Előadás értékelő rendszer érzelemdetektálási alpokon

Lecture evaluate system based on emotion recognition

(revised December 2019)

Kerekes Ákos, Gaják Tibor István, Hannos Attila

**Kivonat**—Manapság a XXI. század egyik legjobban felkapott témája a Mesterséges Intelligencia és azon belül a Deep Learning alkalmazása. Számos területen alkalmazzák a témában elért eredményeket. A dolgozat célja egy képfeldolgozás témakörében megoldandó probléma felvetése majd megoldása Deep Learning eszközök segítségével. Az emberek gyakran nem tudják és nem is akarják eltitkolni érzéseiket. Ezek az érzések kiülnek az arcukra. Célunk egy olyan rendszer fejlesztése ami ezeket az érzelmeket képes detektálni. Az információ bírtokában rengetek alkalmazásmódot el lehet képzelni. Gondoljunk csak bele, hogy egy banknak, vagy akár bármilyen szolgáltatónak az az információ, hogy a szolgáltatást igénybe véve az ügyfél elégedett volt-e vagy szomorúan esetleg mérgesen távozott, az nagyon sokat tud segíteni a szolgáltatás minőségének javításában. Szintén egy alkalmazási lehetőség egy tanárnak felmérni, hogy a diákok hogy érezték magukat az előadáson, milyen érzésekkel távoztak. A fejlesztendő alkalmazás érzelmeket detektál a képeken majd összesíti a talált érzelmeket és megjeleníti kördiagram formájában illetve jellemzi az adott szolgáltatást/előadást egy összesített érzelemmel.

**Abstract**—

xxxx-xxxx/0x/$xx.00 © 200x IEEE Published by the IEEE Computer Society

—————————— ◆ ——————————

# 1 Introduction

A képfeldolgozás területén egyre gyakrabban használnak tanuló algoritmusokat. Gondoljunk csak az önvezető autókra. A hagyományos látás algoritmusai kezdenek a háttérbe szorulni. A választott témánkon keresztül a célunk a képfeldolgozás esetén használatos mély neurális hálózatok megismerése és gyakorlatba ültetése volt. Az arcok, arcokon lévő érzelem detektálása manapság már egy ismert, körüljárt probléma, viszont a felismert érzelem hasznosítására számtalan alkalamzásötletet ki lehet találni. A mi ötletünk egy olyan rendszer elkészítése volt ami az oktatónak lehet a segítségére úgy, hogy a diákok arcát, érzéseit figyelve böngészőn keresztül egy visszajelzést ad az oktatónak a detektált érzelmek alapján, így az oktató könnyebben el tudja dönteni, hogy a diákok alapvetően elégedettek az órával, vagy inkább unottak, mérgesek esetleg szomorúak.

# 2 Korábbi megoldások

## 2.1 irodalomkutatás

Detailed submission guidelines can be found on the author resources Web pages. Author resource guidelines are specific to each journal, so please be sure to refer to the correct journal when seeking information. All authors are responsible for understanding these guidelines before submitting their manuscript. For further information on both submission guidelines, authors are strongly encouraged to refer to <http://www.computer.org/portal/web/peerreviewjournals/author>.

## 2.2 irodalomkutatás

For papers accepted for publication, it is essential that the electronic version of the manuscript and artwork match the hardcopy exactly! The quality and accuracy of the content of the electronic material submitted is crucial since the content is not recreated, but rather converted into the final published version.

All papers in IEEE Computer Society Transactions are edited electronically. A final submission materials check list, transmission and compression information, and general publication materials can be found at: <http://www.computer.org/portal/web/peerreviewjournals/author>.

## 2.3 irodalomkutatás

All tables and figures will be processed as images. You will have the greatest control over the appearance of your figures if you are able to prepare electronic image files. Save them to a file in PostScript (PS) or Encapsulated PostScript (EPS) formats. Use a separate file for each image. File names should be of the form “fig1.ps” or “fig2.eps.”

