(center) input detection depending on variable point p *of max fuzziness (e.g.* $p = 0.7$*). Input reliability uses MFs with the same shape but with fixed* $p = 0.5$*; (right) output.*

Experimental validation: evaluation procedure

- Comparison of the proposed approach against logic OR

Experimental validation: evaluation procedure

- Comparison of the proposed approach against logic OR
- First we evaluated separately each tool (ROC) on a dedicated data set
    - only on tampering classes satisfying each tool's assumptions

Experimental validation: evaluation procedure

- Comparison of the proposed approach against logic OR
- First we evaluated separately each tool (ROC) on a dedicated data set
  - only on tampering classes satisfying each tool's assumptions
- Then we aggregated the 5 curves by

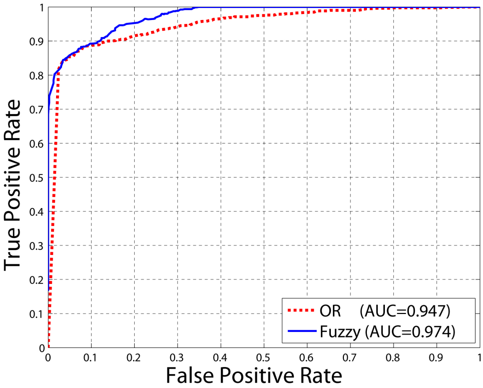
Experimental validation: evaluation procedure

- Comparison of the proposed approach against logic OR
- First we evaluated separately each tool (ROC) on a dedicated data set
  - only on tampering classes satisfying each tool's assumptions
- Then we aggregated the 5 curves by
  - sampling P_{fa} with $step = 0.01$
    - ▷ at each step 5 thresholds giving that P_{fa} for all the algorithms

Experimental validation: evaluation procedure

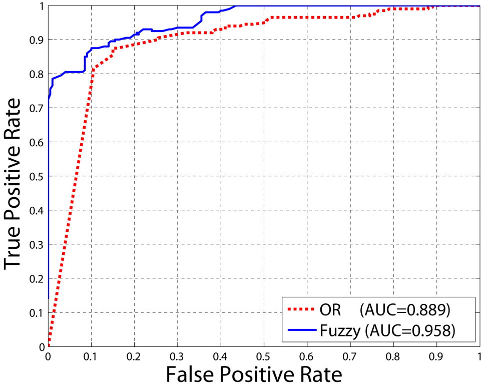
- Comparison of the proposed approach against logic OR
- First we evaluated separately each tool (ROC) on a dedicated data set
  - only on tampering classes satisfying each tool's assumptions
- Then we aggregated the 5 curves by
  - sampling P_{fa} with $step = 0.01$
    - ▷ at each step 5 thresholds giving that P_{fa} for all the algorithms
  - organizing them in 5-uples and using them as
    - ▷ binary thresholds to build the ROC of logical OR
    - ▷ points of maximum fuzziness to build the ROC of fuzzy methods

Experimental validation: Fuzzy vs OR – data set 1



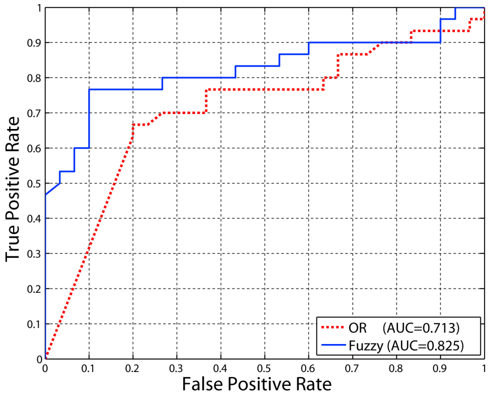
- Performance of piecewise fuzzy omitted, basically the same of smooth
- Fuzzy outperforms logic OR although not dramatically (+3.2% AUC)
  - classes designed so that at least one tool can detect the tampering
  - no unknown processing has been introduced while tampering with

Experimental validation: Fuzzy vs OR – data set 2



- Fuzzy outperforms logic OR (+6.9% AUC)
  - one step closer to a realistic scenario
  - processing introduces doubtful cases
  - Fuzzy approach handles doubt more efficiently

Experimental validation: Fuzzy vs OR – data set 3



- Fuzzy clearly outperforms logic OR (+11.2% AUC)
- Large portion of gain in the leftmost part ($P_{fa} < 0.15$)
  - a realistic scenario
  - encouraging step towards a real-world scenario with totally unknown processing

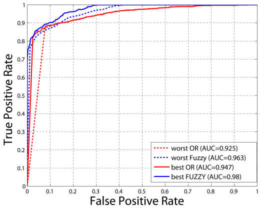
Experimental validation: robustness I

- R_A, R_C and R_E derived from the respective papers ☺
- R_B and R_C defined experimentally ☹
  - typical domain of system design
  - however, assignment may appear as an arbitrary choice depending on experimental data
- We demonstrate the robustness of the proposed approach with respect to relatively small variations of reliability
  - R_B in $[0.3, 0.5]$ and R_C in [0.7, 0.9] with $step = 0.05$
  - repeating experimental procedure
  - comparing best and worst ROC obtained

