



M Ű E G Y E T E M 1 7 8 2

Budapesti Műszaki és Gazdaságtudományi Egyetem
Villamosmérnöki és Informatikai Kar
Automatizálási és Alkalmazott Informatikai Tanszék

Valós idejű viszkozitás mérése a vegyiparban képfeldolgozás segítségével

TDK dolgozat

Készítette:

Mádi István László

Meglécz Máté

Konzulens:

Dr. Csorba Kristóf

2022

Kivonat

Vegyipari gyártási folyamatokban fontos annak a biztosítása, hogy a folyadékok viszkozitása megfelelő értékű legyen. Mindezért gyártás során kiemelt fontosságú a legyártott termék viszkozitásának ellenőrzése. A jelenlegi, megszokott módszerek legnagyobb hátrányai, hogy igen időigényesek és csak a gyártást követően képesek a termék minőségének változását jelezni. Ideális lenne, ha a gyártási folyamatban minél hamarabb detektálható lenne a viszkozitás megváltozása. Erre kíván megoldást találni a Pharmatech labor egyik kutatási témája, ami a viszkozitás videókép alapú meghatározása, mely könnyen integrálható mérési módszer lenne egy folyamatos gyártási megoldásba. A dolgozat célja egy csapatban kialakított mérési megoldás fejlesztése és bemutatása.

A viszkozitás becslés egy ipari kamera előtt lecsepegő folyadékról készített videofelvétel feldolgozásával történik. A videók előfeldolgozáson mennek keresztül a szükséges jellemzők kinyerésének érdekében. A megfigyelt cseppek viszkozitás becslése a gépi tanulás eszköztárával valósul meg a kinyert adatok felhasználásával.

A dolgozatban bemutatásra kerül az előfeldolgozás valós idejű végrehajtásának megvalósítása, a becslési feladat több különböző módszerrel történő megoldása és az így kapott eredmények értékelése.

Abstract

During chemical production processes one of the main concerns is guaranteeing the correct value of the products' viscosity. For the above mentioned reason the monitoring of the product's viscosity has a great significance. The current conventional measurement procedures' outstanding shortcomings are that they can only be executed after the production process has finished and they are time consuming as well. It would be ideal to be able to detect viscosity changes as early as possible in the production process. One of the research topics of the Pharmatech lab is to find a solution to this problem, which is the video-based determination of viscosity, a measurement method that can be easily integrated into a continuous production solution. The aim of our thesis is to develop and present a measurement solution for the above described production drawbacks.

The viscosity estimation is performed by processing a video recording of a liquid dripping in front of an industrial camera. The videos are pre-processed to extract the required characteristics. The viscosity estimation of the observed droplets is performed using machine learning tools using the extracted data.

The paper presents the real-time implementation of the pre-processing, the solution of the estimation problem using several different methods and the evaluation of the obtained results.

Tartalomjegyzék

Kivonat.....	1
Abstract.....	2
Bevezetés	5
1 Jelenlegi megoldások ismertetése	6
1.1 Jelenleg ipari körülmények között használatos módszerek.....	6
1.1.1 Kapilláris viszkoziméter	6
1.1.2 Eső golyós viszkoziméter	6
1.1.3 Rotációs viszkoziméterek	7
1.2 Jelenlegi mérőeszközök értékelése, saját mérési módszer és annak előnyeinek bemutatása	7
2 Mérési módszer ismertetése	9
2.1 A gyártott folyadékot modellező folyadék előállítása.....	10
2.2 Viskozitás mérés.....	11
3 Korábbi eredmények ismertetése	12
3.1 Korábbi eredmények	12
3.2 Továbbfejlesztési célok	12
4 Képfeldolgozás.....	14
4.1 Új csepp tulajdonságok	14
4.1.1 Képkockánkénti új paraméterek	14
4.1.2 Egy csepp összefoglaló paraméterei	15
4.2 Ipari körülményekhez való alkalmazkodás	17
4.3 A képfeldolgozás sebességének bemutatása	18
4.3.1 A jelenlegi szoftver sebességének demonstrációja	18
4.3.2 Az új paraméterek és robusztusabb előfeldolgozás implementációját követő eredmények	20
4.4 Futás idő gyorsítása, valós idejű becslés eléréséhez.	21
4.4.1 Párhuzamosítás processzoron	21
4.4.2 GPU felhasználása futási idő csökkentésére.....	22
4.4.3 Az egyes algoritmusok párhuzamosítása	23

5	Adathalmaz bemutatása, előkészítése	25
5.1	Főkomponens analízis	25
5.2	Viszkozítások	26
5.3	Tisztítás	26
6	Becslés	28
6.1	Osztályozás.....	28
6.1.1	Adathalmaz	28
6.1.2	Multinomial logistic regression	28
6.1.3	Módszer értékelése.....	28
6.2	Regresszió	29
6.2.1	Adathalmaz	29
6.2.2	Háló architektúrája.....	30
6.2.3	Tanítás.....	32
6.2.4	Módszer értékelése.....	33
	Összefoglalás.....	35
	Köszönetnyilvánítás	36
	Irodalomjegyzék.....	37

Bevezetés

Dolgozatunk a Pharmatech laboratórium aktív kutatási témájának keretén belül egy gyártási folyamatba integrálható, képfeldolgozáson és gépi tanuláson alapuló viszkozitás mérési megoldás kidolgozásának bemutatását célozza meg. