Image Forensics (Parte 1)

Forensic Science:

è l'utilizzo di tecniche e metodi attraverso la preservazione, la collezione, validazione identificazione, analisi, interpretazione , documentazione e presentazione di prove digitali derivate da risorse digitali con l'obbiettivo di ricostruire eventi criminosi o aiutando a individuare azioni non autorizzate.

Tipi di Image Forensics:

- Digital forensics:
  - computer forensics
  - network forensics
  - mobile forensics
  - *multimedia forensics*
- Analog forensics

Per quanto guarda **Multimedia Forensics**:

Le tracce vengono fatte a partire dall'acquisizione del media . E l'Image Forensics Cerca proprio di acquisire queste tracce. Le impronte non sono *artefatti* da eliminare, bensì, vengono considerati ***assets***.

Questi servono per ricostruire la catena del processo applicato all'oggetto digitale.

Le tecniche di *Image Forensics* servono per rispondere a queste domande:

- *Qual'è l'origine di questa immagine?* -> ***Source Identification***
  - Risponde a domande del tipo: che dispositivo ha scattato questa foto, che modello? Di che marca?
- *L'immagine ha avuto qualche modifica?* ->***Integrity Verification / Tampering Detection***.
  - Delle volte la semantica delle foto può essere cambiato attraverso il cambiamento dell'istogramma dei colori , cancellando degli oggetti, aggiungendo elementi (Image Splicing)

Queste tracce vengono da tutti i passaggi del ciclo di una foto

- un sistema ottico concentra l'energia riflessa da un oggetto.
- un sensore misura l'energia riflessa
- *in-camera processing* che lo converte in una immagine digitale.
  Ogni passaggi contiene delle imperfezioni che lascia quindi un impronta della pagina.

Noi studieremo gli *artefatti* legati ai processi e a come è fatta una fotocamera , in particolare:

- Chromatic Aberrations -> introdotte dalle lenti.
- Color Filter Array Interpolation -> introdotti dai Sensori
- Rumore -> introdotto dal sensore

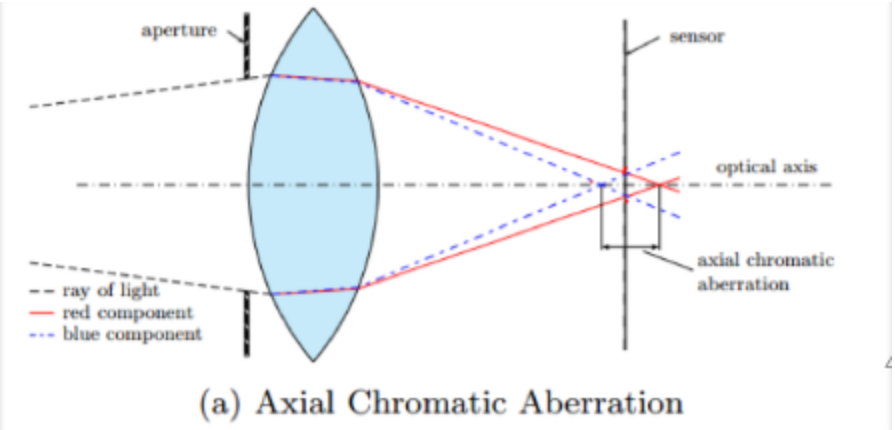
Chromatic Aberrations:

La luce passa attraverso una lente che è focalizzata sul sensore producendo un effetto replica, però le lenti fanno si che la luce subisca una distorsione e ce ne sono di tanti tipi, tra cui questa di cui stiamo parlando.

Questa distorsione è data ***dall'inabilità delle lenti di focalizzare tutte le componenti dello spettro nello stesso punto del sensore*** , data dal fatto che quando un raggio di luce attraversa la lente si ha una rifrazione che dipende dalla differente lunghezza d'onda.

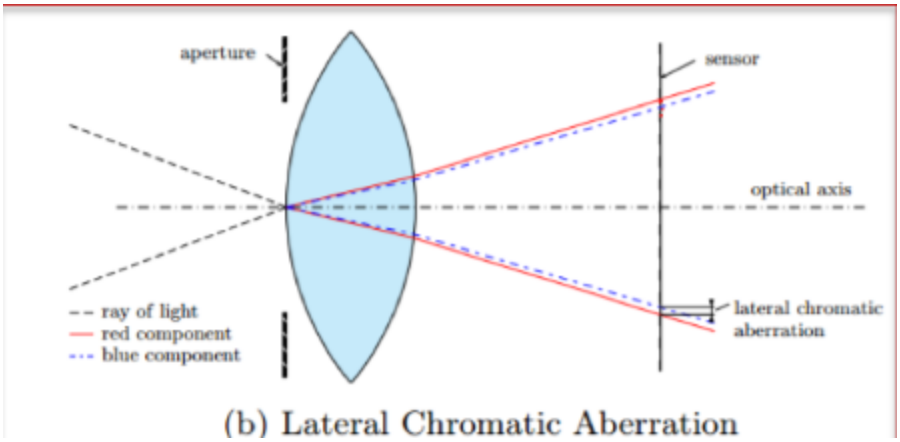
Si ha:

- Axial Chromatic Aberration: è la variazione longitudinale del punto focale tra differenti lunghezze d'onda lungo l'asse ottico.



(a) Axial Chromatic Aberration

- Lateral Chromatic Aberration(***LCA***): si manifesta come uno spostamento nel punto dove la luce raggiunge il sensore con diverse lunghezze d'onda.

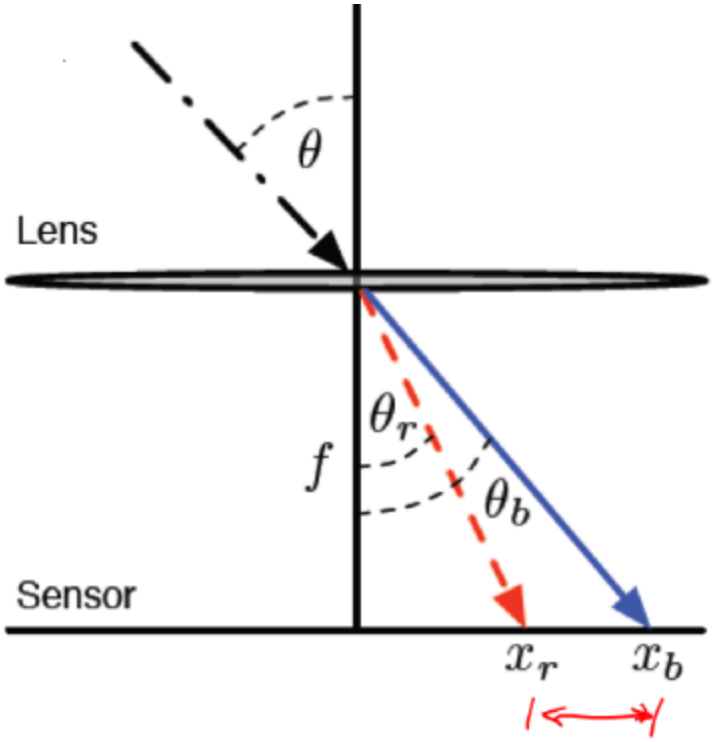


(b) Lateral Chromatic Aberration

(si manifestano come frange di colore sui confini delle foto)
Attraverso la Legge di Snell la luce viene riflessa attraverso questa formula:

$$n_1 \cdot sin(\theta_1) = n_2 \cdot sin(\theta_2)$$

dove θ_1 è l'angolo di incidenza, θ_2 è l'angolo di rifrazione e n_1 e n_2 sono gli indici di rifrazione della media della luce che passa dalla lente.
L'immagine sotto mostra che la luce con lunghezza d'onda corta (raggio blu) e con lunghezza d'onda lunga colpiscono il sensore in posizioni diverse dove $n_r = n_f(\lambda_r)$ e $n_b = n_f(\lambda_b)$.



