



Fostering the Emerging Market of Internet of Things

P2 – Valorització de des dades de la IoT

E6.1 – Mòdul de detecció de vianants i vehicles lleugers



UNIVERSITAT POLITÈCNICA
DE CATALUNYA
BARCELONATECH



LEITAT
managing technologies



IREC
Institut de Recerca en Energia de Catalunya
Catalonia Institute for Energy Research

URV
UNIVERSITAT ROVIRA I VIRGILI



eurecat
Centre Tecnològic de Catalunya



Document	Títol: E6.1 Mòdul de detecció de vianants i vehicles lleugers	
	Disponible a:	
	Paquet de treball: WP6 – Desenvolupament dels mòduls d'anàlisi de dades del <i>use case</i> de mobilitat.	
	Nivell de disseminació: Públic	
Autoria	Escrit per:	<i>Enric Sala (CVC), Sergio Escalera (CVC), Meysam Madadi (CVC), Julio Jaques Jr (CVC)</i>
	Revisat per:	

Resum executiu

En aquest document s'explica resumidament la implementació que s'ha fet d'un mòdul d'anàlisi de vídeos de passos de vianants, així com el seu ús.

En la tasca 6.1 s'ha empaquetat un algorisme que rep vídeos de passos de vianants i automàticament retorna estadístiques com el número de persones i bicicletes que ha creuat en cada direcció, valors del temps que han emprat a creuar i el temps que han hagut d'esperar.

Els algorismes s'han empaquetat en una imatge de *Docker* que sha desplegat a la plataforma CVC Webservices.

Taula de continguts

1	Introducció	3
2	Descripció Tècnica	4
1.	Extracció de la Informació Intrínseca	4
2.	Detecció dels Passos de Vianants	4
3.	Detecció de persones i bicicletes	5
4.	<i>Tracking</i> de les deteccions	6
5.	Localització de les zones d'espera del pas de vianants	6
6.	Anàlisi de les trajectòries i obtenció de les estadístiques	7
3	Desplegament del mòdul	8
4	Codi Font i Docker	10
5	Conclusions	10

1 Introducció

En aquest document s'explicaran els desenvolupaments que s'ha fet a la tasca 6.1 per implementar el mòdul de detecció de vianants i vehicles lleugers de la plataforma de valorització de dades de la IoT.

Concretament, el mòdul de detecció de vianants i vehicles lleugers s'emmarca en el context del cas d'ús 2: "Sistema intel·ligent de control de mobilitat i intensitat del trànsit". Aquest mòdul pretén valoritzar i treure informació interessant de vídeos de passos de vianants. El mòdul rep un vídeo (estàtic i de qualsevol durada) on apareix un o diversos passos de vianants, i retorna estadístiques d'ús de cada pas de vianants. Entre les dades que retorna hi trobem:

- Número de persones que han creuat en direcció A i en direcció B.
- Temps mitjà, màxim i mínim utilitzat per les persones que han creuat en direcció A i en direcció B.
- Temps mitjà, màxim i mínim utilitzat per les persones que han esperat per creuar en direcció A i en direcció B.
- Número de bicicletes que han creuat en direcció A i en direcció B.

Aquest conjunt d'estadístiques obtingudes sobre un pas de vianants permeten obtenir automàticament un valor objectiu sobre el bon o mal funcionament d'un pas de vianants i extreure'n conclusions com les que segueixen. Si a partir d'aquest anàlisi s'observa que el temps mitjà o el temps màxim utilitzat per creuar en una determinada direcció es molt curt, pot ser un indicatiu de què els períodes de temps assignats als semàfors no estan ben ajustats, i la gent ha de creuar corrents. Igualment, si el temps d'espera als semàfors es molt gran, també pot ser un indicatiu d'una mala assignació dels períodes de temps per a cada etapa semafòrica. Un anàlisi horària del mateix pas de vianants, ens pot donar una orientació molt real de a quines hores el pas de vianants és més concorregut i per tant quan pot ser més òptim assignar una patrulla policial per vigilar-ne el bon funcionament.

