Panoramica sulle tecniche di controllo del traffico urbano utilizzanti la Visione Artificiale.

Bologna 4/5/'98 Autore: Enrico Viarani
eviarani@deis.unibo.it – tel ++39+51 6443545

1 - Introduzione

In molti casi si ricercano soluzioni di machine-vision applicate al monitoraggio del traffico perché risultano poco convenienti le spire induttive poste nel manto stradale sia per la difficoltà ed i costi di manutenzione, sia per la bassa qualità dei dati da esse forniti.

La relazione si limita al materiale inerente la computer-vision applicata al controllo del traffico ed in particolare ai tipi di approcci per la raccolta di dati in prossimità degli incroci. Essa si presenta come una sorta di classificazione delle soluzioni incontrate.

2 - Approcci "tripwire" e "tracking"

In generale esistono due tipi di approcci di machine-vision [10] per il controllo del traffico. Nel primo caso ci si riferisce a quei sistemi (detti "tripwire", come AUTOSCOPE e CCATS) in cui si effettua il rilevamento su traguardi sensibili di pixel aventi forma allungata e sottile, come un filo; essi discendono dalla filosofia delle spire e sono utili laddove si vuole conservare il modello da loro usato. Nel secondo caso si tratta di sistemi (di tipo "tracking", come EVA ed IDET) in cui le finestre sensibili sono più estese e si parla di "maschere": questo rende possibile l'inseguimento del veicolo all'interno di una finestra.

I sistemi "tripwire" sono più adatti a raccogliere dati lungo arterie stradali ove è dato per normale un flusso unidirezionale, il numero degli accessi laterali e delle intersezioni è ridotto e la situazione di congestione è un evento raro. Essi consentono di effettuare il conteggio dei veicoli, di misurarne la velocità media su tratti di qualche decina di metri e di effettuarne una classificazione sommaria, sulla base della loro lunghezza, tra un massimo di due misure possibili. Trovano facilmente applicazione in autostrade e superstrade.

Questi sistemi richiedono un semplice setup con pochi parametri da inserire, sono poco sensibili nei confronti delle oscillazioni delle telecamere, implicano un basso carico computazionale grazie al limitato numero di pixels in esame. Essi sono però vulnerabili nei confronti di riflessi ed ombre, necessitano di complessi algoritmi per l'esercizio notturno e di accuratezza nella scelta delle dimensioni dei traguardi sensibili, al fine di evitare errate associazioni tra fari e numero di veicoli transitati e alee riguardo l'appartenenza a più corsie dello stesso veicolo.

Tuttavia, come narra la letteratura più recente, raffinando molto software e modelli dei sistemi "tripwire", essi trovano anche applicazioni urbane per il controllo delle intersezioni.

Agli incroci stradali nasce una coda quando i veicoli si fermano alla linea di stop con semaforo rosso; la fine della coda si sposta indietro ad ogni nuovo veicolo che sopraggiunge. Essa comincia poi a dissolversi (gestione FIFO) quando i veicoli in testa ripartono (il concetto di coda di automezzi è stato anche studiato in maniera più esaustiva da Quinn [33] che lo analizza in termini di "onde d'urto di traffico" generate dalla cadenza dei semafori. Il modello non considera solo le ripercussioni locali, ma predice gli effetti su più intersezioni.