For more information on how to format your figure or table files for final submission, please go to <http://www.computer.org/portal/web/peerreviewjournals/author#figures> and [View transactions art\_guide.pdf (PDF, 4.69MB)](http://www.computer.org/cms/Computer.org/Journal%20templates/transactions_art_guide.pdf).

## irodalomkutatás

An IEEE Computer Society copyright form must accompany your final submission. You can get a .pdf, .html, or .doc version at <http://computer.org/copyright.htm>. Authors are responsible for obtaining any security clearances.

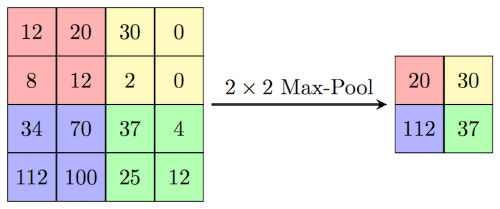
For any questions about initial or final submission requirements, please contact one of our staff members. Contact information can be found at: <http://www.computer.org/portal/web/volunteercenter/staff>.

# C:\Users\Akos\AppData\Local\Microsoft\Windows\INetCache\Content.Word\haar.png2.5 irodalomkutatás

As demonstrated in this document, the numbering of sections is upper case Arabic numerals, then upper case Arabic numerals, separated by periods. Initial paragraphs after the section title are not indented. Only the initial, introductory paragraph has a drop cap.

# Rendszerterv

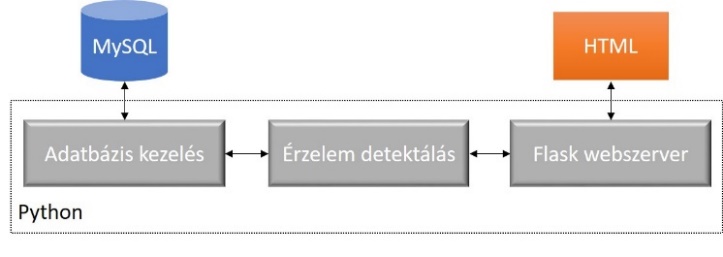
## 3.1 A mély neurális háló felépítése

Az érzelemdetektálásra alapvetően konvolúciós neurális hálót használunk, ahogy az a képfeldolgozás során általánosan megszokott. A hálózatunkat kiegészítjük max-pool rétegekkel. Ennek oka az, hogy amennyiben egymás után helyezünk egyre mélyebb konvolúciós rétegeket, akkor látható, hogy egy idő után az aktivációs tomb mérete óriásira nő. Ezért néhány rétegenként érdemes pooling műveletet végrehajtani, ami csökkenti az aktivációs tomb méretét. Mi a max-pool eljárást alkalmaztuk. Az eljárás működése az alábbi ábrán szemléltetésre kerül.

A konvolúciós rétegeken és a max-pool rétegeken kívül a hálózat dropout rétegeket is tartalmaz. Ennek a rétegtípusnak az a lénysege, hogy a tanítás során minden előreterjesztés lépéskor az egyes rétegek aktivációinak egy hányadát véletlenszerűen kinullázzuk (ezzel gyakorlatilag lekapcsoljuk a hálózatunk egy részét). Ezekre a rétegekre a túlillesztáés (overfitting) lehetőségének csökkentése miatt van szükségünk. Természetesen szükségünk van batch normalizálásra is szintén az overfitting elkerülése miatt illetve a konvergencia javítása érdekében is. A batch normalizálás technológia azt jelenti, hogy az egyszerre feldolgozandó minibatchek átlagait és szórásait folyamatosan számolva az egyes aktivációkat ezen értékek segítségével a tanítás során foylamatosan normalizáljuk. A hálózat vége felé haladva szükségünk van fully connected rétegekre is, a hálózat kimenete is egy ilyen réteg a kategóriáknak megfelelő számú neuronal illetve softmax aktivációval a valószínűségi változók előállításának érdekében. Viszont lényeges momentum, hogy a hálónk csak abban az esetben működik jól, ha a bemenetén egy arcot kapnem pedig egy nagy képet több arccal, sok objektummal. Az arcok leválogatására a képfeldolgozások esetén méltán híres OpenCV könyvtár

