

Mesterséges intelligencia a tűzserész feladatokban – Tűzserész Támogató Információs Rendszer szoftveres alapjai III. rész

Artificial intelligence in explosive ordnance disposal tasks – Software basics of EOD Support Information System Part III

Összefoglalás: A Mesterséges intelligencia a tűzserész feladatokban című négyrészes cikksorozat fő célja a Mesterséges Intelligencia Alapú Tűzserész Támogató Információs Rendszer bemutatása. A cikksorozat III. részében szó esik a képelemzés szoftveres és matematikai hátterének lényegéről, ami alapján a mesterséges intelligencia egy adott képet az előre meghatározott csoportok valamelyikébe sorol. A képelemző szoftver működése jelenti a tűzserészek támogatását végző program legfontosabb elemét.

Kulcsszavak: tűzserész, mesterséges intelligencia, robbanótestek, felismerő rendszer, aknagránát

Abstract: The main goal of the four-part article series entitled Artificial Intelligence in EOD Tasks is to present the EOD Support Information System Based on Artificial Intelligence. In Part III of the series of articles, the essence of the software and mathematical background of image processing, on the basis of which the artificial intelligence classifies a given image into one of the predefined groups. The operation of the image processing software is the most important element of the EOD supporting program.

Keywords: EOD, artificial intelligence, explosive ordnance, recognition system, mortar rounds

1. Bevezetés

A Mesterséges Intelligencia Alapú Tűzserész Támogató Információs Rendszer célja, hogy a tudomány jelen állása szerinti legkorszerűbb feltételeket nyújtsa a tűzserész műveletek irányításának és szervezésének támogatására. Ezt a vezetési támogatást a mesterséges intelligencia alapú képelemző szoftver biztosítja, amely valós idejű és szűrt, tehát megfelelően hitelesített adatokat biztosít a parancsnok részére a minél pontosabb és körültekintőbb döntéshozatalhoz. A rendszer gyakorlati felhasználására két területen, egyrészt a rendőr járőr, másrészt a tűzserész járőr tevékenysége során kerülhet sor.

2. Mesterséges intelligencia alapú tűzserész támogató információs rendszer

A mesterséges intelligencia, vagyis az MI² alapú szoftvert a lakossági bejelentést követően a kiérkező rendőr járőr tudná első lépésben felhasználni, aki ennek segítségével meghatározhatja a robbanótest kategóriáját (nyolc alapvető kategóriát különböztettem meg), illetve a tűzserész műveleti vezetés kérésére további csoport és alcsoport, tehát konkrét eszköz meghatározás is történhetne. Ennek, és az eszköz helyzetének ismeretében sokkal egyszerűbbé válna a robbanótest mentesítés prioritásának (sürgősségének) meghatározása, valamint következtetni lehetne a mentesítéshez szükséges alapvető felszereléseken (pl. robbanóanyag) túl az esetleges speciális eszközigényére (pl. csörlő, emelődaru, vegyvédelmi biztosítás) is. Másik fontos

¹ MH 1. Tűzserész és Folyamőr Ezred, szakaszparancsnok, e-mail: Adam.Balazs@uni-nke.hu, ORCID: <https://orcid.org/0000-0003-0597-4528>

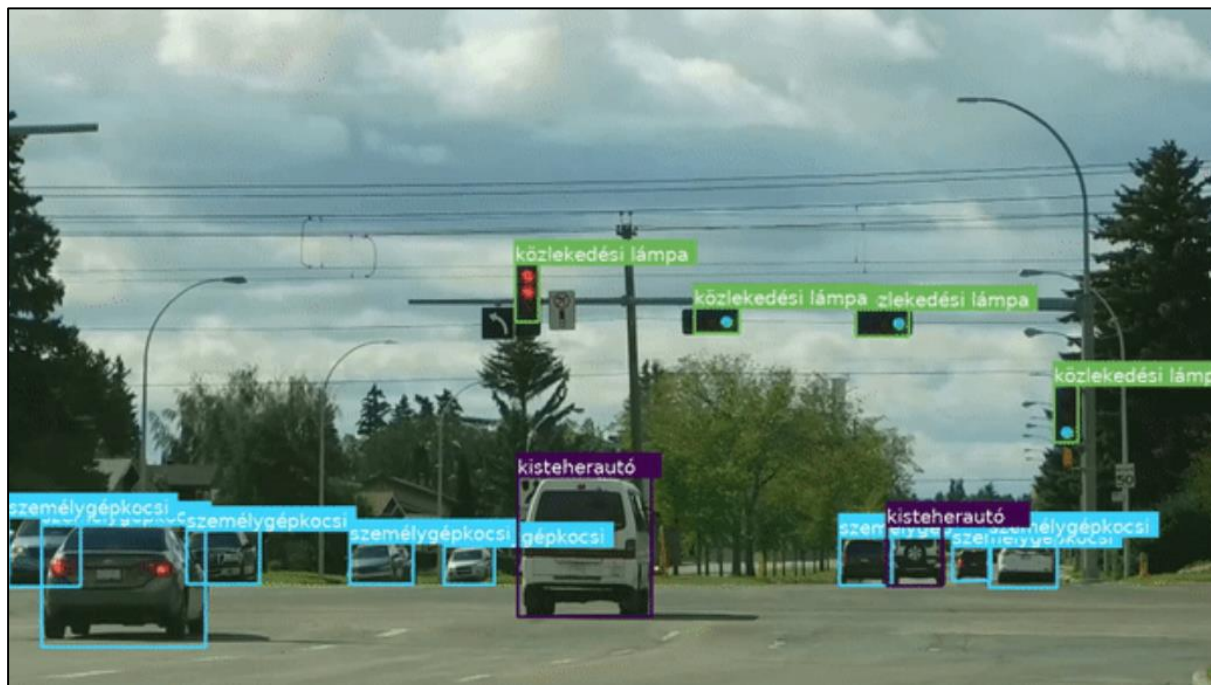
² Angol megnevezése: Artificial Intelligence – AI.

