

**POLITECNICO DI MILANO**  
Corso di Laurea Magistrale in Ingegneria Informatica  
Dipartimento di Elettronica, Informazione e Bioingegneria



**ALGORITMO DI TAMPERING  
DETECTION OTTIMIZZATO  
TRAMITE SEGMENTAZIONE DELLA  
SCENA INQUADRATA**

**Relatore: Prof. Giacomo BORACCHI**  
**Correlatore: Ing. Claudio MARCHISIO**

**Tesi di Laurea di:**  
**Adriano GAIBOTTI, matricola 780200**

**Anno Accademico 2013-2014**



*A Sara*



# Sommario

Uno dei principali problemi, quando si ha a che fare con applicazioni di monitoraggio video, è quello di identificare quegli eventi che possono compromettere la corretta ripresa della scena da parte del sensore. Può capitare, ad esempio, che dell'acqua piovana si depositi sulla lente della camera, rendendo l'immagine acquisita sfocata, oppure che qualcuno sposti la camera in modo che essa non riprenda più la scena che stava monitorando. Il problema di individuare, in maniera automatica, questo tipo di eventi prende il nome di *tampering detection*. Nella letteratura scientifica lo studio di questo problema si è concentrato solamente sulle applicazioni di *videosorveglianza*, dove è necessario che la camera operi con una frequenza di acquisizione elevata. In questo contesto immagini acquisite in istanti di tempo consecutivi hanno un alto grado di correlazione tra loro, in quanto il contenuto visivo varia molto poco.

Lo scopo della tesi è lo sviluppo di un algoritmo di *tampering detection* adatto a operare con frequenze di acquisizioni basse, ad esempio un'immagine ogni minuto. In questi casi, se consideriamo, ad esempio, la ripresa di una strada, in cui passano delle macchine o dei pedoni, abbiamo un'elevata dinamicità che non permette di fare un confronto tra frame consecutivi per identificare gli eventi di nostro interesse. In aggiunta, abbiamo dei cambiamenti di luminosità, tra un'immagine e la successiva, più sostanziali rispetto al caso di acquisizione continua. La nostra proposta è quella di monitorare nel tempo degli indicatori semplici, calcolati considerando solamente il *contenuto visivo* delle singole immagini, dove una *variazione sostanziale* è associata a un evento di *tampering*. Data l'alta variabilità di questi indicatori abbiamo introdotto una *segmentazione* della scena ripresa, estratta durante una fase di *configurazione* dell'algoritmo, in modo da considerare solo le regioni in cui il monitoraggio risulta più efficace. La tesi è stata svolta durante uno stage presso *ST Microelectronics*, particolarmente interessata a sviluppare algoritmi intelligenti di processamento immagini da integrare nei propri dispositivi hardware, e a scenari di impiego per questi. Prove sperimentali hanno confermato l'efficacia di utilizzare la segmentazione rispetto a considerare l'intera scena per individuare eventi di spostamento della camera.



# Ringraziamenti

Ringrazio .....





# Indice

<b>Sommario</b>	<b>iii</b>
<b>Ringraziamenti</b>	<b>v</b>
<b>1 Introduzione</b>	<b>3</b>
<b>2 Stato dell'arte</b>	<b>5</b>
2.1 Tampering Detection . . . . .	5
2.1.1 Identificazione di occlusioni . . . . .	5
2.1.2 Identificazione di spostamenti della camera . . . . .	5
2.1.3 Identificazione di sfocature . . . . .	5
<b>3 Impostazione del problema di ricerca</b>	<b>7</b>
3.1 Modello delle osservazioni . . . . .	7
3.1.1 Sfocatura . . . . .	7
3.1.2 Spostamento della camera . . . . .	9
3.1.3 Occlusione . . . . .	10
3.2 Tampering detection . . . . .	10
<b>4 Soluzione proposta</b>	<b>11</b>
4.1 Estrazione dei descrittori del cambiamento . . . . .	11
4.2 Algoritmo di segmentazione . . . . .	11
4.3 Monitoraggio one-shot . . . . .	11
4.4 Monitoraggio sequenziale . . . . .	11
<b>5 Realizzazioni sperimentali e valutazione</b>	<b>13</b>
5.1 Acquisizione dei dataset . . . . .	13
5.2 Risultati . . . . .	13
<b>6 Direzioni future di ricerca e conclusioni</b>	<b>15</b>
<b>Bibliografia</b>	<b>17</b>