Cascade Classifier Haar cascade alapú függvényét alkalmaztuk. Ezt az eljárást általában arcdetektálásra szokás alkalmazni. A módszer egy speciális képjellemző fajtát használ fel amit Haar típusú képjellemzőnek nevezünk. Ezek a jellemzők egy képrészletet egy bináris ablak segítségével vizsgálnak melynek a kimenete nagy pozitív szám hasonlóság esetén és nagy negatív szám ellentétesség esetén, nulla körüli ha nincs hasonlóság.

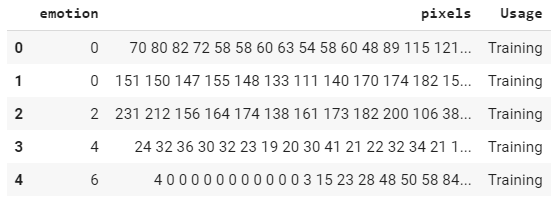
## 3.2 Az alkalmazás felépítése

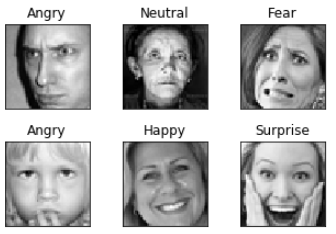
A rendszer lelkét az érzelemdetektáló modul adja ami egy nagy osztályként lett elkészítve. Az alkalmazásunkhoz szálkezelést alkalmaztunk a webfelület kiszolgálásának érdekében. Az alkalmazás indításkor példányosítja az érzelemdetektáló osztályt majd elindítja a webfelületet illetve az érzelem detektálást egy külön szálon a csatlakoztatott kamerán. Az indulási folyamat része a MySQL alapú adatbázishoz való csatlakozás is. Az adatbázisra csatlakozás segítségével tudjuk elmenteni a különböző detektált érzelmeket és tudunk statisztikát készíteni.

# Az alkalmazás megvalósítása

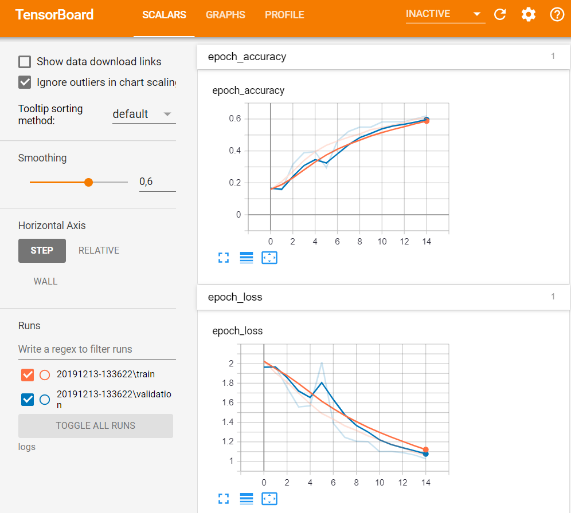
## 4.1 Adatok begyűjtése

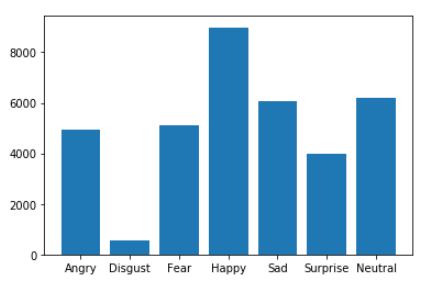
A Deep Learning alapú problémák megoldásának első lépése mindig egy megfelelő adatbázis elkészítése vagy egy már létező adatbázis felhasználása, amennyiben találunk számunkra megfelelőt. Az adatbázis előállítás ennek a területnek az egyik legköltségesebb része, főleg felügyelt tanítás esetén amikor a tanító adatok mellé az elvárt, helyes címkét is társítanunk kell. Publikus adatbázisokat böngészve találtunk rá a Kaggle weboldalon lévő fer2013-as adatbázisra, ami csv fájlformátumban tartalmaz arcokról 48x48 pixel méretű képeket. Az adatbázis hétféle érzelmet különböztet meg. A nullás értékkel jelzett érzelem az angry, az egyes érték a disgust, a kettes a fear, hármas a happy, négyes a sad, ötös a surprise, hatos a neutral.