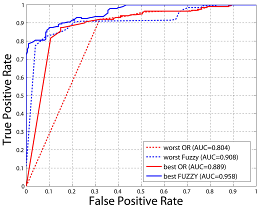
Experimental validation: robustness II

- Robustness with respect to variations of R_B and R_C. Solid lines best case, dotted lines worst case

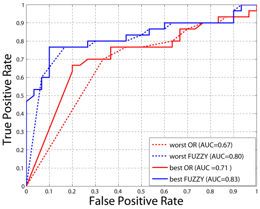
- Sensitivity to variations of reliability in the neighborhood of the assigned values is rather small ☺



1600 images



400 images



60 images

Experimental validation: computational burden

- In general, K forensic tools → 2^K possible interactions (K-uples) belonging either to T_{true}, T_{false} or T_{doubt}
- One compound rule per interaction → 2^K compound rules
- Each compound rule needs to be converted (thanks Matlab! ☹) → 2^{2K} final basic rules
- In our case: $K = 5$, $2^5 = 32$ cases, $2^{(2\times5)} = 1024$ rules
  - On a 3GHz dual-core processor, 4GB RAM, 32bit OS
    - 1 second to build T_{true}, T_{false} and T_{doubt} (once per data set)
    - 0.2 seconds to build the system (once per image)
    - 0.5 seconds to resolve all rules (once per image)

Conclusions and future work

- A framework based on Fuzzy Logic that is capable of
  - fusing the outputs of different forensic tools used in parallel
  - reducing the impact of uncertainty affecting the tools
- Highlights
  - application to a realistic image forensics scenario
  - outperforms classical methods of decision (e.g. *OR*)
- Several topics still need to be explored
  - integration of a wider set of forensic tools
  - accuracy on real-world tampered images
  - suspicious tampered region not known a priori

References I

E. Delp, N. Memon, and M. Wu.
Special issue on digital forensics.
IEEE Signal Processing Magazine, 26(2), 2009.

J.A. Redi, W. Taktak, and J.L. Dugelay.
Digital image forensics: a booklet for beginners.
Multimedia Tools and Applications, pages 1–30, 2011.

J. Lukáš and J. Fridrich.
Estimation of primary quantization matrix in double compressed JPEG images.
In *Proc. of Digital Forensic Research Workshop*, pages 5–8, Cleveland, Ohio, USA, August, 2003.

Hany Farid.
Exposing digital forgeries from JPEG ghosts.
IEEE Transactions on Information Forensics and Security, 4:154–160, 2009.

A.C. Popescu and H. Farid.
Exposing digital forgeries by detecting traces of resampling.
IEEE Transactions on Signal Processing, 53(2):758–767, 2005.

B. Mahdian and S. Saic.
Blind authentication using periodic properties of interpolation.
IEEE Transactions on Information Forensics and Security, 3(3):529–538, 2008.

References II

W. Luo, Z. Qu, J. Huang, and G. Qiu.
A novel method for detecting cropped and recompressed image block.
In *Proc. of International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages II–217 –II–220, april Honolulu, Hawaii, USA, April, 2007.

Z. C. Lin, J. F. He, X. Tang, and C. K. Tang.
Fast, automatic and fine-grained tampered JPEG image detection via DCT coefficient analysis.
Pattern Recognition, 42:2492–2501, 2009.

S. Bayram, H.T. Sencar, and N. Memon.
A survey of copy-move forgery detection techniques.
In *IEEE Western New York Image Processing Workshop*, Rochester, NY, USA, November, 2008.

Irene Amerini, Lamberto Ballan, Roberto Caldelli, Alberto Del Bimbo, and Giuseppe Serra.
A SIFT-based forensic method for copy-move attack detection and transformation recovery.
IEEE Transactions on Information Forensics and Security, in press(3):1099–1110, September.

L. A. Zadeh.
Fuzzy sets.
Information and Control, 8:338–353, 1965.

L. A. Zadeh.
Outline of a new approach to the analysis of complex systems and decision.
IEEE Transactions on Systems, Man, and Cybernetics, SMC-3:28–44, 1973.

References III

M. Kharrazi, H. T. Sencar, and N. Memon.
Improving steganalysis by fusion techniques: A case study with image steganography.
In *Transactions on Data Hiding and Multimedia Security*, pages 123–137, 2006.

G. Chetty and M. Singh.
Nonintrusive image tamper detection based on fuzzy fusion.
International Journal of Computer Science and Network Security, 10:86–90, 2010.

SN Sivanandam, S. Sumathi, and SN Deepa.
Introduction to fuzzy logic using MATLAB.
Springer Verlag, 2007.

T.Bianchi and A.Piva.
Detection of non-aligned double JPEG compression with estimation of primary compression parameters.
In *Proc. of IEEE International Conference on Image Processing (ICIP)*, pages 1969–1972, Sept. Brussels, Belgium, September 2011.

T. Bianchi, A. De Rosa, and A. Piva.
Improved DCT coefficient analysis for forgery localization in JPEG images.
In *Proc. of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 2444–2447, May Prague, Czech Republic, May 2011.

John C. Platt.
Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods.
In *Advances in large margin classifiers*, pages 61–74, 1999.