

Képfeldolgozás alapú lefelismerő rendszer

Által Ádám

2024. május 8.

Tartalomjegyzék

1. Bevezetés	3
2. Irodalomkutatás	5
2.1. A játékosok felismerése	8
2.2. A labda felismerése	9
2.3. A les megállapítása	10
3. Tervezés	12
3.1. Programnyelv	12
3.2. Specifikáció	12
3.2.1. A képek feldolgozása	12
3.2.2. Az objektumok klasszifikációja	13
3.2.3. A les megállapítása	13
3.2.4. Felhasznált bemenetek	14
3.2.5. Kimenet	14

1. Bevezetés

Modern világunkban napról napra egyre nagyobb szerepe jut a sportinformatikának. Ez a folyamat megfigyelhető a labdarúgás területén is (gondoljunk csak a videóbíró pár éve történt bevezetésére), azonban ennek a szakterületnek még bőven van hova fejlődni. A bírói döntések közül az egyik legmegosztóbb a leshelyzetek meghatározása, melyet ma már a videóbíró is segít, azonban a játékosok pozícióját kézzel rajzolják be, ezáltal fennáll a hiba lehetősége. Dolgozatom célja ezen "emberi tényező" eltávolítása lesz a játékosok pozíciójának, és ezáltal a leshelyzet meghatározásának megállapításával.

De hogy is van pontosan ez a lesszabály? Az IFAB¹ hivatalos szabálykönyve[1] alapján 2021 szeptemberében a lesszabály a következőként fogalmazható meg: a játékos leshelyzetben van, ha a labda elrúgásának pillanatában:

- a labdánál közelebb tartózkodik az gólvonalhoz
- a fej, test vagy láb bármely része az ellenfél térfelén van és
- a fej, test vagy láb bármely része közelebb van az ellenfél gólvonalához, mint az ahhoz második legközelebbi játékos (itt illik megjegyezni, hogy az első legközelebbi játékos legtöbbször a kapus)

A játékosok kezeit és karjait nem vesszük figyelembe a les megállapításánál, illetve nincs lesen egy játékos, ha egy vonalban van az utolsó előtti, vagy a két utolsó játékosal.

A fentiek alapján tehát egy olyan szoftver elkészítése a cél, ami egyrészt egy képről meg tudja állapítani az azon szereplő játékosok pozícióját, másrészt a pozíciók alapján el tudja dönteni azt, hogy ki az aki leshelyzetben tartózkodik. Ehhez a következőkben megvizsgálók néhány olyan, már létező képfeldolgozást felhasználó megvalósítást(2. fejezet2), ami képes lehet felismerni és külön választani a bevitelről számomra szükséges területeket, illetve egy olyan algoritmust, mellyel a már jól behatárolt labdarúgók pozícióiból megállapítható

¹International Football Association Board, a Nemzetközi Labdarúgó-szövetség (FIFA) Szabályalkotó Testülete

hogy ki tartózkodik leshelyzetben, majd ezek közül kiválasztom és részletesebben leírom(3. fejezet3) az általam leghatékonyabbnak véltet.

2. Irodalomkutatás

A következőekben áttekintem néhány, a témával foglalkozó projekt megvalósítását, módszereit. Ezt követően megvizsgálom a feladat megvalósítása során felmerülő problémákat, kihívásokat, ezekre megpróbálok megoldást vagy alternatívát találni, majd kiválasztom az általam felhasználandó módszereket.

2021 nyarán a FIFA a honlapján beszámolt arról, hogy videókonferenciát tartott a korábban általuk, a technológiai fejlesztések céljából létrehozott Working Group of Innovation Excellence-szel (magyarul nagyjából Innovációs Munkacsoportnak lehetne hívni), mely során bejelentették, hogy a videóbíró bevezetésének sikere után szeretnék továbbfejleszteni a rendszert egy félautomata lesfelismerő funkcióval. Ennek célja elsősorban nem a leshelyzet megállapítása lenne, hanem egy eszköz biztosítása a játékevezetők számára, mely segítségével könnyebben meghozhatják a döntésüket - a már használatban lévő gólvonal technológiához hasonló módon. Ezen konferencia során a felmerülő kihívások között felmerült a labda elrúgásának pillanata (annak pontos megállapítása), illetve a testrészek helyes felismerése, ugyanis még kihívást jelent a játékosok "csontvázának" modellezése. A konkrét implementációról mindössze ennyi információt lehet tudni egyelőre, a tervek szerint a 2022/23-as szezonban szeretnék bevezetni a rendszert.[2]

2009-ben egy olasz csapat írt egy elég átfogó cikket a témában.[3] Ebben egy valós idejű lesdetektáló rendszer megvalósíthatóságát vizsgálták. Három-három nagy felbontású kamerát helyeztek el mindkét oldalvonal mentén úgy, hogy az optikai tengelyek a gólvonallal párhuzamosak legyenek a perspektívahibák elkerülése érdekében. Ezek felvételeit szinkronizálták, és párhuzamosan feldolgozták, működésüket pedig egy hetedik, központi "felügyelő" csomópont szinkronizálta újra, illetve meghozta a végső döntést. A felvételek feldolgozásának első lépése a háttér eltávolítása volt. Mivel mozgóképpel dolgoztak, ehhez egy mozgásérzékelő algoritmust használtak, amely a legalább 2,5 másodpercig mozdulatlan pontokat feltételezte háttérnek. Ezek után geometriai módszerekkel kivonták a játékosok árnyékait is. A szegmentáció után szükség volt arra, hogy minden elkülönített objektumhoz hozzárendeljék

hogy hova tartozik, mindezt emberi beavatkozás nélkül. Ehhez először egy clusterező algoritmussal és véletlenszerűen választott bemenetekkel csoportokat hoztak létre, majd futásidőben minden szegmentált objektumot hozzárendeltek a létrehozott clusterek egyikéhez. Ezen klasszifikáció után a játékosokat megfeleltették egy "határdobozzal", ami segítségével követni tudták a mozgásukat. A labda felismeréséhez is a mozgó objektumokat használták, azonban először méret alapján kiszűrték a lehetséges pozícióit (a mérkőzés során később, amikor már rendelkezésre állnak előzetes információk azokat felhasználva), majd egy mintafelismerő algoritmussal kiválasztották ezek közül a labda tényleges helyzetét. Több egymást követő kép segítségével a labda sebesség- és mozgásának irányváltozásait megvizsgálva a lövés pillanatát is meg tudták állapítani. Ezzel minden rendelkezésre áll a leshelyzet megállapításának döntéséhez. A fent írtak alapján tehát a csomópontok által feldolgozott adatok egy központi csomóponton kerülnek szinkronizálásra, és itt fog megszületni az eredmény is. A szinkronizáció során a felügyelő csomópont egy virtuális pálya segítségével egyesíti a kapott adatokat, a nagyon eltérő bemeneteket átugorva, vagy a megbízhatóbbnak tűnőket validálva. A rendszer egyaránt tud azonnal jelezni leshelyzet esetén - ekkor csak azon képek kerülnek vizsgálatra, amelyeken hirtelen változás történt a labda mozgásában; illetve meg tudja különböztetni hogy aktív vagy passzív leshelyzetről van szó, egy 3 másodperces ablak során megvizsgálva hogy a lesen lévő játékos játékba tud(na)-e avatkozni.

Egy 2013-as tanulmányban a teheráni Payame Noor Egyetem két munkatársa is foglalkozott a témával.[4] Ebben különösen nagy hangsúlyt fektettek a kamerafelvétel perspektivikus hibáinak kiküszöbölésére. A háttér felismerését Kálmán-szűrő segítségével végezték el, majd felismerték az objektumokat, elkülönítve a játékosokat, játékvezetőt és a labdát. Egy algoritmussal felismerték és megjelölték a leshelyzetben lévő játékosokat, majd megvizsgálva a térfelet, a passzot játszó játékost, és a passz irányát megállapították hogy lesen van-e az adott játékos.

Egy másik, Indiában elkészített megvalósítás a játékosok kontúrjainak



1. ábra. Kék színű pont jelzi a támadó és a védő legközelebbi pontját

a felismerésével oldotta meg a feladatot.[5] A megoldás során a csapatok mezeinek színének előzetes ismeretével éldetektálást hajtanak végre a képen, Gauss-szűrővel csökkentve a zajos/hibás élek számát. Ezt követően a játékosok alapvonalhoz legközelebbi pontjait összehasonlítva megszületik a végső döntés.

2.1. A játékosok felismerése

Az első és talán legfontosabb kérdés maguknak a játékosoknak a felismerése. Ezt strukturálisan három részre lehet osztani: szegmentálás, követés és klasszifikáció. Ezek mind kulcsszerepet játszanak egy működő algoritmus felépítésében.

A mozgó objektumok detektálásának három hagyományos módszere van: *temporal differencing*, háttér kivonás (*background subtraction*), optikai áramlás (*optical flow*).[6]

	háttér kivonás	temporal differencing	optical flow
előnyök	relatív statikus hátterekhez kiváló	alkalmazkodik a dinamikus hátterekhez; könnyű implementálni	hatékony dinamikus háttereknel
hátrányok	különböző statisztikai modellezési feltevések közül kell választani	rosszul működik egész mozgó objektumok esetén; nagyon zajérzékeny	Számításilag összetett; erősen zajérzékeny
komplexitás	közepes	egyszerű	bonyolult

1. táblázat. A mozgásfelismerő módszerek összehasonlítása[7]

Tekintve, hogy a majd felhasználásra kerülő felvételek háttere (a futballpálya) nagyban statikus, illetve mivel gyakran a felismerni kívánt játékosok egésze mozog, és túlzottan bonyolult algoritmust sem szeretnénk használni a háttér kivonása tűnik a kézenfekvő módszernek.

A klasszifikáció is többféle módon történhet. Kézenfekvő megoldásnak tűnik a játékosok mezeinek színét felhasználva elkülöníteni őket. Felmerül azonban az a kérdés, hogy a vizsgált bemenet elején manuálisan állítsuk-e be, hogy melyik szín melyik csapatot reprezentálja, vagy egy felügyeletet nem igénylő (*unsupervised*) algoritmust használunk, mely automatikusan állapítja

meg a felismert játékosok színét. Amennyiben utóbbi mellett döntünk, a [8]-ban háromféle klaszterező algoritmus merül fel: MBSAS (*Modified Basic Sequential Algorithmic Scheme*, módosított alapvető szekvenciális algoritmikus séma): melynek során az objektumokat reprezentáló vektorokat hozzárendeljük egy már létező klaszterhez, vagy egy újat hozunk létre erre a célra, mindezt a már létrehozott klaszterektől való távolságtól függően; BCLS (*Balanced Clustering with Least Square regression*, kiegyensúlyozott klaszterezés legkisebb négyzetek módszerével): ez egy kompetitív algoritmus, amelyben a csoportokat véletlenszerűen inicializáljuk, majd valamilyen összehasonlítás alapján kiválasztjuk a győztes klasztert, melyet frissítünk, végezetül pedig hozzárendeljük az objektumokat a klaszterekhez; Isodata (vagy k-means): a csoportokat szintén véletlenszerűen inicializáljuk, majd addig prezentáljuk a vektorokat, amíg a klaszterek változnak, minden egyes prezentálás során a klasztereket a bemutatott vektorokkal való eltérés függvényében frissítjük.

2.2. A labda felismerése

Leshelyzet megállapításához kulcsfontosságú tudni a labda pontos pozícióját is. A labdafelismerés egy elég gyakran vizsgált feladat, számtalan ilyen projekt született az évek során, ezeket két kategóriába lehet sorolni: közvetlen (direkt) vagy közvetett (indirekt) módszerek.[9]

A közvetett módszerek valamilyen, a labda mozgásával kapcsolatos előzetes tudásra alapozva különböztetik meg a labdát a többi objektumtól. A [10]-ban a labda pályájának vizsgálatával állapítják meg annak pozícióját több "labda-szerű" jelölt közül. A [11]-ban a labda pályáját síkgörbékként modellezik. Ezeket a síkokat görbületi küszöböléssel és 3D vonalaktól vett legkisebb távolság segítségével határozzák meg. Ezek a módszerek jól alkalmazhatóak videóindexeléshez, azonban valós idejű események észlelésére kevésbé, ugyanis nem képesek a labda helyzetét minden képkockán kiértékelni.[9]

A közvetlen módszerek ezzel ellentétben minden képkockán képesek megállapítani a labda helyzetét annak formája, mérete illetve színe alapján. A [12]-ban

Support Vector Machine (SVM) módszerrel klasszifikálták a labdát. Ez jól látható labda esetén igen jól működött (98,3% detektálási, 0,2% fals pozitív arány), azonban a nehezen látható labdák esetén ez jelentősen gyengébb volt (76,2% detektálási, 2,6% fals pozitív arány). A [13]-ban hullám dekompozíció és független komponensanalízis (*Independent Component Analysis*, ICA) segítségével a nehezen látható labda esetén jobb eredményeket értek el (92,08% és 87,06% detektálási arány). A [9]-ban először kör Hough transzformációt alkalmaznak a mozgó objektumokra a képen, hogy megtalálják a leginkább kör alakút. Ezután egy skálainvariáns jellemző transzformáció (*Scale-invariant feature transformation*, SIFT) használatával megvizsgálják, hogy a felismert objektum valóban labda-e. Ezen módszerrel leegyszerűsíthető a betanítás folyamata. Az eredmények detektálás szempontjából a két másik vizsgált módszer között (83.08-90.3%) helyezkedett el, azonban nagyon nagy volt a fals pozitív eredmények előfordulása (9.7 és 16.91%).

2.3. A les megállapítása

A program utolsó lépése a leshelyzet tényének megállapítása lesz. Ennek első lépéseként be fogom tanítani a programot valamilyen egyszerű szimulációs környezetben a leshelyzet megállapítására. Ezt követően a bemeneti felvételekből szerzett információkat (a játékosok és a labda pozícióit) a program leképezi a kétdimenziós környezetre, majd a betanult módon megállapítja, hogy fennáll-e a leshelyzet. Érdekes lehet mindezek előtt megállapítani, hogy mikor volt a labda elrúgásának pontos pillanata, melyik játékos volt, aki megjátszotta azt, illetve, hogy aktív vagy passzív leshelyzet áll fenn, azonban ezek kifejezetten nehéz feladatnak bizonyulnak. Az elrúgás pillanata megállapítható a labda mozgásának, egészen pontosan a sebességének vizsgálatával. Amennyiben a labda hirtelen az eddigieknél nagyobb gyorsulást mutat, jó eséllyel meg lett játszva. Amennyiben sikerül ezt a pillanatot megállapítani, azzal a labdát elrúgó játékost is közel 100%-os pontossággal meg tudjuk állapítani a labda és a játékosok távolságának vizsgálatával, ugyanis a labdához szinte

majdnem minden esetben a labdát megjátszó játékos lesz a legközelebb. Az aktív/passzív leshelyzetek megkülönböztetése ennél jóval bonyolultabb, egy másik játékos "zavarását" nem nagyon lehet gépi környezetben megállapítani, talán a legnagyobb pontossága annak lenne, ha szintén a játékosok pozícióját vizsgálnánk, ezen esetben egyaránt a védekező játékosokhoz és az elrúgott labdához képest. A projektem során ezt is meg fogom vizsgálni.

3. Tervezés

3.1. Programnyelv

Egy informatikai projekt talán legfontosabb döntése, hogy milyen programozási nyelven fogjuk írni programunkat. Ebben a projektben a Python open-source programozási nyelvet fogom használni annak nagy népszerűségéből fakadó támogatottsága és jó modularitása (rengeteg használható könyvtár) miatt. Amennyiben szükséges, a kódot helyenként CUDA C nyelv használatával lehetne gyorsítani.

3.2. Specifikáció

3.2.1. A képek feldolgozása

Bemenetként a szoftver egy videót fog kapni, melyet be kell olvasnia, és feldolgozni a további műveletek előkészítése céljából. Erre a feladatra a python számos ingyenesen hozzáférhető könyvtárat biztosít, melyek közül az OpenCV-t és a Scikit-Image-et is hasonlóan alkalmasnak találtam a projekt számára, így végül az döntött az OpenCV mellett, hogy már dolgoztam vele korábban.

A bemeneti videót tehát az opencv segítségével fogom beolvasni, majd a mozgó objektumok felismerése céljából képkockáinként az irodalomkutatás részben már említett háttérkivonást, konkrétabban annak a k legközelebbi szomszéd (*k-nearest neighbour*) változatát fogom rájuk alkalmazni, ugyanis az viszonylag kisebb mértékű előtér pixelek esetén jól működik[14].

Ezt követően a kapott előtér maszkra zajcsökkentés céljából érdemes morfológiai műveleteket alkalmazni, ezek közül én az úgynevezett ”*opening*” műveletet fogom alkalmazni, ami az *erosion* és a *dilation* kombinációja, és az objektumok alakjának megtartásával képes csökkenteni a zajt, majd ennek az eredményét kombinálom az eredeti képkockával, hogy visszakapjam a színeket.

	Pillow	OpenCV	Scikit-Image	PyTorch
előnyök	sok fájlformátumot támogat, egyszerű a használata	függvények széles köre érhető el, jó teljesítmény	jól dokumentált, jól integrálható tudományos számításokkal	jól használható deep learning-gel kapcsolatos feladatokra, gyors
hátrányok	lassabb a többinél, videófeldolgozásra nem optimális	valamivel komplexebb	bizonyos feladatokra nincs függvénye	általános képfeldolgozásra kevésbé alkalmas

2. táblázat. A python néhány népszerűbb képfeldolgozó könyvtárának összehasonlítása

3.2.2. Az objektumok klasszifikációja

Mivel a projekt célja a les megállapításának automatizálása lenne, ezért a klasszifikációt is érdemes lenne emberi beavatkozás szükségessége nélkül elvégezni. Erre a [8]-ban elvégzett tesztek alapján a k-means klaszterezés módszere tűnik a legmegbízhatóbbnak, így én is ezt fogom használni. Ennek során a játékosok csapatuknak megfelelően el lesznek különítve.

A labda detektálására az eredmények alapján a hullám dekompozíció és független komponensanalízis módszereinek párosítása tűnik a legmegbízhatóbbnak, így ezt fogom alkalmazni. Ennek eredményeként a labdához legközelebbi játékos (ha az egy bizonyos távolságon belül tartózkodik) meg lesz jelölve.

A játékosok és a labda követéséhez a [3]-ban is alkalmazott *bounding box*-os megoldást fogom alkalmazni.

3.2.3. A les megállapítása

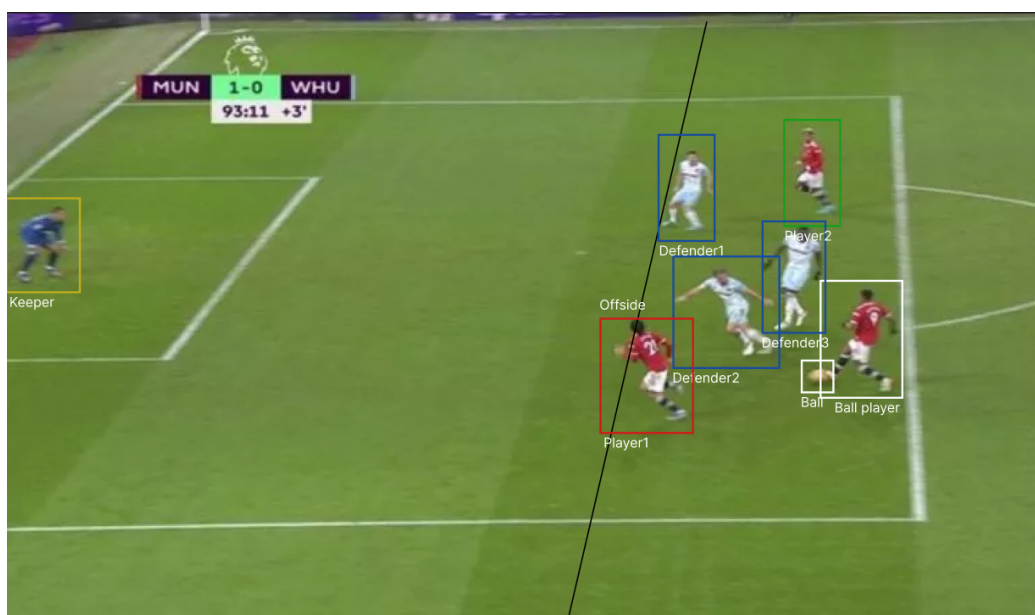
A detektálást és klasszifikálást követően a *bounding box*-ok pozíciójából meg tudjuk állapítani a védekező csapat második játékosát, illetve, hogy mely játékosok azok, akik leshelyzetben tartózkodnak.

3.2.4. Felhasznált bemenetek

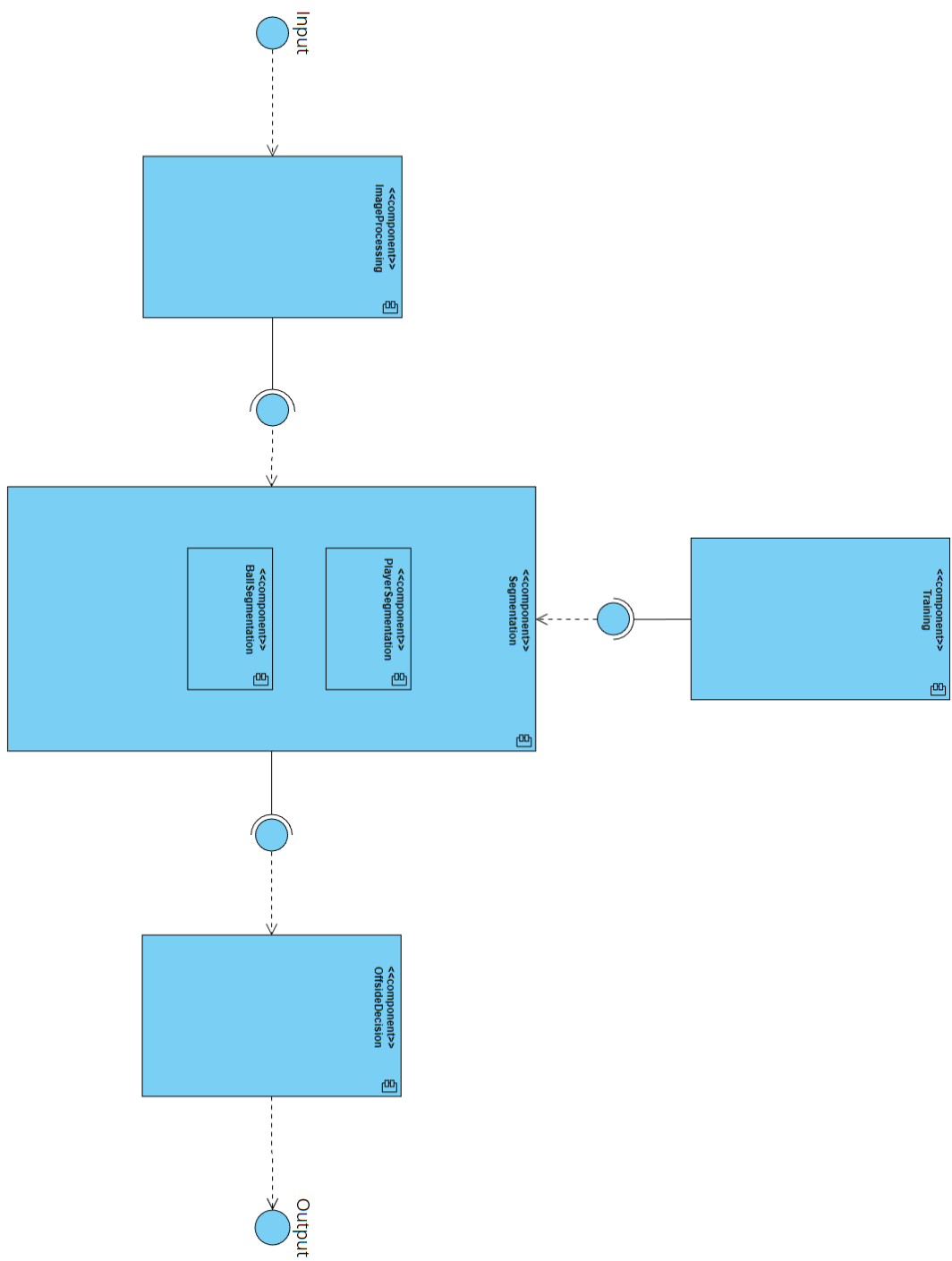
A bemeneti videók kérdése elsőre triviálisnak tűnhet, azonban ha jobban belegondolunk, egy futballmérkőzés közvetítése során a kamera szinte folyamatos mozgásban van, ami rendkívül megnehezíti a felvétel megfelelő feldolgozását. Azonban ha megpróbálunk az interneten rögzített kameraállású felvételeket keresni, azt fogjuk tapasztalni, hogy nem lehet egyszerűen fellelni elegendő mennyiségű és minőségű videóanyagot. Hosszas keresgélés után azt a döntést hoztam, hogy nekivágok a projektnek televíziós közvetítési felvételek felhasználásával, melyeket a Hashtag United hivatalos Youtube csatornájáról[15] szerzek be. Felmerült még bennem, hogy felhasználásra kerülhetne felvétel a Football Manager nevű szimulátoros videójátékból, ugyanis a mérkőzések kamerájának állása több módon is állítható, azonban az ötletet elvettem, mert bár a játék grafikai motorja évről évre egyre realisztikusabb, még mindig nem tudja tökéletesen reprezentálni egy valós futballmérkőzés történéseit.

3.2.5. Kimenet

Az aktív és passzív leshelyzetek közti különbségtétel bizonyos esetekben még a játékvezetők között is képes ellentétes véleményeket kiváltani, egy számítógép pedig még kevésbé képes ezt helyesen eldönteni, így arra a döntésre jutottam, hogy a szoftverem nem fogja konkrétan megállapítani, hogy fennáll-e a leshelyzet, csupán vizuális segítséget nyújt a játékvezetők számára ahhoz, hogy meghozhassák a helyes döntést. Így tehát a kimenet egy, a lehetőségekhez mérten nagyjából valós-idejű videófelvétel lesz, melyen a játékosok *bounding box*-okkal vannak megjelölve, melyek színei és a hozzájuk társított feliratok jelölik, hogy egy adott játékos melyik csapathoz tartozik, illetve, hogy lesen tartózkodik-e éppen.



2. ábra. Figmával készített vázlat a program kimenetéről. Az eredeti kép forrása:[16]



3. ábra. A rendszer komponensdiagramja

Hivatkozások

- [1] The International Football Association Board, *Laws of the Game*. 2021.
- [2] FIFA.com, „FIFA organises remote demonstration of advanced offside technology,” 2021.
- [3] T. D’Orazio, M. Leo, P. Spagnolo, P. L. Mazzeo, N. Mosca, M. Nitti, and A. Distante, „An investigation into the feasibility of real-time soccer offside detection from a multiple camera system,” *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 19, pp. 1804 – 1818, 01 2010.
- [4] M. Assadifard and M. o. M. Javanmard, „Using digital image processing solutions for offside recognition, instead of assistant referees,” *The 2013 World Congress in Computer Science Computer Engineering and Applied Computing*, 07 2013.
- [5] P. N. Patil, R. J. Salve, K. R. Pawar, and P. M. M. P. Atre, „Offside detection in the game of football using contour mapping,” 2018.
- [6] T. Kanade, R. T. Collins, A. J. Lipton, and P. J. Burt, „Advances in cooperative multi-sensor video surveillance,” 1999.
- [7] J. Lee and A. Bovik, „Chapter 19 - video surveillance,” in *The Essential Guide to Video Processing* (A. Bovik, ed.), pp. 619–651, Boston: Academic Press, 2009.
- [8] P. L. Mazzeo, P. Spagnolo, M. Leo, and T. D’Orazio, „Football players classification in a multi-camera environment,” vol. 6475, pp. 143–154, 12 2010.
- [9] M. Leo, T. D’Orazio, P. Spagnolo, P. L. Mazzeo, and A. Distante, „Sift based ball recognition in soccer images,” vol. 5099, pp. 263–272, 07 2008.
- [10] X. Yu, H. Leong, C. Xu, and Q. Tian, „Trajectory-based ball detection and tracking in broadcast soccer video,” *Multimedia, IEEE Transactions on*, vol. 8, pp. 1164 – 1178, 01 2007.
- [11] J. Ren, J. Orwell, G. Jones, and M. Xu, „A general framework for 3d soccer ball estimation and tracking,” in *2004 International Conference on Image Processing, 2004. ICIP ’04.*, vol. 3, pp. 1935–1938 Vol. 3, 2004.

- [12] N. Ancona, G. Cicirelli, A. Branca, and A. Distanto, „Goal detection in football by using support vector machines for classification,” vol. 1, pp. 611 – 616 vol.1, 02 2001.
- [13] M. Leo, T. D’orazio, P. Spagnolo, and A. Distanto, „Wavelet and ica preprocessing for ball recognition in soccer images,” 2005.
- [14] OpenCV, „cv::backgroundsubtractor class reference.”
- [15] H. United, „Full match! - white ensign vs hashtag united.” YouTube playlist.
- [16] caughtoffside.com, „(photo) – the potential offside moment for marcus rashford’s late winner for man united vs west ham.”