Es important remarcar que el mòdul que s'ha desenvolupat no necessita cap més entrada que el vídeo. Els algorismes empaquetats automàticament detecten les zones dels passos de vianants, els vianants, les bicicletes i les zones considerades d'espera per a creuar. El mòdul utilitza tecnologies de *Deep Learning* (Xarxes Convolucionals Profundes) i requereix d'un hardware específic per executar-se, concretament una targeta GPU (*Graphical Processing Unit*). Aquesta restricció de hardware es la que ha fet que l'algorisme empaquetat s'hagi desplegat a la plataforma **CVC Webservices**. Aquesta plataforma ofereix un *webservice* que pot ser cridat des de qualsevol interfície web. El *webservice* rep com a entrada el vídeo a processar, i retorna un fitxer amb estructura JSON on hi ha totes les estadístiques explicades, així com les deteccions que han sigut d'interès, per tal que es puguin mostrar a sobre el vídeo d'entrada, en *el front-end*. A part del desplegament al servidor, també s'ha creat un script que permet executar

el mòdul en model de proves en un sistema Linux proveït d'una GPU i que tingui instal·lada la plataforma *Docker*.

2 Descripció Tècnica

El mòdul de detecció de vianants i vehicles lleugers que s'ha desenvolupat per a la plataforma de valorització de dades de la IoT s'ha desenvolupat en el llenguatge *Python*, creant un mòdul tancat que segueix l'estructura requerida per la plataforma **CVC Webservices**. No obstant, també s'ha afegit un script per poder executar el mòdul directament amb un vídeo, de manera local i fora de la plataforma.

El processament que es realitza a cada vídeo consta de 5 etapes:

- 1- Extracció de la informació intrínseca del vídeo (dimensions del vídeo, imatges per segon, durada).
- 2- Detecció de les zones on hi ha un pas de vianants.
- 3- Detecció de persones i bicicletes que s'acosten i/o creuen el pas de vianants.
- 4- Seguiment de les persones i bicicletes detectades (és a dir associar cada detecció a una persona o bicicleta); dit tècnicament seria fer un *tracking* de les deteccions, que ens donaria les trajectòries de cada persona o bicicleta.
- 5- Localització de les zones d'espera del pas de vianants, a partir de les trajectòries obtingudes.
- 6- Anàlisi del comportament de les persones i bicicletes a partir de les trajectòries obtingudes, i obtenció de les estadístiques.

A continuació s'expliquen els detalls tècnics i d'implementació de cada fase.

1. Extracció de la Informació Intrínseca

En aquesta fase s'utilitza la llibreria *OpenCV*, concretament la seva versió per a *Python*, per a obtenir la mida, la durada i els *frames* (imatges) per segon del vídeo. Aquesta última informació es crítica ja que en el procés d'obtenció de les estadístiques, aquestes es calcularan per número de *frames*, i faltará aquest factor per poder-les passar a segons. La llibreria d'*OpenCV* també s'utilitzarà en les successives etapes per obtenir cadascun dels *frames* del vídeo.

2. Detecció dels Passos de Vianants

El primer que cal fer per poder processar correctament el vídeo es veure quines son les seves zones d'interès. En aquest cas està clar que l'interès està a l'entorn del pas de vianants per tant cal, primer de tot, detectar els passos de vianants. Per a la detecció dels passos de vianants s'ha entrenat una Xarxa Convolutiva Profunda de tipus YoloV3, en l'entorn *Tensorflow*, que s'ha especialitzat en la detecció de passos de vianants. La xarxa rep com a entrada un *frame* del vídeo i retorna les coordenades d'un rectangle que delimita la zona del pas de vianants, per cadascun dels passos de vianants detectats.

Tot i que el nivell de fiabilitat de la xarxa es molt alt, i els vídeos son estàtics (sense moviment), s'aplica la detecció dels passos de vianants a tots els *frames* del vídeo, post processant després aquestes deteccions per tal d'eliminar les que apareixen només a molt pocs *frames*, i quedant-nos amb les que apareixen a la majoria. D'aquesta manera evitem que una detecció errònia en un/s *frame*/s concrets ens perjudiqui l'obtenció de totes les estadístiques.

A la Figura 1 es poden veure alguns exemples de deteccions de passos de vianants obtingudes als vídeos d'exemple.



Figura 1: Exemples de deteccions de passos de vianants.

3. Detecció de persones i bicicletes

La part més crítica de l'algorisme empaquetat es la que fa referència a la detecció de les persones i les bicicletes presents a la imatge. Sense una detecció molt fiable de les persones i bicicletes a cada *frame* de la imatge, seria impossible calcular qualsevol estadística sobre els usos del pas de vianants. Per aquesta tasca tant crítica s'ha entrenat una vegada més una Xarxa Convolutiva Profunda de tipus Yolov3 en l'entorn *Tensorflow*, que s'ha especialitzat en la detecció de persones i bicicletes. La xarxa rep com a entrada una imatge, i retorna les coordenades d'un rectangle que delimita l'espai que ocupa cada persona o bicicleta detectades. Cadascun dels *frames* del vídeo son processats per les xarxes, i cada detecció es emmagatzemada per ser posteriorment post-processada.