Quindi:

$$x_r \approx \alpha \cdot x_b$$

Dove: $\alpha = n_b/n_r = n_f(\lambda_b)/n_f(\lambda_r) = sin(\theta_b)/sin(\theta_a)$
Con estensione 2D:
$(x_r, y_r) = \alpha \cdot (x_b, y_b)$

Parlando ancora ancora di questa aberrazione , **LCA** si ha in ogni punto del sensore e dipende dalla distanza ottica tra sensore e lente. **LCA** è grande nelle regioni di confine di una foto. Il vettore che rende visibile questa foto dato da: $v = (x_r - x_b, y_r - y_b)$.
Il modello consiste in tre parametri

Come si usa LCA?
LCA si stima in due modi:

- aberation tra i canali R e G.
- aberation tra i canali B e G
  Il risultato dovrebbe essere un deallineamento tra i due canali, è possibile quindi cercare i parametri di questo modello per portare i canali ad **essere allineati**, e si può fare attraverso la massimizzazione o minimizzazione di una certa metrica. Si fa infatti una ricerca iterativa brute force che viene fatta un certo numero di volte ricercando il minimo globale.
  Immagine balistica: Data una fotocamera C e un immagine I, viene confrontata con un insieme di immagini prese dalla fotocamera C l'immagine I.
  Diversi dispositivi dello stesso modello di fotocamera usano lenti con proprietà simili, quindi assumiamo che LCA è dipendente dal camera-model .
  Tampering Detection: le manipolazioni delle foto possono portare ad inconsistenze nel Chromatic Aberration, quindi nel momento in cui si ha una cosa del genere questa è un evidenza di un eventuale manomissione della fotografia. Infatti data un immagine I, la c.a viene stimata da l'intera immagine che quindi è una stima globale che viene quindi confrontata con piccoli blocchi.
  Source Identification: la differenza tra un immagine generata dal computer(IA) e le immagini naturali è che le seconde presentano aberrazioni rispetto alle prime.

I limiti del LCA:
Questi metodi funzionano bene su parti delle immagini non compresse, per invece immagini compresse molto con Jpeg ci si aspettano risultati non buoni. Inoltre si riesce a riconoscere il modello della camera in insiemi piccoli.

Color Filter Array (CFA):

• Una fotocamera cattura solo una parte delle informazioni sui colori per ogni pixel. Poichè non si utilizza il **3CMOS**/**3CCD** che per ogni pixel si hanno *tre fotositi* (nelle fotocamere professionali/ di alta gamma).

• I **colori mancanti** vengono ricostruiti attraverso un processo chiamato **interpolazione**.

• L’interpolazione introduce una **correlazione periodica** tra i pixel, creando pattern specifici nei canali di colore.

• Esistono diversi tipi di CFA (es. **Bayer**, **Diagonal**, **Striped**, ecc.), ciascuno con uno schema specifico di disposizione dei pixel.

• L’analisi di questi pattern può essere utilizzata per **identificare la marca e il modello della fotocamera**.

• Gli **artefatti** sono imperfezioni visive introdotte dal processo di interpolazione dei colori nei pixel.

• Si manifestano come **pattern ripetitivi**, **aloni attorno ai bordi** o **strutture periodiche innaturali**

• Sono particolarmente evidenti nelle aree di forte contrasto.

• Questi artefatti possono essere usati per **rilevare manipolazioni** o **analizzare l’autenticità** di un’immagine.

Si utilizza il **CFA** per:

- **Image Ballistic**: data una fotocamera C e un immagine, possiamo caratterizzare un certo insieme di immagini prese con la fotocamera C e l'immagine I accordandosi con tipo di filtro e di algoritmo di interpolazione. Se sono simili allora I è fatta da C.
- **Tampering Detection**: un immagine che viene da una fotocamera digitale in assenza di processo mostrerà un artefatto dato dalla demosaicizzazione ogni tot gruppo di pixel. Questo indicherà quindi un certo tipo di CFA.
  Quindi dato un immagine dove non si ha una correlazione di questo genere fa venire dubbi sull'autenticità di questa immagine.

Sensor Noise:

I pixel sono fatti di silicio e sono creati affinchè quando la luce batte sul fotosito essa converta la luce in elettroni usando un effetto fotoelettrico, la carica del sensore è trasferita fuori da esso ad un A/D Converter (Analog to Digital) e poi processato e salvato come formato immagine. Però questi sensori possono avere dei difetti visto che si parla di processi elettronici, i difetti maggiori dei sensori sono:

- **Difetti nei pixels**: tra questi difetti vengono inclusi: i pixel morti, i pixel traps, hotpoint defects, difetti di colonna e difetti di nuvola(cluster). Questi difetti possono far si che

l'immagine cambi(data la interpolazione).

- **Fixed Pattern Noise** (FPN): rumore creato quando i pixel sono esposti all'oscurità, è un rumore additivo dato dalla temperatura e il tempo di esposizione
- **Photo Response Non Uniformity** (PRNU): il numero di elettroni generati dalla luce incidente su ogni pixel dipende dalla **dimensione** della area fotosensibile del pixel e dalla **omogenità del silicio**. Il risultato, quindi, è che lo stesso quantitativo di fotoni generano una diversa quantità di elettroni e questo vuol dire un output digitale diverso.
  La variazione dell'efficienza degli elettroni di tutti i pixel può essere registrata da una matrice **K** della stessa dimensione del sensore. Quando il sensore è illuminato con un intensità **I** (senza imperfezioni), il sensore registrerà anche la scena con rumore $I + I \cdot K$.
  Quindi $I \cdot K$ è riferito al PRNU.
  IMPORTANTE: l'energia del PRNU dipende dalla intensità della luce ma il pattern del rumore dovrebbe essere sempre lo stesso.

Le proprietà del PRNU rappresentate da K sono le seguenti:

1. *Universalità* : ogni dispositivo esibisce un PRNU.
2. *Generalità* : la componente PRNU si ha in ogni foto indipendentemente dalle opzioni di fotocamera, ottiche o contenuto della scena.
3. *Stabilità*: il fattore **K** è stabile nel tempo e anche sotto delle condizioni ambientali come l'umidità la temperatura ecc...
4. *Robustezza*: la componente PRNU $I \cdot K$ rimane anche dopo l'elaborazione dell'immagine
5. *Unicità*: ogni sensore esibirà un PRNU differente

La stima del FingerPrint

L'impronta è K e si può stimare attraverso le tecniche standard per la stima dei parametri e sapendo quello si riesce a riconoscere il modello del sensore.
Dato un sensore MxN allora:

- $Y[i, j]$ è l'intensità della luce incidente sul pixel *(i,j)*.
- $I[i, j]$ è il segnale quantizzato registrato per il pixel *(i,j)*.
  Si ha:

$$I = g^{\gamma} \cdot [(1 + K) \cdot Y + \Omega]^{\gamma} + Q$$

  dove:

- g è il fattore di guadagno(differente a seconda del canale del colore), esso aggiusta l'intensità del pixelguardando la sensibilità del pixel per ottenere il corretto bilancio del bianco.

- γ è il fattore di correzione gamma.
- K è il rumore a media nulla responsabile del PRNU.
- Ω è la combinazione degli altre sorgenti di rumori (come rumore oscurità ecc...).
- Q è la distorsione dovuto alla quantizzazione e alla compressione JPEG

$$I = (g \cdot Y)^{\gamma} \cdot [1 + K + \Omega/Y]^{\gamma} + Q$$

ma $(1+x)^{\gamma} = 1 + \gamma \cdot x + O(x)$ quindi:

$$(g \cdot Y)^{\gamma} \cdot (1 + K \cdot \gamma + \gamma \cdot \Omega/Y) + Q$$

con $(g \cdot Y)^{\gamma} = I^{(0)}$ viene:

$$I^{(0)} + I^{(0)} \cdot \gamma \cdot K + \Theta = I^{(0)} + I^{(0)} \cdot K + \Theta$$

dove:

- $\Theta = I^{(0)} \cdot \gamma \cdot \Omega/Y$
- la componente γ viene assorbita da K per semplicità nella notazione
  Per trovare l'**impronta** si può sottrarre a I l'immagine senza rumore $I^{(0)}$, unico problema è che noi non sappiamo com'è l'immagine senza rumore.
  Possiamo quindi ottenere $I^{(0)}$ rimuovendo il rumore tramite un filtro denoising F:

  $$\hat{I}^{(0)} = F(I)$$

  Quindi si ha che il rumore $W = I - \hat{I}^{(0)}$:

  $$W = I - \hat{I}^{(0)} = (I^{(0)} + I^{(0)} \cdot K + \Theta) - \hat{I}^{(0)} = (I^{(0)} + I^{(0)} \cdot K + \Theta) - \hat{I}^{(0)} + IK - IK$$

  $$W = I^{(0)} - \hat{I}^{(0)} + K \cdot (I^{(0)} - I) + IK + \Theta = IK + \Xi$$

  dove Ξ è la somma di Θ e dei termini introdotti dal filtraggio.