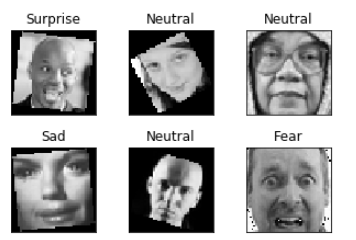
Az adatbázis felépítése:

Az első lépés az adatbázisban lévő adatok megismerése, vizualizációja volt.

A vizualizáció után előkészítettük az adatokat a tanításra, viszont megvizsgálva az adatok eloszlását a csoportok között azt tapasztaltuk, hogy nem egyenletes az eloszlás.





A képeket a nagyobb pontosság elérése érdekében mesterségesen átalakítottuk (image augmentation) így kiegyenlítve a különböző osztályokba tartozó képek mennyiségét. Ilyen augmentációs technikák például a kép tükrözése, kis mértékű forgatása, zajjal terhelése.

Az sklearn programcsomag train\_test\_split függvényével az elkészült augmentált adatbázist három részre daraboltuk 70% train, 20% validation, 10% train arányban.

## 4.2 A tanítás folyamata

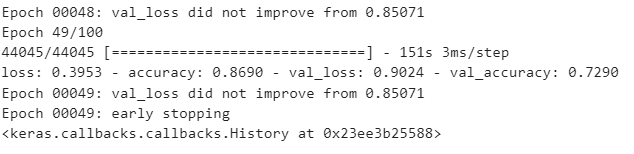
A tanításhoz a Keras keretrendszert használtuk. Az adatbázis létrehozása után definiálnunk kellett a modellünket majd meg kellett határoznunk az optimalizálás módszerét. Különböző callback függvényeket alkalmaztunk a tanítás folyamatának pontosabbá tételének érdekében. Ilyen például a Keras ReduceLROnPlateau függvénye ami a learning rate-re van hatással. Reduce learning rate when a metric has stopped improving. A következő alkalmazott callback függvények a ModelCheckpoint és az EarlyStopping voltak. Ezeket célszerű együtt alkalmazni. A ModellCheckpoint segítségével tudjuk mindig a legjobban teljesítő modellünket elmenteni. Az EarlyStopping egy overfitting elkerülésére szolgáló módszer. A háló hiperparaméterei közé tartozik a tanítás futtatásának hossza is. Amennyiben túl kevés epoch-ig futtatjuk a tanítást a modellünk nem tanul rá a kívánt módon a tanító adatbázisra.

Ha túl sokáig futtatjuk a tanítást abban az esetben túltanul, ami azt jelenti, hogy a tanító adatbázison akár a 100% pontosságot is elérheti, de az általánosítás rovására a tanító adatokat kezdi el memorizálni a hálózat, tehát új képek felismerése esetén valószínüleg pontatlanabbul fog működni. Mivel nehéz meghatároznia kellő epoch mennyiséget, így az EarlyStopping módszert hívjuk segítségül. A módszer lényege az, hogyha a hálónk egy, a hiba minősítésére szolgáló metrikája (általában validation loss) nem csökken a tanító lépések során akkor leállítjuk a tanítást. A módszer rendelkezik egy patience beállítással. Ez mondja meg azt hogy mennyi ideig várunk változatlan hibaérték mellett a tanítás leállításáig. Az utolsó callback amit használtunk a TensorBoard. Ez a kiegészítő egy webes felületen keresztül grafikonok formájában szolgáltat információt a tanításról. Különböző paraméterek optimalizálásakor, a tanítás áttekintésekor nagyon hasznos tud lenni.