A la Figura 2 podem veure les deteccions de vianants (en vermell) i de bicicletes (en taronja) que ha realitzat la xarxa en *frames* concrets dels vídeos de mostra.



Figura 2: Exemples de deteccions de vianants (vermell) i bicicletes (taronja).

4. *Tracking de les deteccions*

Una vegada obtingudes les deteccions de persones i bicicletes a cada *frame* del vídeo, es necessari identificar cada detecció amb un objecte concret. És a dir que cal associar la caixa A obtinguda en el *frame* 1 amb la caixa B obtinguda en el *frame* 2, determinant que son la mateixa persona que s'ha mogut uns pocs píxels. Aquest pas, que ens permet passar de la detecció dels objectes, a la identificació de les seves trajectòries, l'hem fet utilitzant la llibreria *norfair* per *Python*, especialitzada en associar deteccions provinents de xarxes de detecció per tal d'obtenir trajectòries d'objectes. Aquesta llibreria assigna un identificació d'objecte a cada detecció. La llibreria té funcions per visualitzar les deteccions amb un número identificador d'objecte, que permet veure en el vídeo, com l'objecte es va movent. En les imatges de la Figura 2 es pot veure com cada detecció té imprès un numeret groc a la part inferior. Aquest numeret es l'identificador de l'objecte, que es va moment, seguint la trajectòria de l'objecte/persona al llarg del vídeo.

5. *Localització de les zones d'espera del pas de vianants*

Per tal d'obtenir les estadístiques de temps d'espera per creuar els passos de vianants, cal delimitar les zones que es consideren com a zones d'espera per creuar. Aquestes zones normalment seran als 2 extrems laterals de les zones delimitades com a pas de vianants. No obstant, a la primera imatge de la Figura 1 podem veure que no sempre la zona delimitada pel detector de passos de vianants coincideix perfectament amb el pas de vianants, ja que el detector retorna les coordenades d'un rectangle, i no

sempre el pas de vianants està en una posició perfectament horitzontal o vertical. A més, el detector de passos de vianants no retorna en quina de les direccions creuen els vianants i els vehicles. Per tots aquests motius, a l'hora d'establir les zones d'espera per creuar s'utilitzen les trajectòries de les deteccions de persones obtingudes, i el punt on aquestes trajectòries entren a la zona delimitada com a pas de vianants. La mida de la zona d'espera del pas de vianants, així com la distància a la que han d'estar les deteccions que es consideren properes, es calculen a partir de la mida de totes les deteccions de vianants; d'aquesta manera es té en compte en aquestes mesures, la distància entre el pas de vianants i la càmera.

A la Figura 3 podem veure les circumferències de color taronja que delimiten les àrees d'espera per creuar. També es poden apreciar, en diversos colors, les trajectòries de vianants utilitzades per calcular aquestes zones. No cal dir que la part de la circumferència que interseca amb el pas de vianants no es considera zona d'espera.

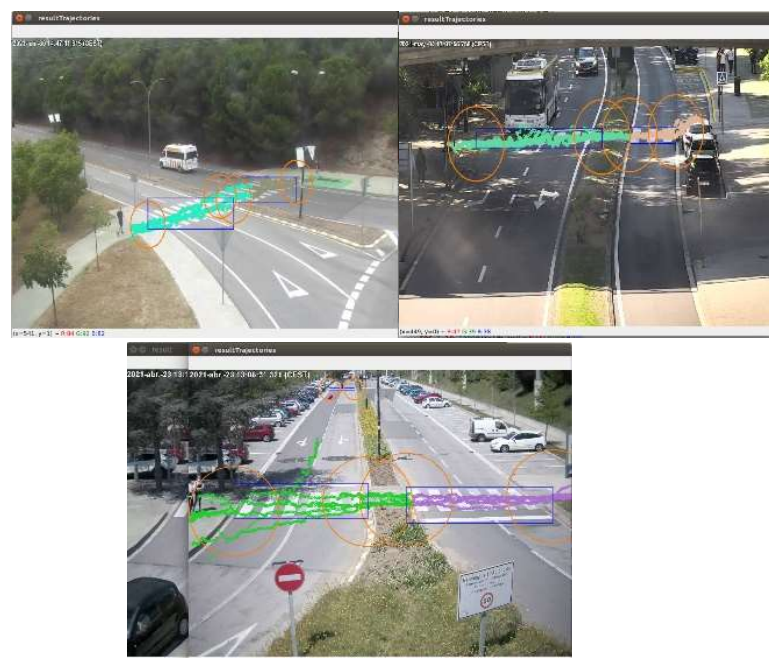


Figura 3: Exemples de càlcul de les zones d'espera del pas de vianants.