I sistemi tripwire caratterizzano la coda di un incrocio segmentando la lunghezza delle corsie per mezzo dei traguardi e calcolando il numero dei veicoli presenti in un tronco come differenza tra quelli entrati e quelli usciti. Questi sistemi prescindono dalla descrizione del moto dei veicoli, il cui studio non è richiesto, se non nei casi in cui sia necessario il calcolo della loro velocità media che si misura come segue: il veicolo che transita sul traguardo produce una variazione dei valori dei suoi pixel che supera la soglia di background avviando così un cronometro; al traguardo successivo (posto sufficientemente vicino in modo da non creare ambiguità) si calcola la velocità media tra i due traguardi sulla base del tempo trascorso. Per la gestione delle code di veicoli i sistemi tripwire implementano uno stack che le rappresenta virtualmente ed alle cui locazioni sono associati i veicoli. La "coda" ha una collocazione spaziale generica all'interno del segmento di carreggiata. Essa esiste dal momento in cui il bilancio in/out non è in pareggio e sul traguardo normalmente situato sulla linea di stop c'è un veicolo fermo (o non è segnalato transito). Coi sistemi tripwire si possono anche descrivere gli stati di "congestione" o "rallentamento" del traffico: essi dipendono dalla media delle velocità medie.

I sistemi con approccio di tipo "<u>tracking</u>" sono utilizzati per risolvere situazioni ove interessi sapere 'quanti' veicoli siano presenti e 'dove' nella scena.

I sistemi di tipo tracking forniscono direttamente più informazioni rispetto ai sistemi tripwire poiché sono sensibili ad un maggiore numero di caratteristiche ottiche. Per questo motivo richiedono routines di software che li rendano immuni anche dal maggior rumore acquisito.

Allo scopo di ridurre il carico computazionale di image-processing si delineano finestre all'interno delle quali si svolge il moto che ci interessa caratterizzare. I sistemi tracking misurano i parametri caratteristici della coda a partire dal moto che deve necessariamente essere rilevato tra due fotogrammi e dal quale non possono prescindere: grazie al moto si possono rilevare i veicoli che saranno etichettati e dunque collocati in punti precisi nell'immagine della scena. L'operazione appena menzionata ne implica il conteggio. In questo caso la gestione parallela della "coda virtuale" non è indispensabile ma, in pratica, evita quella serie di errori come la 'fusione' di due veicoli contigui con le stesse caratteristiche di moto o l'assimilazione di un veicolo fermo allo sfondo.

Utilizzando la loro capacità di compiere misure dimensionali essi risolvono e associano correttamente le ombre ai rispettivi veicoli e riconoscono i cambi di corsia; durante le riprese notturne sono in grado di associare gli spot dei fari ai rispettivi veicoli che possono classificare in tre differenti categorie, al massimo; hanno anche una buona precisione sulla misura della velocità (migliore del 95%).

I sistemi tracking sono sensibili alle condizioni di illuminazione: fondi stradali illuminati da luce scarsa o colorata rendono più difficile a questi sistemi il riconoscimento dei veicoli.

Gli algoritmi di tracking sono sensibili alle oscillazioni della telecamera il cui angolo di ripresa è di fondamentale importanza soprattutto per le conseguenze che esso ha sulla occlusione della vista dei veicoli retrostanti. Le camere dovrebbero essere posizionate alla massima altezza possibile (almeno 7.5m, meglio 9m) col resto del sistema alloggiato in un armadio. Le telecamere possono essere montate su ponti o su pali, meglio se tra le corsie, perché ciò consente di controllare anche il flusso opposto, rende l'occlusione meno evidente e si presta meno ad atti vandalici (il discorso vale anche per gli armadi). Sui sistemi tracking si possono integrare optional come lettura targhe, rilevamento incidenti, conteggio dei cambi di corsia e compressione delle immagini.

I dati così raccolti sono codificati e memorizzati localmente e costituiscono un archivio 'storico' a cui il singolo sistema attinge per auto-reinizializzarsi.

3 - Sistemi Tripwire

Questo tipo di sistemi trae spunto dalla filosofia di quelli dotati di spire induttive. I sistemi a spire e quelli machine-vision di tipo tripwire hanno in comune il modello fisico ma differiscono per il tipo di trasduttore: spire induttive nei primi, telecamera a CCD nei secondi. Quindi anche i set di variabili utilizzati dal software dei due sistemi saranno simili.

Il sistema tripwire collegato alla camera esamina solo alcuni traguardi posti longitudinalmente e trasversalmente alle corsie. Il primo vantaggio che se ne trae è quello di un limitato costo computazionale dovuto a image-processing. Il semplice criterio con il quale si stabilisce la presenza di un veicolo sul traguardo è quello della 'differenza entro soglia' dei valori dei pixel dallo sfondo corrente.

Con opportune correlazioni sui traguardi si possono estrapolare la velocità media (correlandone due successivi) e la lunghezza del veicolo (misurando il tempo di transito a una data velocità). Il numero limitato di pixel esaminati in questo approccio garantisce velocità di elaborazione e risultati soddisfacenti, se non sono richieste accurate misure cinematiche. Con riferimento specifico a [6] la misura real-time del flusso di auto su una carreggiata basata su edge detection con comparazione a soglia dinamica dà risultati con errore inferiore al 10%, ma non è possibile sfruttare il sistema per il rilevamento di code perché troppo semplicistico e predisposto per operare su autostrade piuttosto che in prossimità di incroci.

Per questo, al peso modesto dell'elaborazione di image-processing di questi sistemi, si può abbinare un modello fisico raffinato e ottenere informazioni più complete come nel caso del celebre sistema tripwire "Autoscope" [7] operante in real-time. Esso si adatta automaticamente a numerosi tipi di sfondo stradale senza la necessità di reference-marks allo startup o durante il funzionamento. I suoi progettisti ne garantiscono il funzionamento in presenza di ombre, riflessi, cambi d'illuminazione; ciò fa supporre che siano stati messi a punto filtri per questo tipo di disturbi basati sull'inseguimento e sul labelling del veicolo. Esso funziona in condizioni di traffico congestionato o rallentato, consente un posizionamento arbitrario delle camere immune, entro certi limiti, dagli angoli di vista e richiede un processore 386 con l'ausilio di un preprocessore per immagini. Tutte queste caratteristiche fanno di Autoscope un sistema tripwire con prestazioni evolute e paragonabili a quelle di un sistema tracking.

In fase d'installazione si configurano i traguardi trasversalmente e longitudinalmente alle corsie (3 o 4 cross-arrays “EDCLA” per ogni camera, come fossero spire lungo la corsia) che permettono la corretta interpretazione dei sorpassi e forniscono informazioni sui cambi d’illuminazione. La riduzione dei disturbi derivanti dalle ombre in movimento avviene attraverso la regolazione dinamica della soglia e la modifica del gain (calcolata sulla base di stime statistiche). L’implementazione real-time degli algoritmi di processing comporta in una prima fase la riduzione dei dati ad una serie di caratteristiche e valori misurati. Successivamente si ricavano le informazioni più dettagliate nello stadio di calcolo a più alto livello.

Per rendere un’idea della filosofia tripwire, nella tabella seguente sono elencate e descritte le variabili utilizzate dal modello di Autoscope.

s	numero stazione (linea di rilevamento)
S	numero totale di stazioni
n	numero di veicoli transitanti su una stazione
l	numero della corsia
L	numero di corsie
Δx_s	spazio tra le stazioni s ed s+1
T	numero di intervalli di scanning aventi durata Δt
t_{sln}	tempo al quale la n-esima macchina nella l-esima corsia giunge alla stazione s
v_{sln}	velocità della n-esima macchina alla stazione s, linea di rilevamento l
Q_{sl}	volume alla stazione s, linea l, sul periodo TΔt
Q_s	volume totale su tutte le corsie della stazione s

Dal punto di vista dei trasduttori per la machine-vision vi è un’ampia possibilità di scelta tra i vari tipi di telecamere a colori o B/W, o all’infrarosso, secondo il tipo di ambiente operativo.