alkalmazási területet jelentene, a bejelentések során a tényleges robbanótestek és a nem robbanó testek (NRT) megkülönböztetése. Ugyanis a bejelentések hozzávetőleg 10%-a nem robbanó test,³ ami az éves átlag 2000 riasztás esetében már nem elenyésző szám.⁴ A szoftver segítségével az ezekre a bejelentésekre fordított személyi, anyagi és technikai kiadásokat lehetne megspórolni. A fejlesztés egy későbbi szakaszát követően a tűzszerész járőr feladatvégrehajtását is támogathatja a rendszer, amennyiben a hosszú tanítási folyamat eredményeként a tévesztés kockázata már minimálisra redukálódott. Ebben az esetben a felismerő szoftver önállóan képes lenne azonosítani a robbanótestet és egy tűzszerész mentesítő robot vezérlésébe integrálva az képes lehet a hatástalanítás egyes meghatározott folyamatait autonóm módon végrehajtani.

A támogatás lényege, hogy az előtalált robbanótestről a Magyar Honvédség (MH) Tűzszerész Ügyeletére beérkező első információk alapján minél pontosabban behatárolják az előtalált eszköz kategóriáját és típusát. Az így megszerzett információk alapján az eszköz veszélyességi foka is sokkal nagyobb valószínűséggel behatárolható. Illetve a feladatra küldhető, átirányítható járőrök felszerelésének, fogyó anyagainak és robbanószer mennyiségének függvényében azok sokkal nagyobb precizitással, hatékonysággal irányíthatók át egyik feladatról a másikra. Ezek olyan képességek, amelyek akár már néhány éven belül is az MH 1. Tűzszerész és Folyamőr Ezred (a továbbiakban: Tűzszerész Ezred) parancsnoki állománya számára átfogó és hiteles döntéstámogatást nyújthatnak.

3. Objektum felismerés vagy képelemzés

Napjaink mesterséges intelligencia kutatásai legnagyobb számban az objektum felismerés vagy képelemzés témájában zajlanak. Az objektum felismerés kifejezést gyakran szinonimaként használják a képelemzés és képfeldolgozás szavakra, de a két fogalom között van lényeges eltérés is.⁵



1. ábra: Objektum felismerés a valóságban (*Objektum felismerő (képfelismerő) AI megoldásunk* 2019)

³ EMBER 2020a: 59–63

⁴ EMBER 2020b: 32–42

⁵ *Objektum felismerő (képfelismerő) AI megoldásunk* 2019

Az objektum felismerés fogalmát leggyakrabban azokra a szoftverekre kell érteni, amelyek élő képen, de inkább videó anyagon képesek kategorizálni és felismerni az előre betáplált alakzatokat. Ezek egymástól gyökeresen eltérő méretű, formájú, színű és egyéb tulajdonságú tárgyak, eszközök. Éppen ezért a program célja csupán a tárgyak egymástól történő megkülönböztetése (1. ábra) és nem az azonos tárgyak közötti különbségek vizsgálata.⁶

Ezzel szemben a képelemzés (Image Processing) lényege, hogy az egymáshoz nagyon hasonló tárgyak, tárgykategóriák között az adatbázis felhasználásával a tanulási folyamat eredményeként a létrejövő neurális háló képes legyen minimális, apró színbeli, formabéli vagy más eltéréseket azonosítani, majd betanulni és így valamekkora bizonytalansággal a későbbiekben a felhasználás során meg is különböztetni azokat. Tehát a képelemzés már egy sokkal precízebb és pontosabb beazonosítást eredményez,⁷ mint az objektum felismerés, de amennyiben a nagy eltérésekkel rendelkező objektumok valós idejű felismerése a cél, abban az esetben az objektumfelismerés eszközeinek alkalmazása a leghatékonyabb megoldás.⁸

Az objektum felismerés és a képelemzés technológiákat alkalmazó számos katonai és polgári fejlesztés látott már napvilágot, ezekről terjedelmi megfontolások miatt nem esik szó, de a hivatkozott cikkekben többet is meg lehet tudni róluk.⁹

Az objektum felismerés tűzszerész szakfeladatokban történő egyik alkalmazási lehetősége lenne az improvizált robbanóeszközök¹⁰ egyes elemeinek, mint például a robbanóanyag töltet, a detonátor, az indító mechanizmus, az áramforrás és vezetékek, az időzítő berendezés, a hatásfokozó repeszek, vagy a rejtést szolgáló különböző álcázó burkolat felismerése.¹¹ Ezek azonosítása azért kiemelt fontosságú, ugyanis a Közel-Kelet műveleti területein a saját csapatainkra mért támadások jelentős része ilyen IED-k alkalmazásával kerül végrehajtásra.¹²

4. Képelemzés elvi működése

A konvolúciós neurális hálóról¹³ a cikksorozat 2. részében már esett szó, ezúttal pedig célszerű azok működései elvét is megvizsgálni (2. ábra), hogy megértsük szerepüket a képelemzés területén. A konvolúciós hálózatok jelentősen eltérnek a többi neurális háló típusoktól. Ez leginkább azzal magyarázható, hogy ezeket a hálókat kifejezetten képi felismerésre fejlesztették ki.¹⁴

⁶ SHARMA 2019; KORSTANJE 2020

⁷ A képelemzés két fajtáját különböztetik meg, az analóg és digitális felismerést. Analóg felismerés esetén egy elkészült kép egyes pontjaihoz, pixelihez rendelnek szín értéket, így a kép egy számokkal leírható adathalmazzá válik, ami utána már változtatható. Ezt a technológiát használják a régi képek vagy videók digitális újra színezéséhez. A digitális képek esetében pedig a képek pixelihez már alapvetően hozzá van rendelve egy adat (x, y koordináta és szín), így az átalakítás nem szükséges. *Analog Image Processing vs. Digital Image Processing* [é. n.]