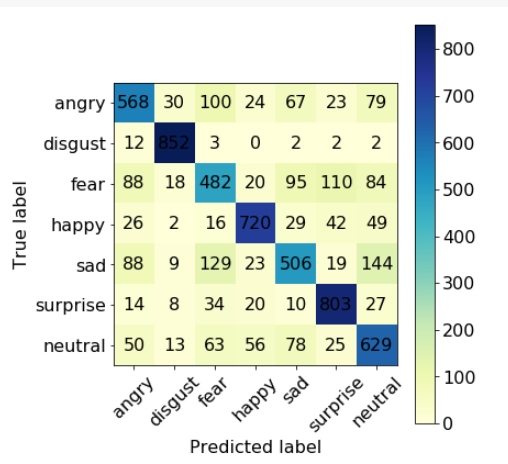
## 4.3 hiperparaméter optimalizálás

A háló hangolására, javítására szoktak alkalmazni hiperparaméter optimalizációt. Egy mély neurális hálónak nagyon sok hiperparamétere van. A teljesség igénye nélkül párat megemlítve: learning-rate, ami a gradiens alapú módszerek lépésének nagyságát módosítja, a háló rejtett rétegeinek száma, az egyes rétegekben a neuronok száma, konvolúciós réteg esetén a szűrő mérete, batch méret. Látható, hogy óriási a hiperparaméter tér. Ha minden paramétert minden másik paraméterrel alkotott kombinációjában ki akarnánk próbálni akkor az algoritmusunk sose készülne el. A hiperparaméter optimalizálás gyorsítására automatizálására használhatunk különböző csomagokat, amik segítségével adott értékek közül véletlenszerűen választva minden epochra más-más hiperparaméter párokkal taníthatjuk a hálónkat.

# 5 Kiértékelés

A modellt többször újratanítottuk különböző paraméterekkel. Eleinte a hiperparaméter optimalizálást kézzel végeztük majd utána automatizálva epochonként változtatva a háló paramétereit.

Az ábrán a legjobb modell betanításának befejezése látható. A validációs hibánk 0.9 köré csökkent, a validációs pontosságunk 73% volt.

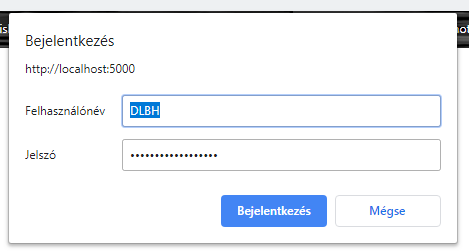
A modell értékelésére meghatároztuk a confusion matrixot.

A matrix kirajzolásával láthatjuk, hogy a teszt adatbázisunkon a különböző képeket hogyan azonosítja a modell, hol téved a leggyakrabban, hol a legpontosabb.

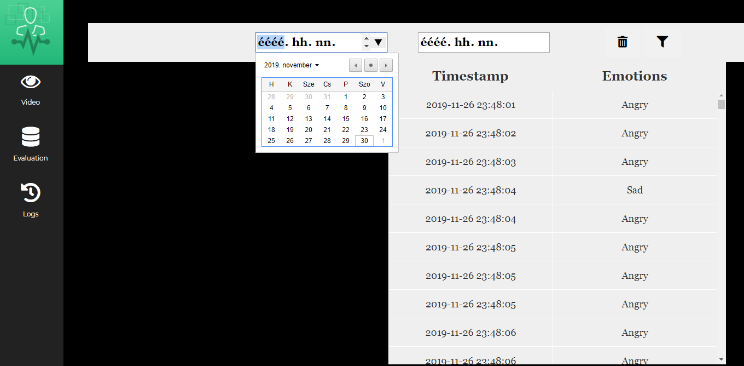
Az ábrából leolvasható, hogy a modell a legtöbbet a sad képet azonosítja helytelenül neutralnak. A legpontosabb a disgust érzelem meghatározásakor. A modell pontossága a teszt adatbázison körülbelül 64% volt.