Stima dell'Impronta:

1. Introduzione di Massima Verosimiglianza (Maximum Likelihood Estimation - MLE)

📝 **Cosa significa “Likelihood Function”?**

La **funzione di verosimiglianza** () è uno strumento statistico che ci permette di stimare un **parametro sconosciuto** () basandoci su un insieme di **osservazioni** ().

• Normalmente, una distribuzione di probabilità ci dice: *“Dato un parametro , qual è la probabilità di osservare ?”*

• La funzione di verosimiglianza invece fa l'opposto: *"Dato un insieme di osservazioni , qual è il valore di che rende queste osservazioni più probabili?"*

📊 Obiettivo della MLE

L'obiettivo è **massimizzare la funzione di verosimiglianza** rispetto al parametro .

Se la funzione è differenziabile, il massimo si ottiene risolvendo:

🔑 Log-Likelihood (Logaritmo della verosimiglianza)

Lavorare direttamente con la funzione di verosimiglianza può essere complicato perché coinvolge **prodotti di probabilità**.

Per semplificare, si usa il **logaritmo naturale** della funzione di verosimiglianza:

Questo trasforma i **prodotti in somme**, rendendo i calcoli più semplici.

2. Stima dell'Impronta del Sensore (Fingerprint Estimation)

📝 Cosa stiamo cercando di fare?

Ogni fotocamera ha un'impronta digitale unica chiamata **PRNU (Photo Response Non-Uniformity)**, che è causata da piccole variazioni nei pixel del sensore.

• **Dati**: Abbiamo un insieme di **immagini (**$I_1, I_2, \ldots, I_d$) scattate dalla stessa fotocamera.

• **Obiettivo**: Estrarre il **rumore di riferimento** che è presente in tutte le immagini.

• Ogni immagine contiene un **residuo di rumore** .

🛠️ Come si procede?

1. **Estrazione del residuo di rumore (**W_i**)** da ciascuna immagine.
2. **Media dei residui di rumore** per:

   • Rimuovere il rumore casuale.

   • Isolare il rumore sistematico (PRNU).

   • Rimuovere i dettagli della scena.

In pratica, stiamo cercando un **modello comune di rumore** che rappresenta la firma unica della fotocamera.
Noi sappiamo che: $W_i = KI_i + \Xi_i$, dove Ξ è modellato come un rumore bianco a media nulla e

varianza σ^2 e indipendente da KI_i. Se la ricomponiamo come $\frac{W_i}{I_i} = K + \frac{\Xi_i}{I_i}$ allora questa segue una distribuzione gaussiana con media K e varianza $(\sigma/I_i)^2$.

3. Stima del Rumore del Sensore (Sensor Noise Estimation)

📝 Equazione del Rumore

Il rumore osservato può essere rappresentato come:

$$\frac{W_i}{I_i} = K + \frac{\Xi_i}{I_i}$$

• W_i: Rumore estratto dall'immagine .

• I_i: Intensità dei pixel dell'immagine.

• K: Rumore di riferimento (PRNU).

• Ξ_i: Rumore casuale aggiuntivo.

📊 Funzione di Verosimiglianza per il Rumore

Per stimare il valore ottimale di , si costruisce una funzione di verosimiglianza.

$$L(K) = \prod_{k=1}^{d} \frac{1}{2\pi(\sigma_f/I_k)^2} e^{-\frac{(W_k/I_k - K)^2}{2(\sigma_f/I_k)^2}}$$

• La formula tiene conto delle distribuzioni di probabilità dei residui di rumore osservati.

📈 Log-Likelihood e Massimizzazione

Per trovare il valore ottimale di , si usa la **log-verosimiglianza**:

$$\ln L(K) = -\frac{d}{2}\ln(2\pi(\sigma_f/I_k)^2) - \sum_{k=1}^{d} \frac{(W_k/I_k - K)^2}{2(\sigma_f/I_k)^2}$$

🛠️ Stima Finale di

Derivando la funzione rispetto a e ponendo a zero:

$$\frac{\partial log(L(K))}{\partial K} = \sum_{k=1}^{d} \frac{W_k/I_k - K}{\sigma^2/(I_k)^2} = 0$$

di conseguenza:

$$\hat{K} = \frac{\sum_{k=1}^{d} \frac{W_k}{I_k}}{\sum_{k=1}^{d} \frac{\sigma^2}{(I_k)^2}}$$

Questa è la stima finale del rumore sistematico (PRNU).

4. Stima Finale del PRNU

• Si ottiene il **PRNU stimato (**)** combinando i residui di rumore di più immagini:

$$\hat{K} = \frac{\sum_{k=1}^{d} W_k I_k}{\sum_{k=1}^{d} (I_k)^2}$$

📊 Interpretazione del Risultato

• Questo valore rappresenta una **“firma digitale”** unica per il sensore della fotocamera.

• Confrontando questa firma con altre immagini, possiamo **identificare se un’immagine proviene da una specifica fotocamera**.

Identificazione della Fotocamera a partire dall'Immagine:

Data un immagine I_Q e l'impronta K, allora il rumore della nostra immagine W_Q conterrà l'impronta K?
Si fanno due ipotesi:
-H_0 (ipotesi di non corrispondenza con K): $W_Q = \Xi$
-H_1(ipotesi di corrispondenza con K): $W_Q = I_Q K + \Xi$
Si utilizza per riuscire a identificare una delle due ipotesi l' NCC (*normalized cross-correlation*):

$$\rho(I_Q K, W_Q)$$

dove questa corrisponde ad un operazione di questo genere dove I, J sono della dimensione MxN:

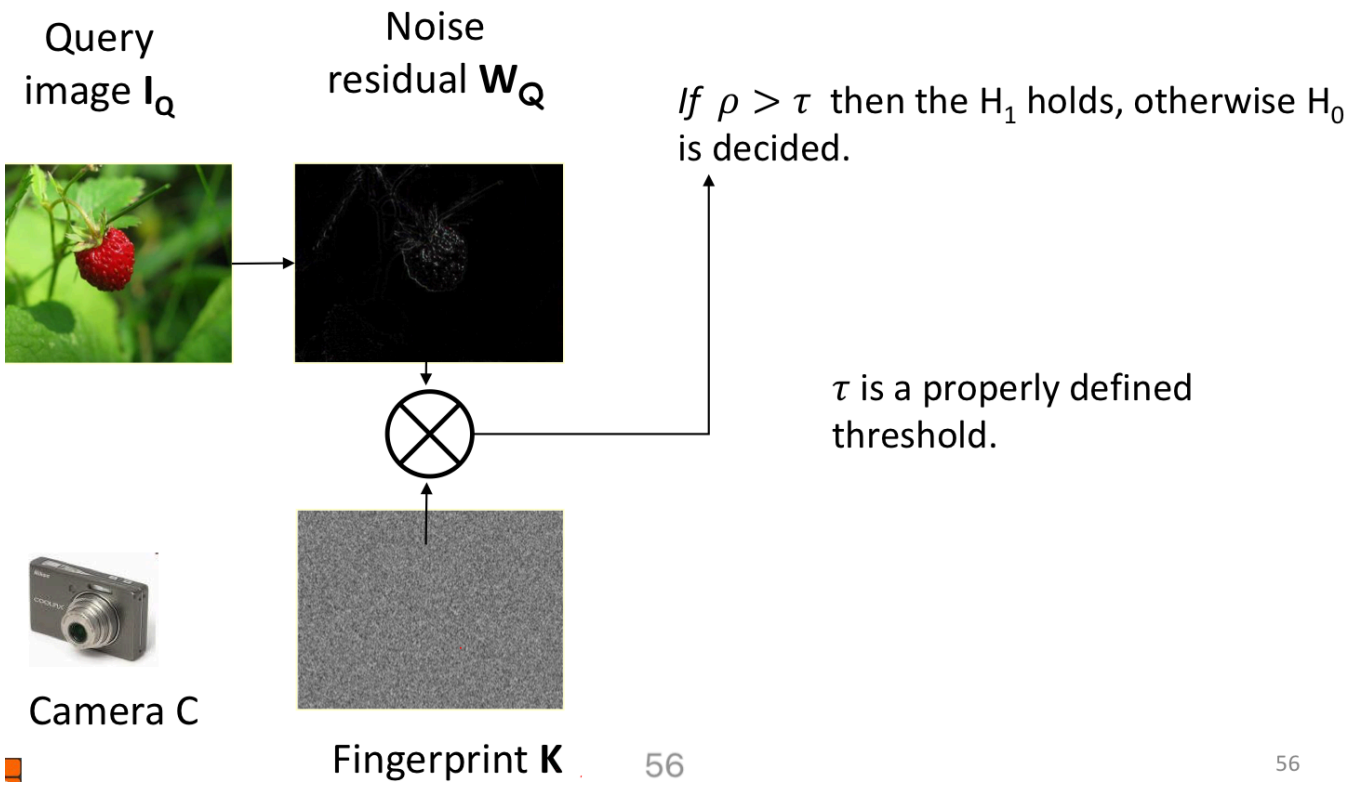
$$\rho(\mathbf{I}, \mathbf{J}) = \frac{\sum_{x=0}^{M-1} \sum_{y=0}^{N-1} \left(\mathbf{I}[x,y] - \bar{\mathbf{I}}\right)\left(\mathbf{J}[x,y] - \bar{\mathbf{J}}\right)}{\left\|\mathbf{I} - \bar{\mathbf{I}}\right\| \cdot \left\|\mathbf{J} - \bar{\mathbf{J}}\right\|}.$$

medie dei valori

Se $\rho \to 1 \Rightarrow$ si ha correlazione

$$\|\mathbf{I} - \bar{I}\| = \sqrt{\sum_{x=0}^{M-1} \sum_{y=0}^{N-1} (I[x,y] - \bar{I})^2}.$$

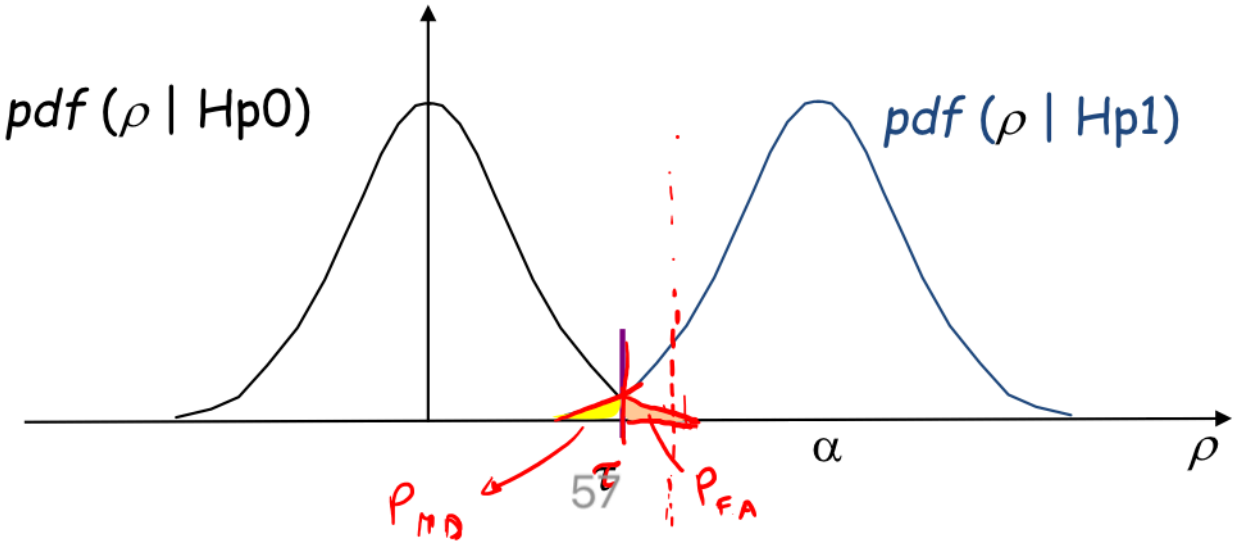
NCC sarà quindi una variabile randomica che dipenderà da tutti i tipi di disturbi con una valore positivo quando rispetta ipotesi H_1 altrimenti H_0. La varianza di entrambe incrementa con la diminuzione della dimensione dell'immagine mentre può essere inaffidabile nel caso si sia usato poche foto per fare la stima del PRNU.



56

Il threshold(o soglia) viene decisa considerando due tipi di errori:

- **Falso Allarme**: immagine non contiene K nonostante il detector decide che K è presente in W_Q (area arancione).
- **Missing Detection**:l'immagine contiene K ma il detector afferma che K non è presente. (area gialla).



Ma che succede se si attua un cropping della foto?

Si desincronizza tutto quindi non è più possibile trovare una corrispondenza con la Fingerprint della foto, quindi in presenza del cropping:

$$\max_{s_1,s_2} \rho(s_1,s_2;I,J) = \rho(s;I,J) = \frac{\sum_{x=0}^{M-1}\sum_{y=0}^{N-1}(I[x,y]-\bar{I})\cdot(J[x+s_1,y+s_2]-\bar{J})}{||I-\bar{I}||\cdot||J-\bar{J}||}$$

Questo valore viene calcolato per ogni possibile shift spaziale con tutti i possibili valori che lo shift può prendere. Facendo così, presa la max correlazione ρ_{peak} e il suo corrisposto shift s_{peak}, allora se $\rho_{peak} > \tau$ allora il fingerprint è stato individuato.

Ma si ha un modo più robusto di questo utilizzato il Peak to Correlation Energy (**PCE**) che è definita così:

$$PCE = \frac{\rho(s_{peak},I,J)^2}{\frac{1}{MN-|V|}\cdot\sum_{s\notin V}\rho(s,I,J)}$$

Dove V è un piccolo insieme di vicini con alta corrispondenza.

In presenza di Scaling e/o Rotazione?

Si fa una ricerca Brute Force utilizzando il PCE come funzione utilizzando tutti i possibili fattori di scala e i parametri di. rotazione.