⁸ *Image Processing Definition, Examples and Application* [é. n.]

⁹ NÉMETH-VIRÁGH 2022: 2–7; NÉMETH-VIRÁGH 2023: 2–6

¹⁰ Angol megnevezése: Improvised Explosive Device – IED.

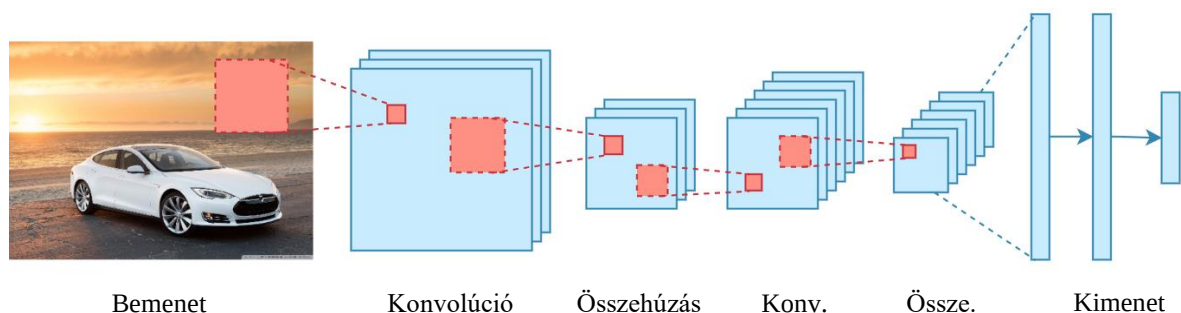
¹¹ KOVÁCS 2012: 37–52

¹² TOMOLYA-PADÁNYI 2012: 34–67

¹³ Angol megnevezése: Convolutional Neural Network – CNN.

¹⁴ TÓTH 2019

Fő előnye elődeihez képest, hogy emberi felügyelet nélkül, automatikusan is képes felismerni a fontosabb eltéréseket. Például macskákról és kutyákról készült képek alapján képes megtanulni az egyes csoportok jellegzetes vonásait, így azokat meg tudja különböztetni egymástól. Minden CNN modell hasonló architektúrát követ, amint az az alábbi ábrán látható.¹⁵



2. ábra: Konvolúciós folyamatok (DERTAT 2017) alapján a szerző szerkesztése

Az elemzésre váró kép a rendszer bemeneti információja, ez az input. A háló a kép egyes kijelölt részeiről egy számot generál a konvolúció matematikai formuláival. A konvolúciós műveletek összekapcsolódásával, azok sorozataként jön létre a neurális háló, amelyet számos teljesen összekapcsolt réteg alkot. A folyamat végén, az osztályozás (többosztályos) során létrejön a kimenet, a softmax. A CNN fő eleme a konvolúciós réteg. A konvolúció egy matematikai művelet két információhalmaz összevonására. Esetünkben a konvolúciót a bemeneti adatokra alkalmazzuk egy konvolúciós szűrő segítségével, így létrejön a jellemzőtérkép.¹⁶

1	1	1	0	0
0	1	1	1	0
0	0	1	1	1
0	0	1	1	0
0	1	1	0	0

3. ábra: Konvolúció bemenete (DERTAT 2017) alapján a szerző szerkesztése

1	0	1
0	1	0
1	0	1

4. ábra: Konvolúciós szűrő (DERTAT 2017) alapján a szerző szerkesztése

A példa kedvéért a 3. ábrán látható konvolúciós réteg a bemeneti kép, míg a 4. ábrán a konvolúciós szűrő, az úgynevezett kernel. A szűrő alakja miatt a példában bemutatott

¹⁵ What Is Deep Learning? 3 things you need to know [é. n.]

¹⁶ DERTAT 2017

folyamatot 3×3 -as konvolúciónak nevezik. A konvolúciós műveletet úgy hajtjuk végre, hogy ezt a kernelt a bemenetre helyezzük. Minden helyen elemenkénti mátrixszorzást végzünk a kernelben és a bemenetben található értékekre, majd az eredményt összegezzük. Ez az összeg kerül be a jellemzőtérképbe. Azt a „zöld területet”, ahol a konvolúciós művelet végbemegy, receptív mezőnek nevezzük. A szűrő méretéből adódóan a receptív mező is 3×3 -as méretű. Az 5. ábrán látható, ahogy a szűrő a bal felső sarokban van, a mátrixműveletek elvégzése után pedig, a konvolúciós művelet kimeneteként a jellemzőtérképen megjelenik a „4” kapott érték (6. ábra). Ezután jobbra kell csúsztatni a kernelt, és végre kell hajtani ugyanazt a műveletet, hozzáadva az eredményt a jellemzőtérképhez. Ezt a módszert folytatva elkészül a teljes térkép (7. ábra).¹⁷

1×1	1×0	1×1	0	0
0×0	1×1	1×0	1	0
0×1	0×0	1×1	1	1
0	0	1	1	0
0	1	1	0	0

5. ábra: Bemenet és a kernel (DERTAT 2017) alapján a szerző szerkesztése

4		

6. ábra: Jellemzőtérkép részlet (DERTAT 2017) alapján a szerző szerkesztése

4	3	4
2	4	3
2	3	4

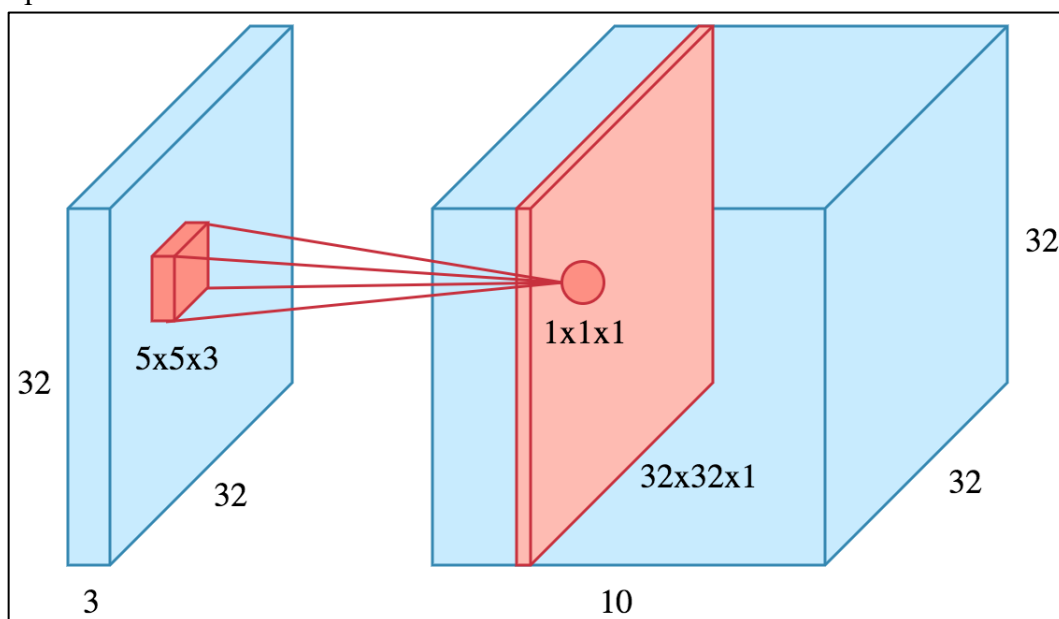
7. ábra: Teljes jellemzőtérkép (DERTAT 2017) alapján a szerző szerkesztése

A példa szerinti konvolúciós művelet két dimenzióban (2D) került bemutatásra, 3×3 -as szűrő használatával, de a tényleges felhasználás során ezeket a konvolúciókat három dimenzióban (3D) hajtják végre. A valóságban egy kép 3D-s mátrixként jelenik meg magassággal (y koordináta), szélességgel (x koordináta) és mélységgel, ahol a mélység a színcsatornáknak (RGB)¹⁸ felel meg. A konvolúciós szűrőnek meghatározott magassága és szélessége van, például 3×3 vagy 5×5 , és kialakításánál fogva lefedi a bemenet teljes mélységét, ezért is kell 3D-snek lennie. Az eredményes vizsgálathoz több konvolúciót kell végrehajtani egy

¹⁷ DERTAT 2017; BROWNLEE 2020

¹⁸ Red – Green – Blue színmodell

bemeneten, mindegyik esetben más-más szűrőt használva, amely így eltérő jellemzőterképet eredményez. Ezt követően egyesíteni kell ezeket a jellemzőterképeket, és ez lesz a konvolúciós réteg végső kimenete. A 8. és 9. ábrákon egyetlen szűrő alkalmazásával készült folyamatra láthatunk példát.¹⁹

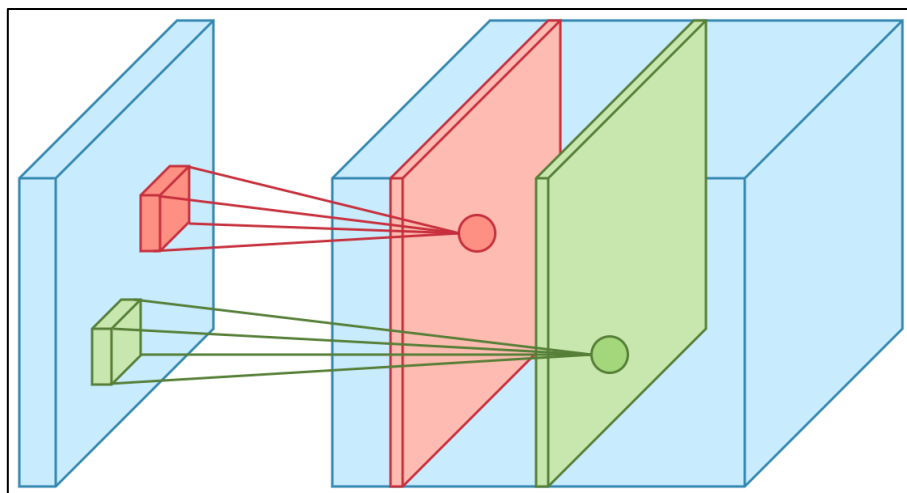


8. ábra: 3D konvolúció (DERTAT 2017)

Tegyük fel, hogy van egy $32 \times 32 \times 3$ -as képünk, és egy $5 \times 5 \times 3$ -as kernelt használunk, azaz a konvolúciós szűrő mélysége megegyezik a kép mélységével. A szűrő egy adott helyen lefedi a bemenet egy kis részét, itt végrehajtja a fent leírt konvolúciós műveletet. Az egyetlen különbség az, hogy ezúttal a mátrixszorzás összegét 3D-ben végezi és nem 2D-ben, de az eredmény továbbra is műveletenként egy-egy skalár. A bemeneten a szűrő áthelyezésével (a fenti módon) minden helyen megtörténik a konvolúció, amely eredményét a jellemzőterkép összesíti. Ez a jellemzőterkép $32 \times 32 \times 1$ méretű, a 8. ábra jobb oldalán piros szeletként látható. Ha 10 különböző szűrőt használunk, akkor 10 db $32 \times 32 \times 1$ méretű tereptérképet kapnánk, és ezeket a mélységi dimenzió mentén egymásra rakva megkapnánk a konvolúciós réteg végső kimenetét: egy $32 \times 32 \times 10$ méretű 3D mátrixot, amely a 8. ábra jobb oldalán nagy kék dobozként látható. A tereptérkép magassága és szélessége változatlan, azaz továbbra is 32, ami a kitöltésnek köszönhető. A 9. ábrán az látható, hogyan készül két jellemzőterkép a mélységdimenzióban eltolva egymástól. Az egyes szűrők konvolúciós műveleteit egymástól függetlenül hajtják végre, és az eredményül kapott jellemzőterképek diszjunktak, vagyis nincs közös részük.²⁰

¹⁹ DERTAT 2017

²⁰ DERTAT 2017

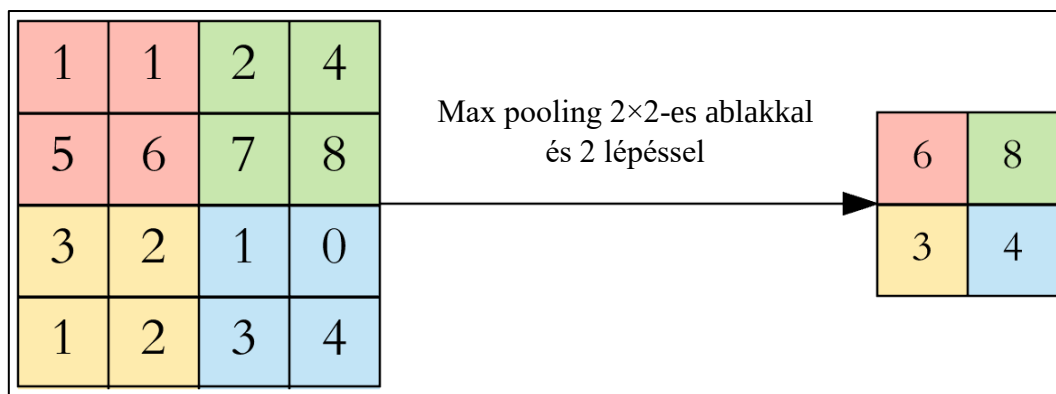


9. ábra: 3D konvolúció mélységi eltolással (DERTAT 2017)

A konvolúció után általában pooling (összehúzás) művelet történik a magasság és a szélesség csökkentése érdekében. Ez lehetővé teszi a paraméterek számának csökkentését, ami egyrészt lerövidíti a tanulási folyamat idejét, másrészt pedig a túltanulás kockázatát csökkenti. Az egyesített rétegek külön-külön leképezik az egyes tereptárgyak jellemtérképét, csökkentve a magasságot és a szélességet, a mélységen viszont nem változtatnak. A leggyakrabban használt összehúzó eljárás a max pooling, amely azt jelenti, hogy a pooling ablakban található maximális értéket veszi fel az adott terület értékének. A konvolúciós művelettel ellentétben a poolingnak nincsenek paraméterei. Egy ablakot csúsztat a bemenete fölé, és egyszerűen felveszi az ablakban található értékek közül a maximális értéket. A konvolúcióhoz hasonlóan meg kell határozni az ablak méretét és lépésszámát. Példaként egy 2×2 -es ablak és 2-es lépésszámú²¹ összevonás eredménye látható a 10. ábrán. Minden szín más ablakot jelöl. Mivel mind az ablak mérete, mind a lépésszám 2, az ablakok nem fedik egymást. Ennek a folyamatnak a célja, hogy csökkenjen a jellemtérkép szélessége és magassága. Összehasonlítva a jellemtérkép méreteit az összevonás előtt és után azt kapjuk, hogy ha a gyűjtőréteg bemenetének mérete pl. $32 \times 32 \times 10$ volt, akkor a pooling paraméterek használatával az eredmény egy $16 \times 16 \times 10$ -es jellemzőtérkép lesz. A térkép magassága és szélessége is a felére csökken, de a mélység nem változik, mert a pooling egymástól függetlenül működik a bemenet minden egyes mélységi szeletén. A bemeneti kép 2D koordinátái (súlyok) száma negyedére csökkent. Figyelembe véve, hogy a konvolúciós hálók általában több millió súllyal dolgoznak, nagyon fontos az adattömeg csökkentése. A CNN architektúrákban a pooling rendszerint 2×2 ablakkal és 2 lépéssel, míg a konvolúció 3×3 -as ablakkal, 1 lépéssel történik.²²

²¹ Az ablak kettő egységet mozog oldalra. Ezt az értéket konvolúció során érdemes 1-nek hagyni, pooling során pedig többnek is lehet, amennyiben megegyezik az ablak méretével nem lesz átfedés az ablakok között.

²² DERTAT 2017; BROWNLEE 2020; *Max Pooling* [é. n.]



10. ábra: Max pooling (DERTAT 2017) alapján a szerző szerkesztése

A teljes felismerési folyamat nagyrészt ezekből az alapfolyamatokból áll, kiegészülve még olyan tevékenységekkel, amelyekről terjedelmi megfontolás miatt a publikációban ezúttal nem esik szó, de lényegében ezeknek az értékeknek az összehasonlításán alapszik a képi felismerés elve. Tehát a mintahalmaz ilyen módon létrejövő értékei kerülnek összehasonlításra a bemeneti kép értékével, ami alapján a mesterséges intelligencia meghatározza, hogy a kép melyik csoportba tartozik.

5. Szoftveres háttér

A kutatás kezdetén fontos volt egy olyan szoftver keresése, amely megfelelő hatásfokkal és pontossággal dolgozik, mégis kevés programozási ismeretre van szükség használatához. Az MI Koalíció Bevezetés a Mesterséges Intelligencia világába című kurzusát hallgatva ismertem meg a „Teachable Machine” alkalmazást, amit a kutatás során is használtam.

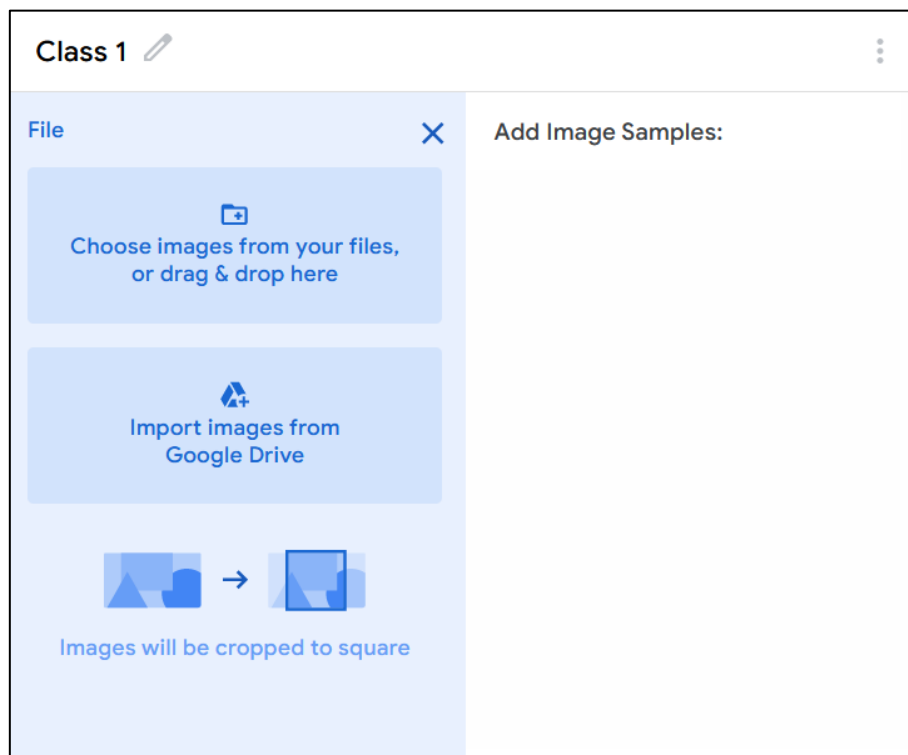
A Teachable Machine egy webalapú eszköz, a Google fejlesztése, amely gyors és egyszerű használatot biztosít, így mindenki számára elérhetővé teszi a gépi tanulási modellek létrehozását akár kép, zene vagy testhelyzetek felismerésére van szükség. A webes alkalmazás első verzióját 2017-ben ismerhették meg a felhasználók. Azóta számos fejlesztés és javítás történt nemcsak a grafikus kezelőfelületen, hanem a neurális hálókészítés beállításainak lehetőségeiben, valamint a hálóképzés sebességében és pontosságában is. Segítségével képfelismerő algoritmus készíthető a készülék kamerája, vagy a meglévő fényképek használatával. Képes hangok és dallamok felismerésére is, mikrofon vagy hangfájlok alkalmazásával, továbbá képessé tehető az emberi mozdulatok és testtartások felismerésére valós időben webkamera segítségével.²³

A neurális háló képzése három egyszerű lépésben történik, amelyek a következők: adatgyűjtés és osztályozás, modellképzés és az elkészült modell exportálása, felhasználása. Adatgyűjtés során egy megfelelő mennyiségű és minőségű adattal feltöltött halmazt kell létrehozni (ezek a kívánt tevékenység alapján állhatnak kép, hang vagy videófájlokból). A feltöltött elemeket csoportosítani szükséges, ami a példák osztályokba vagy kategóriákba rendezését jelenti, ezeket kell majd a neurális hálónak megtanulnia. Az adatbázis feltöltése történhet a webkamerával valós időben készített képekből, a Google Driveről,²⁴ illetve adattároló eszközökről ‘png’, ‘jpeg’, vagy ‘bmp’ kiterjesztéssel (11. ábra). A képeket a program automatikusan négyzet alakúra vágja ki. Kétféle feltöltési lehetőség közül lehet választani, 240 × 240 pixel méretű színes képek vagy 96 × 96 pixel méretű fekete-fehér képek. Természetesen a jobb képminőség

²³ Teachable Machine [é. n.]

²⁴ A Google Drive egy fájlok tárolását, szinkronizálását és szerkesztését lehetővé tevő szolgáltatás, amelyhez minden Google fiókkal rendelkező felhasználó számára 15 GB felhőalapú tárhelyet biztosít a cég.

és színes adathalmaz jobb felismerhetőséget is eredményez, így ajánlott ezzel a módszerrel használni az alkalmazást.²⁵



11. ábra: Adatok osztályokba rendezése (Teachable Machine [é. n.]