6. Anàlisi de les trajectòries i obtenció de les estadístiques

Una vegada es tenen delimitades les zones on hi ha els passos de vianants, les zones d'espera per creuar el pas de vianants i les trajectòries de totes les persones i bicicletes detectades, no es fa difícil processar aquestes dades per tal d'obtenir les estadístiques especificades. Processant totes les trajectòries obtingudes, s'ha creat una llista on, per cada pas de vianants, es guarda el primer i últim *frame* de cada trajectòria, així com les posicions inicials i finals. El mateix tipus de llista s'ha creat amb les zones d'espera per creuar el pas de vianants. Amb aquesta llista es fa senzill obtenir durant quants *frames* una persona està creuant o esperant per creuar. I amb la informació dels *frames* per segon del vídeo, fàcilment podem convertir aquesta dada en una dada de

temps. Una vegada sabem quant de temps ha trigat cada persona a creuar i a esperar, es senzill calcular estadístiques com el temps mig, màxim o mínim, així com el número de persones que han creuat.

3 Desplegament del mòdul

Degut a les necessitats de ser executat en un hardware específic, aquest mòdul s'ha hagut de desplegar a la plataforma *CVC Webservices*, que compta amb targetes GPU per a realitzar els processaments. No obstant i per mantenir la portabilitat, tot el codi, l'entorn i les llibreries utilitzades s'han dockeritzat en un fitxer Dockerfile.

A falta del desenvolupament del *front-end* final, que es desenvoluparà al WP7, el mòdul es pot testear amb la pròpia interfície del *CVC Webservices*, accedint a la url: <https://api.cvc.uab.es/webservice>

A la Figura 4 es pot veure la interfície actual del CVC Webservices, que permet pujar un vídeo i esperar que en retorni el resultat.

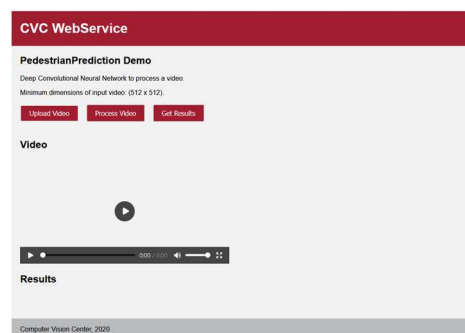


Figura 4: Interfície del CVCWebseices per demanar el processament de vídeos.

Una vegada el vídeo s'ha processat, i prement el botó "*Get Results*", el servidor retorna la resposta en format JSON. En la resposta en format JSON es poden veure les estadístiques per cada pas de vianants. A més a més, en la resposta en format JSON també s'hi inclouen totes les deteccions, per si el *front-end* que faci la crida al *CVCWebservices* vol mostrar, a sobre del vídeo original, totes les deteccions.