# Elenco delle figure

3.1	Esempio di sfocatura . . . . .	8
3.2	Sequenza di otto frame consecutivi acquisiti ogni minuto . . .	9
3.3	Esempio di spostamento della camera . . . . .	9



# Elenco delle tabelle









# Capitolo 1

## Introduzione

Negli ultimi anni le applicazioni di tipo multimediale sono aumentate in maniera esponenziale, soprattutto per quanto riguarda i contenuti video. L'abbassamento dei prezzi e delle dimensioni dei *sensori* e delle componenti hardware



## Capitolo 2

# Stato dell'arte

In questo capitolo elenchiamo quelle che sono le principali tecniche, presenti nella letteratura scientifica, utilizzate per identificare tentativi di manomissione su camere di videosorveglianza.

### 2.1 Tampering Detection

Nei moderni sistemi di videosorveglianza troviamo spesso algoritmi utilizzati per identificare particolari eventi all'interno della scena ripresa dalla camera. Ad esempio è possibile avere un software in grado di identificare le targhe delle automobili che superano il limite di velocità , oppure la presenza di oggetti incustoditi in una stazione [1]. Affinché questi algoritmi funzionino correttamente, è importante che le immagini, che verranno poi processati da questi sistemi, mantengano una certa qualità.

#### 2.1.1 Identificazione di occlusioni

#### 2.1.2 Identificazione di spostamenti della camera

#### 2.1.3 Identificazione di sfocature



## Capitolo 3

# Impostazione del problema di ricerca

In questo capitolo andiamo a descrivere, in maniera formale e rigorosa, l'ambiente e le problematiche che l'algoritmo di tampering detection andrà ad affrontare. Il primo paragrafo illustra quali sono gli eventi che siamo interessati a identificare, mentre il secondo paragrafo formalizza il concetto di tampering detection.

### 3.1 Modello delle osservazioni

Il nostro campo di osservazione si concentra su quegli eventi che si interpongono tra la scena ripresa da una camera e il sensore che deve acquisire le immagini. Non vogliamo, cioè, identificare degli eventi particolari che avvengono nella scena, come un oggetto lasciato incustodito, bensì vogliamo identificare quegli eventi tali per cui il sensore non è più nelle condizioni di riprendere, in maniera ottimale, la scena. Nel seguito cerchiamo di dare una definizione formale di questi eventi.

#### 3.1.1 Sfocatura

Il fenomeno della sfocatura avviene quando un elemento trasparente o semi-trasparente si interpone tra la lente della camera e la scena ripresa, causando una perdita nei dettagli della scena ripresa.

La figura 3.1 mostra un esempio di sfocatura applicata a una scena reale. Essa può avvenire per i più svariati motivi:

- per *cause naturali*, come ad esempio dell'acqua piovana che si deposita sulla lente, oppure per la condensa dovuta all'umidità e alle basse temperature, oppure un raggio di sole incidente sull'obiettivo della camera;



Figura 3.1: Esempio di sfocatura

- per *intervento dell'uomo*, che a sua volta può avvenire in maniera intenzionale (e in questo caso si può parlare di *manomissione*) oppure non intenzionale. Ad esempio, si può direttamente intervenire sulla messa a fuoco, nel caso sia possibile cambiarla manualmente; oppure (come nel caso della figura 3.1) è possibile applicare una sostanza semitrasparente sulla lente della camera, come il gas di un deodorante spray.