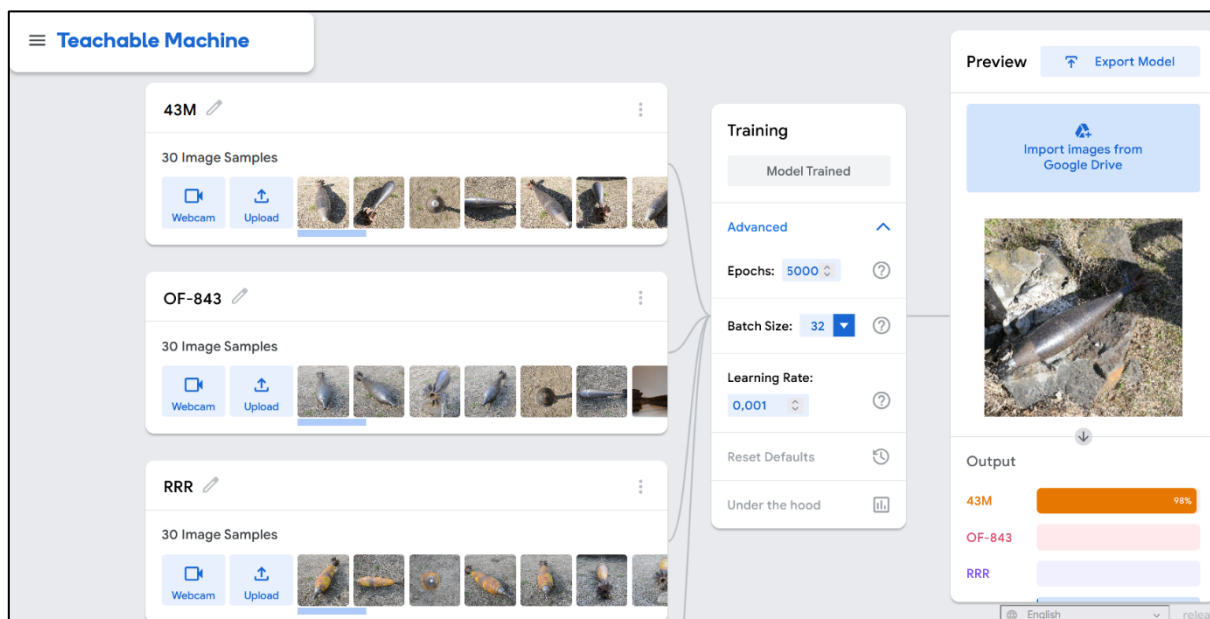
Modellképzés során (12. ábra) a tanítás gombra kattintva, megkezdődik a tanulási folyamat, amelyhez a rendszer nem a saját számítógép erőforrásait használja fel, hanem a Google szervereit, így minimalizálva a felhasználó hardverigényét és maximalizálva a feladat elvégzésének sebességét. A tanítás során két változó értékét lehet állítani, az epoch²⁶ és batch size²⁷ változókat. A rövid időn belül elkészülő modell azonnal tesztelhetővé válik, így hamar megállapítható, hogy a modell képes-e megfelelően besorolni az új teszt példákat. A pontosság vizsgálatához próbafájlokat lehet készíteni a webkamera segítségével vagy azok feltölthetők adattárolóról is. Az alkalmazás a tesztfájltra vetítve meghatározza, hogy az egyes csoportokkal, osztályokkal milyen százalékos arányban egyezik meg a tartalom. A modell exportálása, felhasználása során az elkészült neurális háló különböző formátumokban (Tensorflow.js, Tensorflow [Keras] és Tensorflow Lite) kimenthető az alkalmazásból (13. ábra), hogy azt a különböző projektekben lehessen alkalmazni. Egy ilyen plusz felhasználási lehetőséget biztosít a Google is, így az Android operációs rendszerekkel rendelkező telefonokra alkalmazást lehet programozni a kapott mesterséges intelligencia használatára.²⁸

²⁵ Teachable Machine [é. n.]

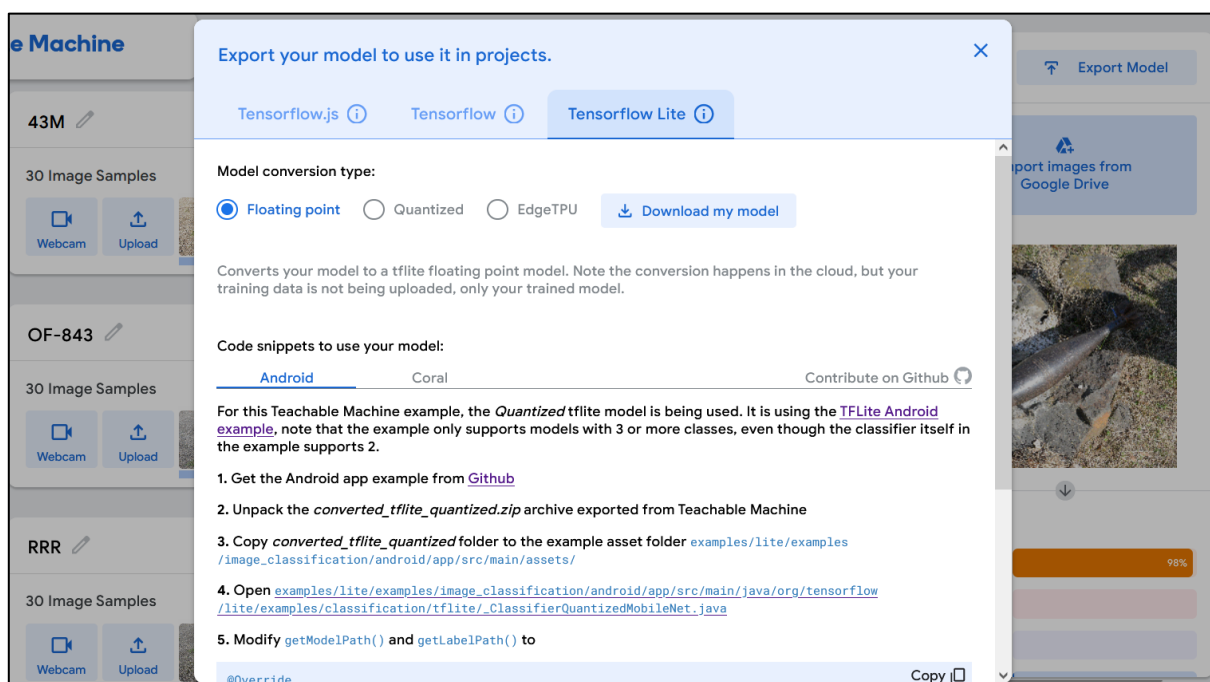
²⁶ Egy epoch a mintahalmaz egyszeri végigfuttatása a neurális hálón. Ha például ez az érték 50, akkor 50 esetben fut végig a minta feldolgozása és a súlyozás. A kis epoch szám alul tanított, a nagy epoch szám túltanított neurális hálót eredményez. De alapvetés, hogy minél nagyobb az epoch szám, várhatóan annál jobb modellt kapunk.