$$P = \max_{(r_i,\beta_j)} PCE(r_i,\beta_j)$$

con $i \in [1,U], j \in [1,V]$

Video Source Identification

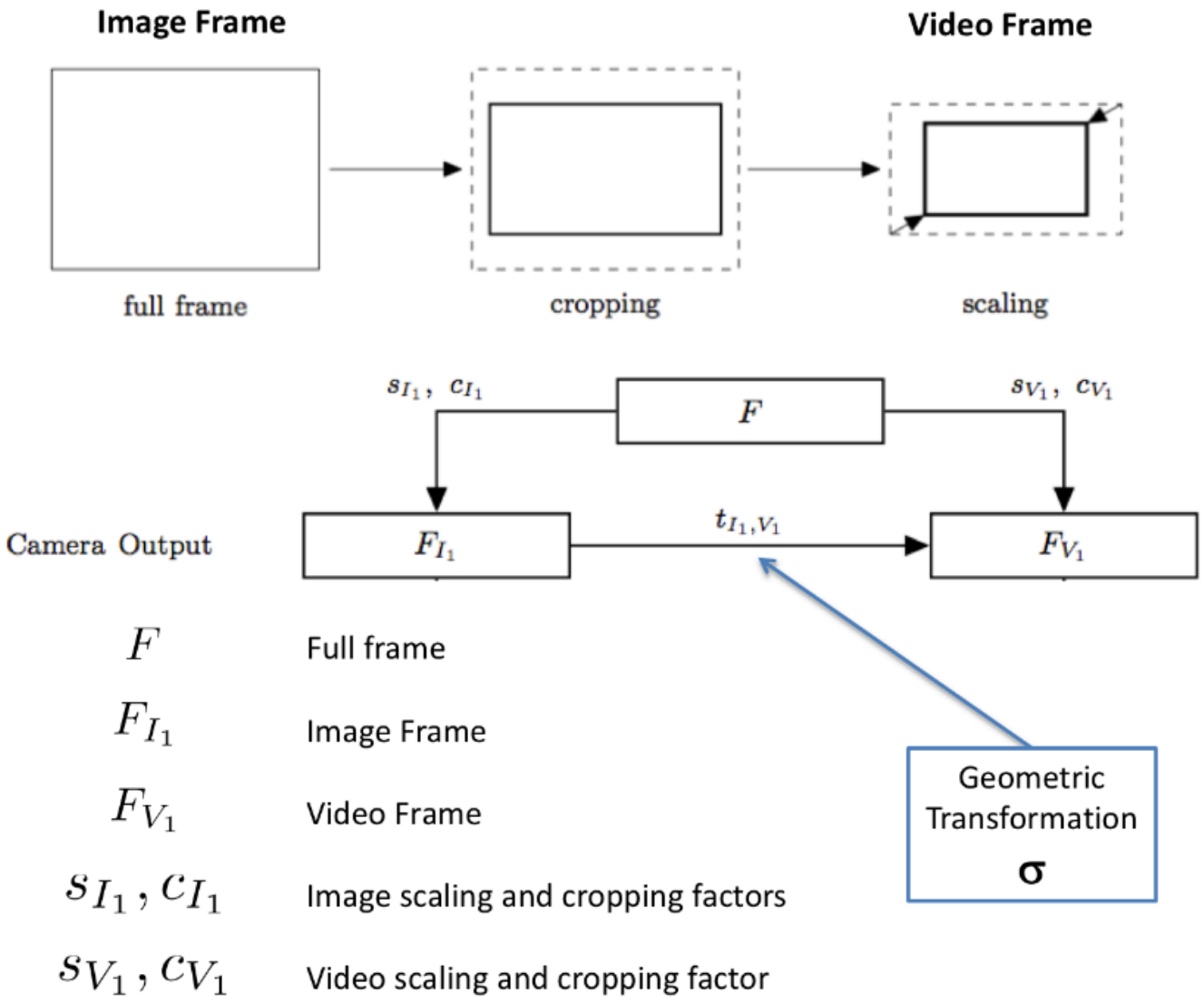
Si utilizza lo stesso ragionamento che si è fatto per le immagini, si prende un insieme di frame, li applichiamo un filtro antirumore, si calcola il rumore e poi si stima K.
Il problema si ha con l' **Hybrid Source Identification** ovvero quando si vuole identificare la sorgente di un video confrontandola con un immagine.

(E' utile perchè l'85% dei media condivisi sono catturati usando i cellulari con cui fanno sia video che foto)

Problema:
I video sono catturati con una risoluzione minore rispetto alle immagini, infatti i video casomai possono essere catturati in 4K utilizzando un frame di 8 Megapixels mentre le immagini possono essere catturate con 20 Megapixels.
Inoltre durante la registrazione del video, viene fatto un crop centrale per adattare la grandezza del sensore a ratio desiderato (solitamente 16:9).

Di conseguenza i PRNU preso dalle immagini e dai video non possono essere comparati direttamente e spesso è inefficace fare uno scaling della foto (*immagine viene **ridimensionata** per adattarsi a nuove dimensioni, mantenendo le proporzioni originali. I pixel vengono scalati in base al rapporto di ingrandimento o riduzione.*) a causa del cropping(*tagliare via una sezione del frame*).

Quindi le immagini e i video acquisiti dallo stesso dispositivo sono legati da una concatenazione di trasformazioni.



Quindi l'**Hybrid Source Identification** consiste nell'individuare la fonte di un video in base a un riferimento derivato da fermi immagini.

K_I viene stimato da N immagini → Fingerprint

K_V viene stimato da N frame del video su cui investigare con risoluzione $m_v \times n_v$

Questi due fingerprint sono diversi e abbiamo bisogno di una **Trasformazione Geometrica** σ, ci aspettiamo infatti una trasformazione che individui K_V in K_I di conseguenza:

$$K_I(si + t_x, sj + t_y) = K_V(i, j)$$

con $0 < s < 1$ e :
$\forall i = 1, \ldots, m_I$ with $m_I > m_V$
$\forall j = 1, \ldots, n_I$ with $n_I > n_V$

allora:

- H_0 : $K_I^\sigma \neq K_V \forall \sigma \rightarrow$ non si ha nessuna corrispondenza.
- H_1 : $K_I^\sigma = K_V \exists \sigma \rightarrow$**corrispondenza, si fa quindi il** $PCE(K_I^\sigma, K_V)$ **e si vede se stesso K**.

E' molto costoso computazionalmente ma i fattori di cropping e resize sono determinati dal modello del dispositivo, è quindi possibile costruire una tabella di ricerca che elimina la necessità di una ricerca esauriente quando è disponibile il modello del dispositivo.

Stabilizzazione Elettronica dell'Immagine (Electronic Image Stabilization)

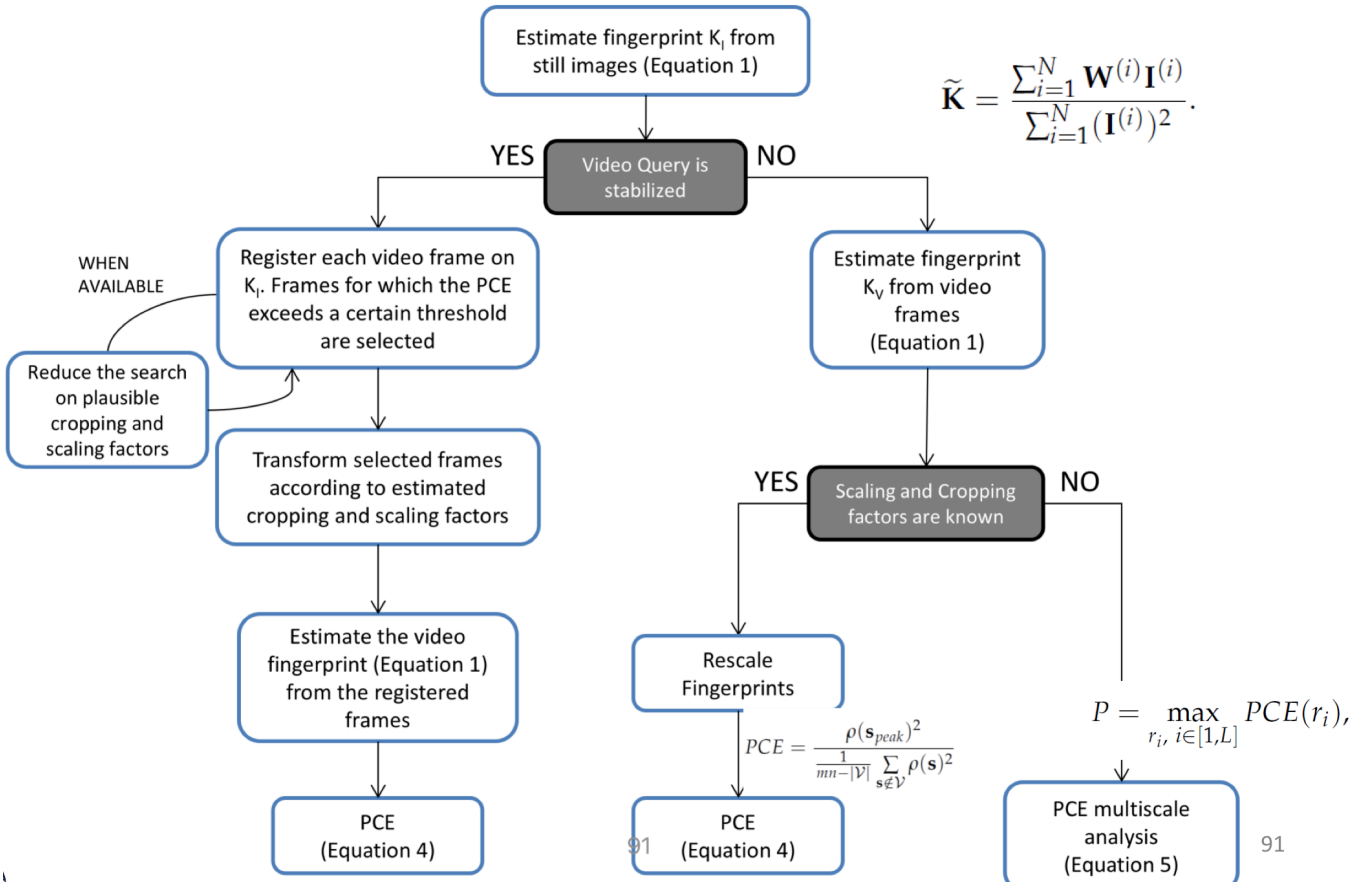
I dispositivi moderni hanno tutti una modalità per la stabilizzazione della telecamera (EIS), questo viene fatto stimando i movimenti dell'utente che riprende.
Gli approcci all'avanguardia eseguono la stabilizzazione video adattando il percorso originale della fotocamera 2D con una data trasformazione geometrica: a seconda della complessità è caratterizzato da una quantità diversa di gradi di libertà (DOF).
• Solo traduzione → 2 DOF
• Traduzione + ridimensionamento + rotazione → 4 DOF