In Autoscope appare un cenno riguardo a prove effettuate con un trasduttore di tipo radar in grado di identificare le masse (veicoli) che riflettano le radioonde elettromagnetiche con lunghezza d’onda di qualche cm. Tale tipo di trasduttore fornirà in ogni modo un’immagine video che sarà però scevra dal rumore dovuto alle condizioni d’illuminazione dell’ambiente, dei fari delle auto e della pubblica illuminazione. Risultati di questo test si sono rivelati migliori di un 5% rispetto a quelli ottenibili con normali telecamere. Sono ovviamente da tenere in considerazione un maggior costo per trasduttore ed il problema dell’inquinamento elettromagnetico al giorno d’oggi non più trascurabile.

4 - Sistemi tracking

Un sistema tracking analizza una o più “finestre” dell’immagine dalla forma sagomata in funzione della scena e delle regioni d’interesse. Si “definisce regione di interesse” di un sistema tracking quel sottoinsieme di pixel di una immagine ove si sviluppa il moto che si intende studiare.

In più di un caso sono stati raggiunti buoni risultati in termini di velocità, costo, affidabilità. Per esempio in [5] si è ottenuta una elaborazione real-time con rate 6 frames/sec con un microprocessore 486-DXII e frame-grabber “OFG” accessorio. Gli intenti iniziali erano quelli di ottenere cicli più brevi di un secondo ed errori inferiori al 10%, il tutto contenuto in un costo di 3000$ per sistema; esso doveva funzionare in qualsiasi condizione meteo e d’illuminazione. Il sistema è dotato di uno stack FIFO per la gestione della coda.

Il software [5] si compone di quattro blocchi: 1-setup, 2-acquisizione, 3-pattern-detection, rimozione dei disturbi, 4-calcolo della lunghezza della coda.

La parte 2-acquisizione consiste nell’estrazione dei contorni degli oggetti e nel calcolo delle primitive della scena invarianti per cambiamenti globali d’illuminazione. L’immagine risultante da questa elaborazione, chiamata Ict, è utilizzata per aggiornare l’immagine di background Ibt. La velocità di aggiornamento dipende sia dal periodo del rosso al semaforo sia dalle condizioni di congestione per non assumere a “sfondo” i veicoli fermi in coda durante attesa o congestione. Tutte le forme che non fanno parte del background sono estratte per differenza tra l’immagine corrente e quella di riferimento |Ict – Ibt|: queste sono veicoli, ma anche insiemi di rumore come riflessi, luci, ombre.

Nella terza parte di pattern-detection, tre moduli rimuovono il rumore e il modulo di tracking rileva e registra tutte le forme in movimento come primitive corrispondenti a veicoli. In questo modo si eliminano le forme non in movimento come luci, ombre e riflessi fermi. Il modulo di variazione toglie le forme introdotte dai fari sulla strada. Queste sono caratterizzate da un falso moto apparente introdotto dai fari (che quindi rende impossibile al tracking la loro eliminazione) e da una stabilità delle primitive che appaiono. Quindi, per rimuoverle, bisogna valutare la stabilità delle primitive tra immagini successive. Il modulo “texture” rimuove le ombre dei veicoli sfruttando la loro natura.

Il quarto stadio calcola la lunghezza della coda. Ogni forma sopravvissuta al passo precedente è etichettata. Si genera una coda se almeno un’etichetta si trova in prossimità della linea di stop. A partire dalla prima posizione ci si muove indietro, di etichetta in etichetta, riempiendo gli spazi vuoti che cadono sotto una soglia il cui valore è scelto durante l’inizializzazione e corrisponde alla massima distanza interveicolare che ci si può ragionevolmente aspettare in una coda (approssimativamente la lunghezza di un veicolo). L’esaurimento della coda è determinato dalla impossibilità di iterare il processo. I risultati del collaudo eseguito con immagini da nastri registrati in condizioni di sole/pioggia, nubi passeggere veloci, cielo coperto hanno stabilito che le condizioni di cielo coperto sono le migliori; il funzionamento notturno ha prestazioni nettamente inferiori a quelle diurne; nelle riprese con sole vi sono molti problemi dovuti al rumore, non totalmente eliminati dal modulo texture.