Riprendendo [2], questo fenomeno può essere modellato come un operatore di *degradazione*  $D$  applicato a un'immagine  $y$ , considerata priva di errori, i.e.,

$$z = D[y]. \quad (3.1)$$

In particolare, all'interno dell'operatore  $D$  si può considerare il contributo dovuto a un operatore di *sfocatura*  $B$  (dall'inglese *blur*) e un termine  $\eta$  corrispondente al rumore, i.e.,

$$z(x) = D[y](x) = B[y](x) + \eta(x), \quad x \in X \quad (3.2)$$

dove abbiamo indicato con  $x$  le coordinate dei *pixel* dell'immagine. Possiamo assumere la sfocatura  $B$  come un operatore *lineare* di *convoluzione*,

$$B[y](x) = \int_X y(s)h(x, s)ds, \quad (3.3)$$

dove  $h(x, s)$  rappresenta un filtro *gaussiano* o *uniforme*, il cui risultato consiste nel rendere le differenze di intensità, tra pixel adiacenti, più morbide (*smooth*).

Nel caso più generale possiamo considerare che la camera acquisisca un sequenza di  $N$  osservazioni  $\{z_i\}, i = 1, \dots, N$ , quindi la formula 3.2 si può riscrivere come

$$z_i(x) = D[y](x) = B_i[y_i](x) + \eta(x), \quad x \in X. \quad (3.4)$$



Figura 3.2: Sequenza di otto frame consecutivi acquisiti ogni minuto



Figura 3.3: Esempio di spostamento della camera

La sequenza delle immagini  $\{y_i\}, i = 1, \dots, N$ , può variare in maniera significativa nel suo contenuto, in base alla scena ripresa. Un esempio è illustrato nella figura 3.2, in cui le immagini riprese dalla camera sono acquisite ogni minuto. Nonostante la scena ripresa sia la stessa, il suo contenuto varia parecchio. Il continuo passaggio di automobili rende ciascuna immagine diversa dalle altre e, di conseguenza, usare tecniche di identificazione di cambiamenti basate sul confronto di frame consecutivi porta facilmente a un'alta generazione di falsi positivi, dato che diventa difficile distinguere il caso in cui è cambiato il contenuto delle immagini ( $y_i \neq y_{i+1}$ ) da una sfocatura ( $B_i \neq B_{i+1}$ ).

### 3.1.2 Spostamento della camera

Lo spostamento della camera avviene quando cambia la scena ripresa. Le cause possono essere, ancora una volta, di tipo naturale, ad esempio una raffica di vento che sposta la camera, oppure un intervento malevolo da parte di qualcuno. La figura 3.3 mostra un esempio di spostamento della camera tratto da un caso reale. Possiamo formalizzare il concetto di spostamento della camera nel modo seguente: consideriamo la sequenza  $\{y_i\}$  di immagini generate da una camera in una certa posizione, e la sequenza  $\{w_i\}$  di imma-

gini generate dalla stessa camera in una posizione differente.

Possiamo, dunque, considerare la sequenza di immagini  $\{z_i\}$  in cui avviene uno spostamento della camera all'istante  $T^*$  nel seguente modo:

$$z_i(x) = \begin{cases} y_i(x) + \eta(x) & \text{per } i < T^* \\ w_i(x) + \eta(x) & \text{per } i \geq T^*, \end{cases} \quad (3.5)$$

dove  $\eta(x)$  è un rumore stazionario.

Anche per lo spostamento della camera vale la considerazione fatta nel caso della sfocatura: il contenuto delle immagini varia con il passare del tempo, quindi identificare lo spostamento confrontando frame consecutivi nel tempo genera un alto numero di falsi positivi.

### **3.1.3 Occlusione**

## **3.2 Tampering detection**



## Capitolo 4

# Soluzione proposta

4.1 Estrazione dei descrittori del cambiamento

4.2 Algoritmo di segmentazione

4.3 Monitoraggio one-shot

4.4 Monitoraggio sequenziale



## Capitolo 5

# Realizzazioni sperimentali e valutazione

### 5.1 Acquisizione dei dataset

### 5.2 Risultati



## Capitolo 6

# Direzioni future di ricerca e conclusioni



# Bibliografia

- [1] <http://www.mitan.it/security-solution/videosorveglianza/sistemi-di-videosorveglianza-e-registrazione/>. Visitato il giorno 09/03/2015.
- [2] Cesare Alippi, Giacomo Boracchi, Romolo Camplani, and Manuel Roveri. Detecting external disturbances on the camera lens in wireless multimedia sensor networks. *Instrumentation and Measurement, IEEE Transactions on*, 59(11):2982–2990, 2010.