Una conseguenza è che due pixel che condividono le stesse coordinate su due diversi frames potrebbero essere stati acquisiti con diverse porzioni del sensore della fotocamera

Come si identifica da un video con EIS?

Si analizza i primi dieci frame di un video stabilizzato

- Ogni frame ha un K_I
- Solo i fotogrammi videoregistrati il quale PCE supera la soglia di aggregazione sono utilizzati per stimare K_V.
- Una volta che entrambi le impronte K_I e K_V sono disponibili, il loro PCE è compiuto il valore di correlazione viene confrontato con il threshold per decidere tra H_0 e H_1.



91

Video Dataset

ID	Model	Image Resolution	Video Resolution	Digital Stab
C1	Galaxy S3	3264 × 2448	1920 × 1080	off
C2	Galaxy S3 Mini	2560 × 1920	1280 × 720	off
C3	Galaxy S3 Mini	2560 × 1920	1280 × 720	off
C4	Galaxy S4 Mini	3264 × 1836	1920 × 1080	off
C5	Galaxy Tab 3 10.1	2048 × 1536	1280 × 720	off
C6	Galaxy Tab A 10.1	2592 × 1944	1280 × 720	off
C7	Galaxy Trend Plus	2560 × 1920	1280 × 720	off
C8	Ascend G6	3264 × 2448	1280 × 720	off
C9	Ipad 2	960 × 720	1280 × 720	off
C10	Ipad Mini	2592 × 1936	1920 × 1080	on
C11	Iphone 4s	3264 × 2448	1920 × 1080	on
C12	Iphone 5	3264 × 2448	1920 × 1080	on
C13	Iphone 5c	3264 × 2448	1920 × 1080	on
C14	Iphone 5c	3264 × 2448	1920 × 1080	on
C15	Iphone 6	3264 × 2448	1920 × 1080	on
C16	Iphone 6	3264 × 2448	1920 × 1080	on
C17	Lumia 640	3264 × 1840	1920 × 1080	off
C18	Xperia Z1c	5248 × 3936	1920 × 1080	on

1. Lato di Riferimento (Reference Side) :
Questa parte rappresenta il **set di riferimento**, usato per **calibrare o confrontare i dati** provenienti dal set di query.
E' formato da:
• **100 immagini flat-field:**
• Immagini di superfici uniformi (cieli o muri).
• Utilizzate per **isolare il rumore del sensore (PRNU)** senza interferenze provenienti dai dettagli della scena.
• **150 immagini di scene interne ed esterne:**
• Fotografie di ambienti interni (stanze, corridoi) ed esterni (paesaggi, edifici).
• Servono per testare l'efficacia degli algoritmi su **scenari realistici e variegati**.
• **1 video del cielo con movimento lento (>10s):**
• Utile per **analizzare il comportamento del rumore nei video**, riducendo al minimo la complessità visiva.

2. Lato di Query (Query Side) :
Questa parte rappresenta il **set di test**, dove i dati sono utilizzati per valutare algoritmi o modelli.

• **Video di superfici piatte, scene interne ed esterne:**
Tre categorie principali:

1. **Superfici piatte (flat)**
2. **Scene interne (indoor)**
3. **Scene esterne (outdoor)**

Ogni categoria contiene almeno **3 tipi di video**, differenziati dal tipo di movimento della fotocamera:

• **(i) Fotocamera fissa (still camera)** → Nessun movimento.
• **(ii) Operatore in movimento (walking operator)** → Movimento lieve e controllato.
• **(iii) Panning e rotazione (panning and rotating camera)** → Movimenti ampi e rotatori della telecamera.

Almeno 9 video per dispositivo: Ogni dispositivo (es. smartphone, fotocamera digitale) ha almeno **9 video** (3 categorie × 3 tipi di movimento). Ogni video dura più di **60 secondi**, garantendo una durata sufficiente per l'analisi.

K_I è stimato con 100 immagini piatte. Mentre K_V viene stimato con 100 frame del video.
In questi Dataset ci sono per ogni dispositivo i parametri di rescaling e cropping (con e senza la stabilizzazione).

ID	Scaling	Central Crop along x and y	Rotation (CCW)
C10	[0.806 **0.815** 0.821]	[243 **256** 261] [86 **100** 103]	[−0.2 **0** 0.2]
C11	[0.748 **0.750** 0.753]	[380 **388** 392] [250 **258** 265]	[−0.2 **0** 0.2]
C12	[0.684 **0.689** 0.691]	[287 **294** 304] [135 **147** 165]	[−0.2 **0** 0.6]
C13	[0.681 **0.686** 0.691]	[301 **318** 327] [160 **181** 195]	[−0.4 **0** 1]
C14	[0.686 **0.686** 0.689]	[261 **301** 304] [119 **161** 165]	[−0.4 **0** 0]
C15	[0.696 **0.703** 0.713]	[298 **322** 345] [172 **190** 218]	[−0.2 **0.2** 1.6]
C16	[0.703 **0.706** 0.708]	[315 **323** 333] [178 **187** 201]	[−0.2 **0.2** 0.4]
C18	[0.381 **0.384** 0.387]	[548 **562** 574] [116 **121** 126]	[0 **0** 0]