Una maggiore attenzione al problema notturno è stata dedicata dall’esperimento TITAN [27] per il riconoscimento notturno dei veicoli attraverso immagini con frequenza 4 Hz. Esse sono analizzate in modo indipendente tra loro ed i veicoli sono identificati a partire dai fanali anteriori con un ciclo suddiviso come segue:

1) calcolo dei massimi dell’immagine con soglie adattate. Come risultato si ottengono tutti i fanali, i loro riflessi e i fari pubblici: questo rumore sarà sottratto nei passi successivi;

2) applicazione di un gradiente per distinguere i riflessi (che appaiono spesso come diffusi e meno luminosi) dai veri fari delle auto; restano punti che sono massimi locali e del gradiente;

3) unificazione dei fari e dei riflessi mediante “dilatazione adattata”; questa

trasformazione è basata sul fatto che le intensità dei massimi locali dovuti a riflessi sono minori di quelle dei fari e quindi tendono a sparire dopo la dilatazione;
4) i fari di uno stesso veicolo sono confrontati insieme. Questo metodo, che ci permette di porre un marker su ogni veicolo, elide quelli che hanno solo un fanale.

5 – Studio del moto

Ci occuperemo degli algoritmi in grado di studiare il moto a partire da sequenze di immagini descrivendo anche gli algoritmi accessori a questo scopo, come, ad esempio, quello di segmentazione.
Nei sistemi tracking, lo studio del moto è affrontato generalmente in tre modi: <u>metodi con il gradiente</u>, <u>metodi basati sulle peculiarità</u> (“features”) e metodi <u>RHT</u>.

I secondi sono generalmente di difficile applicazione nel nostro contesto; si possono infatti ricercare caratteristiche comuni a scene allo scopo di inizializzare le regioni di interesse del sistema [12] ma, date le dimensioni in pixel dei veicoli, è difficile estrarre da loro features, come invece è stato fatto [3], allo scopo di classificarli ma a partire da riprese laterali e dove essi occupano un numero ragionevolmente grande di pixels. Nel caso di esperimenti il cui obiettivo è quello di automatizzare la guida di veicoli è importante riconoscere se nella scena compare la vista posteriore di un veicolo [25] o un segnale stradale, così come nei casi in cui occorre leggere le targhe degli automezzi [26].

I “metodi con il gradiente” [22] indagano sulle relazioni tra i gradienti spaziali e temporali dell’intensità attraverso le “equazioni del flusso ottico”. Le loro prestazioni sono soddisfacenti in regioni ove vi sia sufficiente variazione spaziale dell’intensità e dove il moto tra i fotogrammi sia relativamente ridotto. Sfortunatamente nel caso del traffic–monitoring, gli oggetti presentano piccole variazioni di intensità le quali possono essere tanto piccole al punto da essere fuori dal campo di applicazione di questo tipo di approccio. I metodi basati sulle peculiarità estraggono dai fotogrammi punti di interesse (angoli, linee, punti per lo zero o regioni), per determinare una zona di moto analizzando il cambio di posizione delle peculiarità tra più fotogrammi. Anche qui, a causa del piccolo numero di pixel occupato da ciascun veicolo nell’immagine, i contorni e gli angoli dei livelli di grigio sono brevi e possono essere a malapena risolti.

Il terzo metodo, “RHT”, sviluppato da Kittler e Bober [22], si basa su <u>regioni ove si sviluppa il moto</u>; esso utilizza la combinazione della trasformata di Hough e dei “Robust statistical kernels”. E’ dimostrato che RHT può estrarre i parametri del moto da due fotogrammi (come lo spostamento o i parametri di trasformazioni affini) con stupefacente precisione. Inoltre la tecnica offre la possibilità di effettuare simultaneamente la segmentazione e la stima del moto ed è quindi computazionalmente efficiente. In una sistema tracking con RHT, per facilitare ulteriormente la segmentazione del moto e ridurre i carichi computazionali, si applicherà uno stadio di pre-processing di segmentazione a livelli di grigio. Esso rimuove le ambiguità derivanti dall’approccio basato sul moto delle regioni con livello di grigio monotòno e accelera la segmentazione del moto riducendo il numero dei passi.