El resultat del processament del vídeo es una estructura JSON com la que es pot veure a continuació. Per evitar que aquest exemple tingués una longitud excessiva, s'han deixat només dues deteccions per cada pas de vianants, tot i que en la resposta original hi son totes.

```
"VideoInformation": {
  "Duration": 29.12,
  "Fps": 25,
  "NZebraCrossingAreas": 2,
  "Width": 1920,
  "height": 1080
},
"ZebraCrossingAreasResults": [
  {
    "AvgTimeCrossingDirA": 4.74,
    "AvgTimeCrossingDirB": 0,
    "AvgTimeWaitingAreaA": 1.04,
```

```

"AvgTimeWaitingAreaB": 1.4480000000000002,
"Detections": [
  [
    1,
    82.80050210270396,
    500.8554244543735,
    154
  ],
  [
    1,
    88.10774869433938,
    502.3295621835399,
    155
  ]
],
"MaxTimeCrossingDirA": 5.08,
"MaxTimeCrossingDirB": 0,
"MaxTimeWaitingAreaA": 2,
"MaxTimeWaitingAreaB": 2.44,
"MinTimeCrossingDirA": 4.4,
"MinTimeCrossingDirB": 0,
"MinTimeWaitingAreaA": 0.08,
"MinTimeWaitingAreaB": 1.4480000000000002,
"NByciclesCrossingDirA": 0,
"NByciclesCrossingDirB": 0,
"NPersonsCrossingDirA": 5,
"NPersonsCrossingDirB": 0,
"ZebraCrossingAreal": 0
},
{
  "AvgTimeCrossingDirA": 5.76,
  "AvgTimeCrossingDirB": 0,
  "AvgTimeWaitingAreaA": 1.28,
  "AvgTimeWaitingAreaB": 1.66,
  "Detections": [
    [
      2,
      906.7215325508414,
      499.2264822523778,
      311
    ],
    [
      2,
      907.622706915689,
      498.4622901920369,
      312
    ]
  ],
  "MaxTimeCrossingDirA": 5.76,
  "MaxTimeCrossingDirB": 0,
  "MaxTimeWaitingAreaA": 1.28,
  "MaxTimeWaitingAreaB": 3.24,
  "MinTimeCrossingDirA": 5.76,
  "MinTimeCrossingDirB": 0,
  "MinTimeWaitingAreaA": 1.28,
  "MinTimeWaitingAreaB": 1.66,
  "NByciclesCrossingDirA": 0,
  "NByciclesCrossingDirB": 0,
  "NPersonsCrossingDirA": 2,
  "NPersonsCrossingDirB": 0,
  "ZebraCrossingAreal": 1
},
],
"status": "done"
}

```

A part de la implementació realitzada per a ser desplegada a la plataforma CVC Webservices, també s'ha desenvolupat un *script* que permet testear el codi de manera aïllada en un sistema Linux, sempre que disposi d'un hardware de GPU. En el següent apartat s'explica com tenir-hi accés.

4 Codi Font i Docker

Tot el codi font utilitzat en el projecte, així com el fitxer de Docker que crea tot l'entorn on executar el projecte, s'ha pujat al repositori de Github accessible des de:

<https://github.com/FEM-IoT/CVC>

El fitxer README.md del repositori explica com desplegar el mòdul en un sistema Linux proveït d'una GPU, així com la manera d'executar el mòdul de manera local per a processar un vídeo concret.

5 Conclusions

- En el projecte P2 per a la creació d'una plataforma de valorització de dades de la IoT i més concretament en el WP6 de desenvolupament del cas d'ús de mobilitat, s'ha portat a terme la tasca T6.1 anomenada "Desenvolupament d'un mòdul de detecció de vianants i vehicles lleugers".
- Aquest mòdul ha de permetre obtenir informació d'interès i per tant de molt valor, de les imatges de passos de vianants obtingudes amb càmeres instal·lades a l'espai públic.
- S'ha creat un mòdul que només rebent un vídeo estàtic on apareguin un o diversos passos de vianants, i sense cap interacció manual, retorna estadístiques objectives que permeten avaluar el bon o mal funcionament del pas de vianants.
- Les estadístiques retornades indiquen el número de persones i/o bicicletes que ha creuat en cada direcció, el temps que han trigat a fer-ho, i el temps que han hagut d'esperar per creuar.
- El mòdul s'ha desenvolupat amb el llenguatge Python, utilitzant el *framework* de TensorFlow, i tècnicament està molt recolzat en l'ús de Xarxes Convolucionals Profundes (*Deep Neural Networks*).
- El mòdul s'ha desplegat a la plataforma CVC *Webservices* ja que per a ser executat requereix de l'ús de Hardware específic dedicat, concretament targetes GPU (*Graphic Processing Unit*).
- La plataforma CVC *Webservices* es una de les tres plataformes (juntament amb DATURA i EMMA) que conformen la plataforma de valorització de dades del projecte P2.
- L'entorn d'execució del mòdul s'ha Dockeritzat per tal de poder executar fàcilment el mòdul en noves instal·lacions.
- A part del desenvolupament realitzat per ser desplegat a la plataforma CVC *Webservices*, s'ha creat un script per tal de poder executar individualment i de manera local un vídeo concret.
- Tot el codi font, així com el fitxer de Docker, s'ha pujat i deixat disponible a un repositori de Github, per tal que sigui públic i accessible per a tothom.