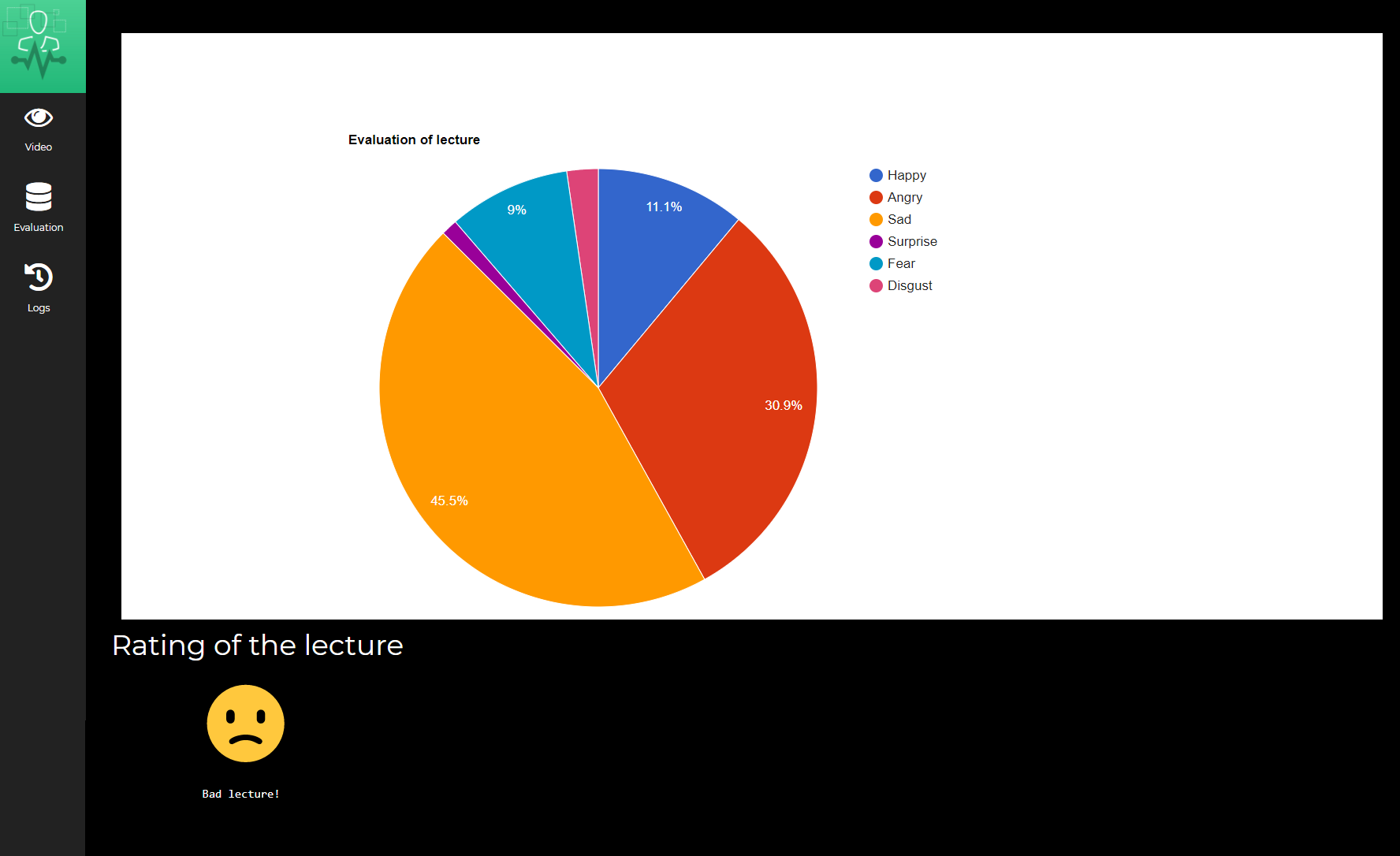
# Az elkészült applikáció bemutatása

Az elkészült aplikáció webes felületen keresztül képes videót vagy webkameraképet elemezni. Képes kijelezni a detektált arcokat és mellé kijelezni a detektált érzelmeket. A detektált érzelmeket adatbázisban tároljuk. A felhasználónak lehetősége van idő szerint szűrni az adatbázist, illetve elkészült egy összegző diagram is ahol a különböző érzékelt érzelmek összesítése látható. Az összegzett adatok alapján az algoritmus eldönti, hogy az adott tanóra, előadás milyen hangulatban telt a diákok körében.









# Conclusion

A feladat megoldása során meg tudtunk ismerkedni az alapvető Deep Learning témakörébe tartozó módszerekkel. Eredményesen be tudunk tanítani egy mély neurális hálózatot és ezzel bővíteni tudtuk mérnöki szemléletünket. Érdekes problémákkal kerültünk szembe, például a terület óriási erőforrásigényével. Korábban olyan algoritmusokkal ami akár több napig, hétig futnak, nem találkoztunk.

# Future planes

Az alkalmazás továbbfejlesztésével kapcsolatban több ötletünk is van. Az első a kép feltöltésének és kielemzésének a lehetősége. Különböző egyéb technikákat, rétegeket, háló felépítést kipróbálva szeretnénk a modellünket még pontosabbá tenni.

**References**

1. Shervin Minaee, Amirali Abdolrashidi Expedia Group University of California, Riverside Deep-Emotion: Facial Expression Recognition Using Attentional Convolutional Network <https://arxiv.org/pdf/1902.01019.pdf>