In definitiva, l’algoritmo consiste in tre fasi: pre-processing (inclusa la segmentazione del moto a livelli di grigio), analisi del moto (comprendente una ulteriore segmentazione, se richiesta) e fase di post-processing dove si unificano le regioni con le stesse caratteristiche.

Il primo passo per l'analisi del moto tende a ridurre il numero di regioni per semplificare le operazioni seguenti. Questa semplificazione poggia sull'analisi delle differenze di immagini e permette di raccordare tra loro la maggioranza delle regioni statiche. La stima del moto è applicata a quelle. "L'estimatore di moto" usa un modello di moto traslativo poiché, anche se non perfettamente appropriato, è quello che garantisce il miglior risultato al minor costo computazionale.

Nella pre-segmentazione o 'segmentazione del frame di riferimento' otterremo gruppi di pixels con intensità simili per i quali è importante che, di regola, appartengano ad un singolo oggetto. Dal momento che l'asfalto e le carrozzerie delle auto sono generalmente a tinta unita, la segmentazione a livelli di grigio consiste in un raggruppamento di pixels appartenenti ad oggetti (auto) o allo sfondo (strada). L'algoritmo di classificazione (clustering) è ottenuto modificando quello di Kottle e Sun che a sua volta è stato ottenuto adattando quello classico "k-means" utilizzante uno spazio tridimensionale di features, le due immagini coordinate e l'intensità dei pixels: questi ultimi sono assegnati ad un cluster nel corso di un processo iterativo. Ad ogni iterazione un pixel i è assegnato a un cluster secondo il criterio di rendere minima una funzione [22].

Per evitare inutili carichi computazionali derivanti dalla stima del moto, è utile applicare l'algoritmo di stima solo alle regioni in movimento. Dopo una stima sommaria sul movimento delle regioni, quelle statiche sono ridotte ad un unico cluster (anche qui dettagliate spiegazioni sono fornite in merito).

La fase finale di post-processing raggruppa tra loro le regioni aventi moto coerente. Ciò è richiesto perché c'è da aspettarsi che l'algoritmo di segmentazione a livelli di grigio produca frazionamenti ridondanti allo scopo dell'analisi del moto. Se due regioni vicine risultano avere parametri di moto simili tra loro, esse sono probabilmente appartenenti allo stesso oggetto in movimento e saranno unificate. L'unificazione (merging) di regioni con moto coerente si replica finché non sia più possibile unirne nessuna ad altre.

Fatto questo, otteniamo una mappa di segmentazione 2D con grandi regioni di sfondo e piccole regioni in evidenza (auto). Una stima accurata delle traslazioni è fornita per ogni regione.

Riassumendo, lo schema è il seguente:

1) segmentazione a livelli di grigio;
2) merging delle regioni che non presentano differenze tra frames consecutivi;
3) merging delle regioni aventi moto tra loro coerente.

6 - Misurazione di occupazione del suolo con la FFT

Un ulteriore approccio all'analisi dei parametri del traffico riguarda la loro misurazione applicando la trasformata di Fourier alle immagini (Fast Fourier Transform).

Con questo sistema [2] è possibile elaborare in real-time un flusso di immagini. E' richiesto un processore 386 per ottenere una frequenza di 0.2 frames/sec su profilo da 512 pixels.

L'approccio descritto si propone di misurare i parametri delle code piuttosto che di rilevarne solamente la presenza, come normalmente richiesto ad implementazioni FFT. Esso è applicato ad un profilo che è a sua volta composto di sub-profili di larghezza variabile. Per compensare ogni effetto di spostamento di visuale fisica dell'immagine, la dimensione del sub-profilo varia in accordo con la distanza tra camera e geometria della camera. Il rilevamento del moto si basa su una tecnica di sottrazione applicata ai profili posti sui veicoli lungo la strada, mentre il rilevamento degli ultimi si basa sull'applicazione dell'edge-detection sui profili.

Gli algoritmi di image-processing sono spesso divisi tra quelli che operano nel dominio dello spazio o delle frequenze. Nel caso del traffico, per una strada vuota senza veicoli, ombre, né segni stradali, l'istogramma della strada contiene al più pochissimi valori di grigio alti e spostati verso l'origine. Quando la strada contiene oggetti, l'istogramma si sposta verso valori di grigio più alti. Questo effetto è stato sfruttato da diversi ricercatori per rilevare oggetti. Alternativamente l'ampiezza della FFT per una strada vuota contiene un'alta componente nell'origine (DC) e soltanto pochi valori alle alte frequenze. Questi ultimi sono dovuti al rumore dell'immagine e a quello elettrico. Quando la strada contiene oggetti, le componenti alle alte frequenze hanno ampiezze più grandi e, quando la strada contiene ulteriori oggetti, anche le componenti delle alte frequenze hanno ampiezze più alte. Queste componenti delle alte frequenze sono dovute alle varie tonalità di grigio e alle forme dei veicoli. Così lo spettro delle frequenze della superficie stradale può essere usato per la vehicle-detection e per la misurazione di altri parametri del traffico. Questo effetto può anche essere sfruttato per rilevare congestione e code, ed è stato introdotto in questa applicazione da Rourke e Bell [2].

I profili di scansione sono lunghi 512 pixels e la velocità di elaborazione dell'algoritmo è sufficiente per l'indagine in real-time dello stato di una coda di auto, dal momento che è difficile che questo cambi radicalmente in pochi secondi.

Il semplice algoritmo di sottrazione per l'indagine sul moto è applicato al profilo di due fotogrammi consecutivi, previo filtraggio per riduzione del rumore. Si studia l'istogramma della regione di interesse nel fotogramma comparandolo con una soglia per rilevare il moto. Per ridurre la quantità di dati e per eliminare il rumore dovuto alle piccole oscillazioni della telecamera, le regioni di interesse devono essere profili lungo la strada larghi al più tre pixel. Se estraiamo un profilo largo un pixel da ogni fotogramma, ad esso è prima applicato un filtro di media. Quindi si confronta la differenza tra due profili per rilevare un possibile moto. Quando c'è moto accade che la differenza tra i profili è maggiore rispetto al caso di assenza di moto. Perciò esso può essere rilevato semplicemente scegliendo una soglia. La dimensione dei profili per caratterizzare le code è un parametro importante, poiché potrebbe esserci moto in qualche sua parte e non essercene nelle rimanenti. Così si divide il profilo lungo la strada in un numero di sub-profili più piccoli di dimensioni variabili

Una delle caratteristiche interessanti risiede nella concatenazione delle due operazioni sequenziali: rilevamento del moto e rilevamento dei veicoli. Se il primo algoritmo non rileva il moto, il secondo effettua il controllo sulla presenza dei veicoli, per poi sondare nuovamente un eventuale moto o ricadere sull'indagine della presenza dei veicoli, se di questi non c'era traccia.

7 – Conclusioni

Da un giudizio sulle informazioni e sulla letteratura raccolte, risulta che buona parte delle problematiche inerenti l'applicazione della Visione Artificiale al traffic-monitoring è stata risolta con successo, anche se talvolta con approcci differenti e tra loro incompatibili.