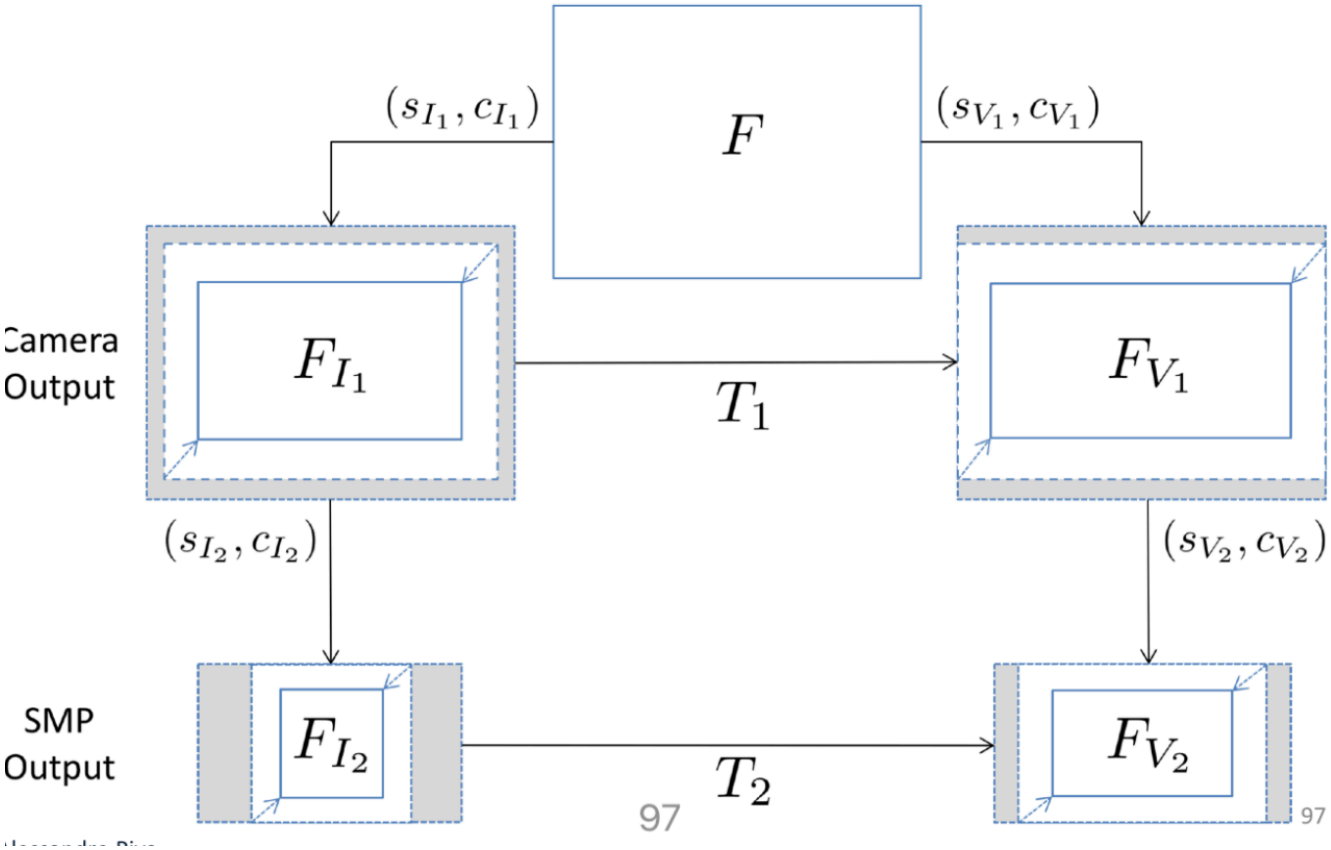
In questa figura, in alto, si hanno i parametri di scaling, cropping e rotazione linkati ai dispositivi, questi valori sono stati calcolati sui primi dieci frame le video di riferimento con i valori **min, median,max**.

Piattaforme Social Media

Le immagini e video che vengono postate sui Social sono modificate prima di essere postate:

- Vengono ridimensionate (ma l'Aspect Ratio viene preservato, ovvero il rapporto larghezza,lunghezza)
- Compresse
  Queste due trasformazioni vengono considerate come delle trasformazioni geometriche.

Workflow of the geometric transformation



Il diagramma spiega come le immagini/video subiscono **trasformazioni geometriche sequenziali** durante il processo di caricamento e pubblicazione su piattaforme social. Ogni passaggio può alterare parametri critici come **scaling (s)** e **compressione (c)**, influenzando l'analisi forense delle immagini (es. identificazione tramite PRNU).

1. F
   • Rappresenta la **funzione di trasformazione geometrica** che agisce sulle immagini/video.
   • Questa trasformazione può includere operazioni come **ridimensionamento (resizing)**, **compressione** o altre modifiche geometriche.
2. F_{I_1} e F_{V_1}
   • F_{I_1} **:** Rappresenta il **frame iniziale** di un'immagine/video ottenuto direttamente dalla **fotocamera (Camera Output)**.
   • F_{V_1}**:** È il frame trasformato dopo essere stato processato dalla **funzione** F (ad esempio durante il caricamento su una piattaforma social).
3. **Parametri (s, c)**
   • Ogni frame ha associati dei parametri:
   • s : Parametro di **scaling** (ridimensionamento).
   • c : Parametro di **compressione** o altre caratteristiche geometriche.
   • Questi parametri cambiano ad ogni trasformazione.

4. T_1 **e** T_2
   • T_1 **:** Trasformazione che avviene tra il frame originale della fotocamera (F_{I_1}) e il frame processato (F_{V_1}).
   • T_2 **:** Trasformazione che avviene tra il frame successivo (F_{I_2}) e (F_{V_2}), dopo un'ulteriore elaborazione o ri-adattamento.
5. F_{I_2} **e** F_{V_2}
   • F_{I_2} **:** Frame finale ottenuto dopo ulteriori trasformazioni geometriche (Social Media Platform - SMP Output).

   • F_{V_2} **:** Frame finale del video dopo l'elaborazione completa su piattaforme di social media.

Profile Linking:

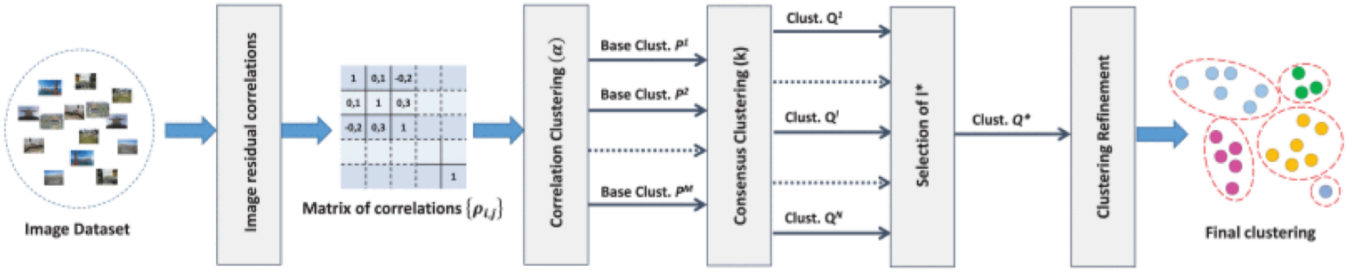
Serve a capire quando due contenuti sono stati catturati dallo stesso dispositivo, qui i fattori di cropping e di scaling devono essere numericamente stimati poichè:

- il dispositivo potrebbe essere sconosciuto
- E il Social Media Platform applica i suoi fattori di resize. e scaling.

Un altro problema è che i profili dei Social Media Platform contengono contenuti che non provengono direttamente dal solito dispositivo, quindi l'SPN (Sensor Pattern Noise) non può essere stimato da tutto il media disponibile.
SPN ricordiamo include più tipi di rumore in confronto al PRNU che contiene i rumori dato da la **la variazione di sensibilità dei pixel.**
Abbiamo bisogno di uno step per raggruppare i media che appartengono ad un specifico dispositivo.
E si fa raggruppando i media che hanno il rumore **più simile**.

Clustering

Per valutare il grado di similitudine tra i rumori residui si usa una matrice di correlazione, le immagini sono raggruppate e i gruppi vengono soggetti a rifinizioni per rimuovere quelli in comune.