2. Ching-Da Wu and Li-Heng Chen The University of Texas at Austin FACIAL EMOTION RECOGNITION USING DEEP LEARNING <https://arxiv.org/pdf/1910.11113.pdf>
3. Suraj Tripathi1, Abhay Kumar1, Abhiram Ramesh1, Chirag Singh1, Promod Yenigalla1 1Samsung R&D Institute India – Bangalore Deep Learning based Emotion Recognition System Using Speech Features and Transcriptions <https://arxiv.org/ftp/arxiv/papers/1906/1906.05681.pdf>
4. Shan Li, and Weihong Deng∗ , Member, IEEE A Deeper Look at Facial Expression Dataset Bias <https://arxiv.org/pdf/1904.11150.pdf>
5. Shima Alizadeh, Azar Fazel Convolutional Neural Networks for Facial Expression Recognition <https://arxiv.org/pdf/1704.06756.pdf>
6. Tianyuan Chang, Guihua Wen, Yang Hu, JiaJiong Ma Facial Expression Recognition Based on Complexity Perception Classification Algorithm <https://arxiv.org/ftp/arxiv/papers/1803/1803.00185.pdf>
7. Ram Krishna Pandey, Member IEEE, Souvik Karmakar, A G Ramakrishnan, Senior Member IEEE and N Saha Improving Facial Emotion Recognition Systems Using Gradient and Laplacian Images <https://arxiv.org/pdf/1902.05411.pdf>
8. Samanyou Garg GROUP EMOTION RECOGNITION USING MACHINE LEARNING <https://arxiv.org/ftp/arxiv/papers/1905/1905.01118.pdf>
9. Facial Emotion Recognition using Convolutional Neural Networks <https://arxiv.org/pdf/1910.05602.pdf>
10. Predicting Personal Traits from Facial Images using Convolutional Neural Networks Augmented with Facial Landmark Information <https://arxiv.org/pdf/1605.09062.pdf>
11. DeXpression: Deep Convolutional Neural Network for Expression Recognition <https://arxiv.org/pdf/1509.05371.pdf>