Nonostante siano numerosi i campi ove è possibile introdurre miglioramenti, i sistemi di visione artificiale sembrano pronti per risolvere i problemi di traffic-monitoring. Le misure di velocità sono soddisfacenti, mentre quelle di conteggio sono più soggette al rumore tipico di questo tipo di riprese. La classificazione funziona discretamente fino a due/tre classi differenti ed in buone condizioni di illuminazione.

Tuttavia, nel materiale raccolto mancano spesso notizie sul carico computazionale delle soluzioni descritte e ciò induce in certi casi il sospetto che sia difficile raggiungere cicli di elaborazione con durata sufficientemente bassa, utili allo scopo di rilevare con efficacia il parametro di lunghezza di coda, rispetto ai cicli semaforici. Ancora, in molti casi non si affronta il problema di luce notturna (per non parlare di altre difficoltà di ripresa), limitando la verifica della soluzione a poche e selezionate sequenze di fotogrammi. Fortunatamente esistono articoli che si sono occupati del test e del confronto tra vari sistemi e dove emergono dati numerici sulle rispettive performance.

[26] Gli algoritmi variano in funzione dei modelli, sono influenzati dai vincoli fisici dell'ambiente di lavoro (illuminazione e contrasto, profili e tipi di veicoli, prospettiva, tipi di corsie...) e sfruttano molti tool differenti (matematici, statistici, cinematici, morfologici,...).

Ritengo che migliorare l'affidabilità del sistema rispetto alle condizioni di illuminazione della scena sia uno dei principali obiettivi da porsi.

7 – Bibliografia.

2 FATHY, M., SIYAL, M.Y.: ‘Real Time Image Processing Approach To Measure Traffic Queue Parameters’ IEE Proc. – *Vis. Image Signal Process.*, Vol. 142, No. 5, October 1995

3 DUBISSON JOLLY, M.P., LAKSHMANAN, S., K. JAIN, A.: ‘Vehicle Segmentation And Classification Using Deformable Templates’ *IEEE Trans. on Pattern Analysis And Machine Intelligence*, Vol. 18, No. 3, March 1996

5 AUBERT, D., BOUZAR, S., LENOIR, F., BLOSSEVILLE, J.M. ‘Automatic Vehicle Queue Measurement At Intersections Using, Image-Processing’, *Road Traffic Monitoring And Control*, 23-25 April 1996, Conf. Pubblication No. 422, IEE, 1996

6 SOTO, A., CIPRIANO, A.: ‘Image Processing Applied To Real Time Measurement Of Traffic Flow’ *0-8186-7352-4/96*, 1996 IEEE

7 MICHALOPULOS, P.G.: ‘Vehicle Detection Video Through Image Processing: The Autoscope System’ *IEEE Trans. On Vehicular Technology*, Vol. 40, No. 1, February 1991

10 CHATZIIONAU, A., HOCKDAY, S.L.M., KAIGHN, S., PONCE., L.: ‘Video Image Processing Systems: Applications In Transportation’ *0-7803-2587-7/95*, 1995 IEEE

12 PARODI, P., PICCOLI, G.: ‘A Feature-Based Recognition Scheme For Traffic Scenes’ *no reference on copy*

22 BADENAS, J., BOBER, M., PLA, F.: ‘Motion And Intensity-Based Segmentation And Its Application To Traffic Monitoring’ *no reference on copy*

25 MATTEWS, N.D., AN, P.E., CHARNLEY, D., HARRIS, C.J.: ‘Vehicle Detection And Recognition In Greyscale Imagery’ *Control Eng. Practice*, Vol. 4, pp. 473-479, 1996 Elsevier Science Ltd

26 BRIQUET, P.: ‘Video Processing Applied To Road And Urban Traffic Monitoring’ *Sixth International Conference On Road Traffic Monitoring And Control* Conf. Publ. No. 335, pp. 153-157, IEEE 1992

33 MONTGOMERY, F.O., QUINN, D.J.: ‘Queue Management Strategies Using Fixed Time Signal Control *no reference on copy*