$$\rho_{i,j} = \langle W_i, W_j \rangle$$

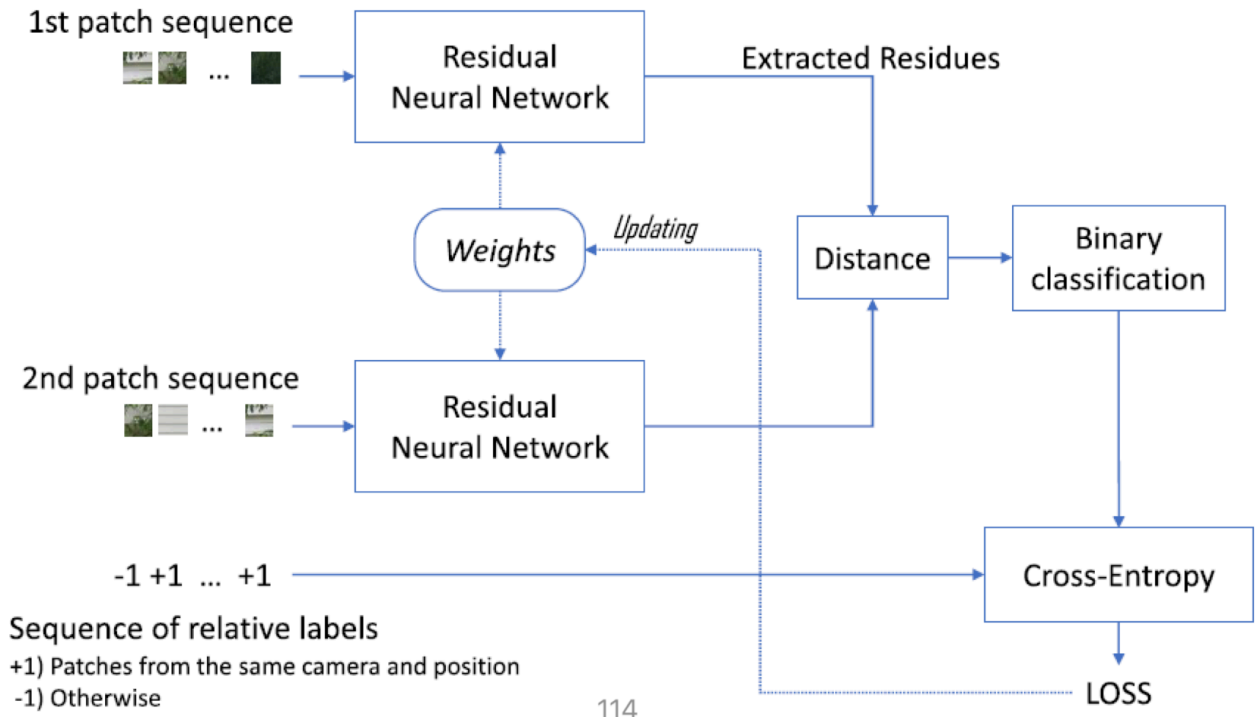
I SMP fanno già loro un clustering e hanno una classifica della copertura.

SMP	Average Correct Classification
Flickr	95.7
Google+	96.5
Tumblr	97
Twitter	37.5
Instagram	84
Tinypic	100

Il residuo del rumore (noise residual) è il segnale di rumore una volta che viene applicato il filtro antirumore alla foto. Esso non contiene solo il PRNU ma anche altri artefatti relativi alla fotocamera. Non è specifico del device, infatti abbiamo provato a rimuoverli quando si lavorava con il PRNU post processando il rumore rimasto.
Comunque, questi artefatti possono essere preservati ed evidenziati ottenendo il **Noiseprint** per permettere l'identificazione del modello del dispositivo.

Otteniamo questo attraverso una rete CNN (**Convolutional Neural Networks**) ispirata alle **Reti Siamesi** (due CNN con stessa architettura & stessi pesi), una volta che la fase di *training* la rete si ferma e può usare le immagini catturate da qualsiasi modello di camera (sia fuori che dentro gli insiemi di dispositivi con il quale è stato fatto il training).
L'obiettivo di questa rete neurale è estrarre il pattern del rumore che ha l'immagine passata in input. Queste reti tengono le loro architetture e le inizializzano con i parametri ottimali per AWGN (**Additive White Gaussian Noise**) Image Denoising. Poi aggiornano i parametri attraverso una fase di training.

Procedura di Apprendimento: minimizzano la distanza tra zone di stesse fotocamere.



Spiegazione del Diagramma sulla Residual Neural Network per NoisePrint

Questo diagramma rappresenta il **flusso di lavoro di un sistema di addestramento basato su una Residual Neural Network (ResNet)**, utilizzato per estrarre e confrontare i **residui di rumore** (Noise Residues) da immagini o patch di immagini, con l'obiettivo di identificare se provengono dalla **stessa fotocamera e posizione**.

1. Input: Sequenze di Patch

• **1st patch sequence:**
• Una sequenza di **patch** (piccole porzioni di immagine) viene fornita come input alla prima **Residual Neural Network**.

• **2nd patch sequence:**
• Un'altra sequenza di **patch** viene fornita a una **seconda Residual Neural Network**.

Le patch possono provenire dalla **stessa immagine** o da immagini differenti.

2. Residual Neural Network (ResNet)
• Ogni sequenza di patch passa attraverso una **Residual Neural Network**.
• Il compito delle ResNet è **estrarre i residui di rumore** dalle immagini, rimuovendo le informazioni del contenuto visivo.
• I residui rappresentano le **impronte digitali del sensore** (come il PRNU) e altri schemi di rumore.

3. Estrazione e Confronto dei Residui
• I **residui estratti** vengono confrontati utilizzando una **metrica di distanza**.

• La distanza misura **quanto sono simili i due residui di rumore**:
• **Distanza piccola:** I residui provengono probabilmente dalla **stessa fotocamera e posizione**.
• **Distanza grande:** I residui provengono probabilmente da **fotocamere diverse o posizioni diverse**.

4. Binary Classification

• Il valore della distanza viene utilizzato per una **classificazione binaria**:
• **+1:** Le patch provengono dalla **stessa fotocamera e posizione**.
• **-1:** Le patch provengono da **fonti diverse**.

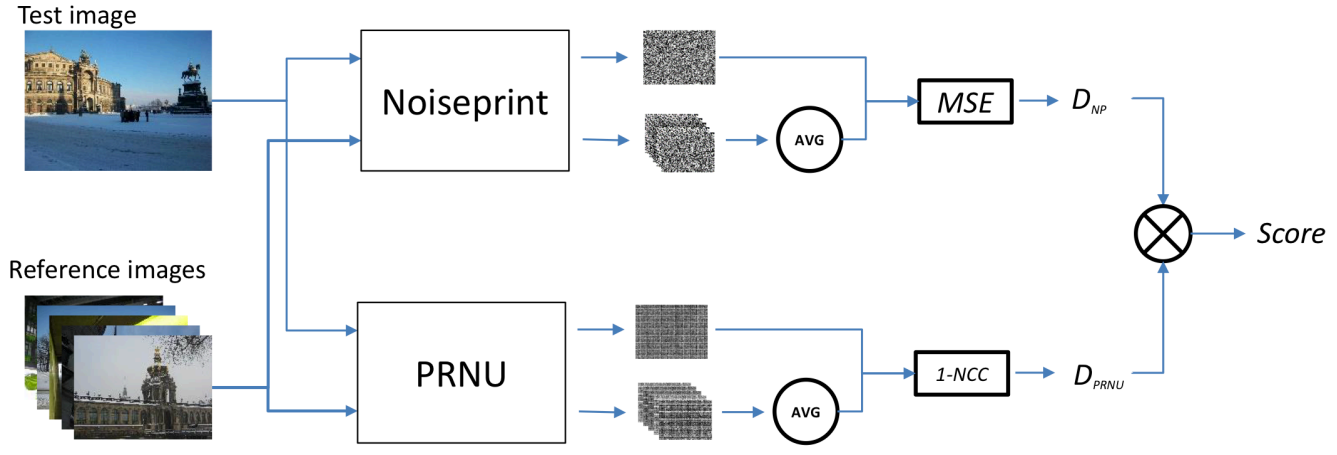
5. Funzione di Perdita (Cross-Entropy Loss)
• La **Cross-Entropy Loss** viene calcolata confrontando il risultato della classificazione con le **etichette relative** (+1 o -1).
• L'obiettivo della loss è **minimizzare l'errore di classificazione**.
• I **pesi della Residual Neural Network vengono aggiornati** attraverso il processo di backpropagation per migliorare l'accuratezza della classificazione.

6. Aggiornamento dei Pesi
• I **pesi delle due Residual Neural Network** vengono aggiornati durante l'addestramento in base alla perdita calcolata.
• Questo permette alla rete di migliorare la capacità di **estrarre residui di rumore significativi** e di distinguere meglio patch provenienti dalla stessa o da fotocamere diverse.

Per identificare il modello ha un accuratezza del 100% mentre il PRNU ha il 77%.
Mentre per identificare il dispositivo PRNU ha il 70% e il metodo con il Noiseprint 62%

Questo rumore residuo estratto migliora le tracce che vengono da un eventuale manomissione.

Se si combinano i due metodi, si può ottenere uno score totale.



Nel noiseprint si fa una media e poi si fa MSE(Medium Square Error), mentre nell'PRNU si fa 1-NCC ovvero **Normalize Cross Correlation** e da li si ottiene il punteggio.