

Sistema elettronico intelligente di videosorveglianza indoor per tracciamento di persone

Relatore: Chiar.mo Prof. Ing. RODOLFO ZUNINO

Correlatore: Ing. PAOLO GASTALDO

Candidato: SAVERIO ZUCCOTTI

Nel corso degli ultimi anni le applicazioni di videosorveglianza hanno conosciuto un forte sviluppo, evolvendo verso sistemi automatici in grado di monitorare l'area assegnata anche senza l'operatore umano. Questo è reso possibile da particolari procedure che consentono di estrarre dalle immagini di un filmato le eventuali sagome umane presenti e di inseguirle, tracciandole, durante tutta la permanenza sulla scena.

Lo svolgimento della tesi ha riguardato le modifiche da apportare ad un sistema di videosorveglianza preesistente al fine di semplificare la sequenza di operazioni da esso compiute. Infatti, sotto determinate ipotesi operative, è possibile evitarne alcune così da diminuire l'onere computazionale complessivo e velocizzare i tempi di esecuzione.

Il lavoro oggetto di trattazione si è quindi innestato su un sistema già funzionante messo a punto nel corso degli ultimi anni dai vari tesisti che si sono avvicinati sul progetto.

Il principio di funzionamento è piuttosto semplice: una telecamera dotata di alzo e brandeggio riprende una scena trasmettendo le immagini acquisite ad un computer collegato. Un apposito software, denominato *VideoSurv*, provvede a confrontare ciascun fotogramma con il suo sfondo, ovvero con l'immagine della scena ripresa in assenza di intrusi. In particolare, l'algoritmo utilizzato implementa una tecnica di tipo *template matching* basata sulla trasformata della distanza. Non appena viene individuata una sagoma umana, viene associato ad essa un rettangolo colorato che la seguirà per tutta la sua permanenza sulla scena. Questo tracciamento è reso ancor più efficace dalla capacità del sistema di ordinare alla telecamera opportune rotazioni volte a mantenere l'intruso sempre al centro dell'inquadratura. Questa eventualità, tuttavia, comporta il ricalcolo e

l'aggiornamento dello sfondo. In ogni caso, *VideoSurv* provvede a visualizzare su monitor l'immagine che era stata acquisita, con l'aggiunta però un rettangolo colorato per ogni match.

A tutto ciò corrisponde un'ideale architettura software multithread che realizza l'intero ciclo di operazioni. Si avranno quindi sei thread che, attivati in maniera sequenziale sugli elementi di un buffer che contiene le immagini acquisite dalla telecamera, elaborano ciascun fotogramma dall'istante della sua acquisizione fino al momento della generazione dell'output.

Nell'ordine di attivazione, quindi, i sei thread sono quello di cattura (*grabbing*), conversione dal formato JPEG a Bitmap (*decompression*), aggiornamento dello sfondo (*background*), ricerca di profili umani (*detection*), tracciamento e movimento della telecamera (*tracking*) e visualizzazione (*view*).

Nonostante questa architettura garantisca la piena efficacia, ci si è chiesto se esistano casi in cui è possibile evitare l'applicazione dell'intero ciclo di operazioni. Ad esempio, se la telecamera non si è spostata e lo sfondo è quindi rimasto invariato, si può pensare di sottoporre il fotogramma ad una sommaria analisi preventiva. Così facendo, si stabilisce se l'eventuale soggetto prima individuato ha davvero compiuto spostamenti sensibili o se altri intrusi sono comparsi sulla scena. Qualora non si sia verificata nessuna di queste due circostanze, si può concludere che non vale la pena applicare al frame corrente la detection completa.

Questo nuovo modulo introdotto prende il nome di *pre-detection*. L'algoritmo prescelto è volutamente semplice per renderlo il più rapido possibile. Per prima cosa, infatti, si provvede a realizzare una *simple-difference* tra l'ultimo fotogramma sottoposto a detection e quello corrente.

L'immagine binaria ottenuta viene virtualmente suddivisa in strisce verticali larghe quanto il template in uso. A ciascuna di esse è associato un contatore per la stima delle dimensioni dei blob ricavati dalla *simple-difference*. Se in nessuna striscia il contatore supera un certo valore di soglia, si conclude che sulla scena non è intervenuto alcun cambiamento sostanziale.

L'architettura del sistema viene quindi modificata sia per valutare le ipotesi di applicazione della *pre-detection*, sia per realizzare la medesima. A valle del thread di decompressione si valuta se il frame corrente sia stato ottenuto o meno dopo un movimento della telecamera.

In caso di risposta affermativa, il sistema procede con la vecchia sequenza preesistente. Viceversa, se non vi è stato movimento, si passa il fotogramma al thread di pre-detection. Qualora questi si concluda con successo (nessuna grossa novità riscontrata), il ciclo di elaborazione si conclude e il sistema si prepara ad acquisire una nuova immagine. In caso contrario, il fotogramma deve essere passato a tutti gli altri thread per ricevere la detection. Si può notare che le condizioni di attivazione dei thread di aggiornamento dello sfondo e di pre-detection sono mutuamente esclusive, nel senso che le condizioni di stasi o di movimento della telecamera escludono a seconda dei casi uno dei due thread.

La nuova architettura che si è ottenuta modificando quella preesistente è stata quindi testata mediante una serie di sessioni di lavoro sperimentali, nel corso delle quali si è valutato il comportamento e le prestazioni del sistema.

Le prove condotte hanno dato esito pienamente soddisfacente, considerato che la pre-detection ha operato correttamente e secondo le previsioni.

In particolare, sono stati effettuati diversi tentativi variando di volta in volta le dimensioni del template o il valore di soglia utilizzato per interpretare il risultato della simple-difference.

Si è così osservato che, a spostamenti minimi del soggetto rilevato, corrispondeva in fase di pre-detection un'immagine binaria con blob sostanzialmente trascurabili. Viceversa, movimenti più marcati determinavano un graduale aumento di pixel neri finché il loro numero non diventava fatale e provocava il fallimento della pre-detection stessa.

Per quanto riguarda le prestazioni del sistema, i tempi rilevati hanno ulteriormente confermato la validità dell'idea di pre-detection: in condizioni ottimali, il tempo del ciclo di elaborazione del singolo fotogramma è passato dai 355 millisecondi ottenuti con l'architettura originaria ai 160 raggiunti grazie alla pre-detection. Ciò vuol dire che l'abbattimento dei tempi ha superato il 50%.

Anche nei casi in cui la pre-detection si è conclusa con il fallimento, i tempi complessivi di esecuzione si sono attestati mediamente sui 340 millisecondi, di cui solamente il 7% è imputabile all'onere aggiuntivo introdotto dalla pre-detection.