²⁷ A batch size lényege, hogy egy tanulási ciklusban meghatározott számú képet tanítunk be a neurális hálónak. Ha a tanulóhalmaz 80 képet tartalmaz és a batch size 16, akkor $80 \div 16 = 5$ batch-be lesz szétosztva a 80 kép. Ha mind az 5 batch átment a tanulás folyamatán, akkor kész egy epoch. Ha az 5 batch kétszer is átmegy a tanulási folyamaton, akkor kész kettő epochs és így tovább.

²⁸ Teachable Machine [é. n.]



12. ábra: Modellképzés (Teachable Machine [é. n.])



13. ábra: Modell exportálása és letöltése (Teachable Machine [é. n.])

A program megfelelő működéséhez és használatához szükség van a már sokszor említett nagyméretű képi adatbázisra, aminek kialakításához a Tűzszerész Ezred laktanyájában, a kiképző kabinetben készítettünk fényképeket. Az általam használt adatbázisról a cikksorozat következő részében esik szó.

6. Összegzés

A cikkben bemutatásra kerültek azok a matematikai és informatikai megoldások, amelyek segítségével a képelemző mesterséges intelligencia működik. Ezek a folyamatok biztosítanak lehetőséget, hogy a program, adott képet a korábban definiált csoportok valamelyikéhez sorolja. A kutatás során ehhez a feladathoz a Teachable Machine webalapú eszközt használtam, amely megfelelő hatásfokkal készített a kívánalmaknak megfelelő neurális hálókat. Az elkészült

neurális hálók exportálhatók a programból, megkönnyítve ezzel a további fejlesztés és például egy telefonos applikáció létrehozásának lehetőségét.

Ez a publikáció az Innovációs és Technológiai Minisztérium ÚNKP-21-1-I-NKE-93 kódszámú Új Nemzeti Kiválóság Programjának a Nemzeti Kutatási, Fejlesztési és Innovációs Alapból finanszírozott szakmai támogatásával készült.



INNOVÁCIÓS ÉS TECHNOLÓGIAI
MINISZTERIUM

(Folytatjuk)

Felhasznált irodalom

- Analog Image Processing vs. Digital Image Processing* [é. n.]. Online: <https://www.javatpoint.com/analog-image-processing-vs-digital-image-processing>
- BROWNLEE, Jason (2020): *How Do Convolutional Layers Work in Deep Learning Neural Networks?* Online: <https://machinelearningmastery.com/convolutional-layers-for-deep-learning-neural-networks/>
- DETTAT, Arden (2017): *Applied Deep Learning - Part 4: Convolutional Neural Networks*. Online: <https://towardsdatascience.com/applied-deep-learning-part-4-convolutional-neural-networks-584bc134c1e2>
- EMBER István (2020a): A lőszermentesítés szerepe az építőiparban. *Építőanyag*, 72(2), 59–63. Online: https://epitoanyag.org.hu/wp-content/uploads/2020/05/10.14382_epitoanyag-jsbcm.2020.9.pdf
- EMBER István (2020b): The role and the risks of explosive ordnance decontamination in Hungary. *SCIENCE & MILITARY (VEDA A VOJENSTVO)*, 16(1), 32–42. Online: http://ak.aos.sk/images/repozitar/sam/sam_1_2021/sam_1_2021_5.pdf
- Image Processing Definition, Examples and Application* [é. n.]. Online: <https://www.zr-tech.co.uk/Image-Processing/>
- KORSTANJE, Joos (2020): *What is the difference between Object Detection and Image Segmentation?* Online: <https://towardsdatascience.com/what-is-the-difference-between-object-detection-and-image-segmentation-ee746a935cc1>
- KOVÁCS Zoltán (2012): Az improvizált robbanóeszközök főbb típusai. *Műszaki Katonai Közlemény*, 22(2), 37–52. Online: https://mkk.uni-nke.hu/document/mkk-uni-nke-hu/2012_2_03%20IED-k%20f%C5%91bb%20t%C3%ADpusai%20-%20Kov%C3%A1cs%20Z.pdf
- Max Pooling* [é. n.]. Online: <https://paperswithcode.com/method/max-pooling>
- NÉMETH András – VIRÁGH Krisztián (2022): Mesterséges intelligencia és haderő – Polgári alkalmazási lehetőségek V. rész. *HADITECHNIKA*, 56(5), 2–7.
- NÉMETH András – VIRÁGH Krisztián (2023): Mesterséges intelligencia és haderő – Katonai alkalmazási lehetőségek VII. rész. *HADITECHNIKA*, 57(1), 2–6.
- Objektum felismerő (képfelismerő) AI megoldásunk* (2019). Online: <https://www.regens.com/hu/-/objektum-felismeres-kepfelismeres-mesterseges-intelligenciaval-bemutato-video>
- SHARMA, Pulkit (2019): *Image Classification vs. Object Detection vs. Image Segmentation*. Online: <https://medium.com/analytics-vidhya/image-classification-vs-object-detection-vs-image-segmentation-f36db85fe81>
- Teachable Machine* [é. n.]. Online: <https://teachablemachine.withgoogle.com/>
- TOMOLYA János – PADÁNYI József (2012): A terrorizmus jelentette kihívások. *Hadtudomány*, 22(3–4), 34–67. Online: https://www.mhht.eu/hadtudomany/2012/3_4/HT_2012_3-4_Tomolya_Padanyi.pdf
- TÓTH László (2019): *Mesterséges neuronhálók és alkalmazásaik*. Online: <https://www.inf.u-szeged.hu/~tothl/ann/Neuronhalok-egyben.pdf>

What Is Deep Learning? 3 things you need to know [é. n.]. Online:
<https://www.mathworks.com/discovery/deep-learning.html>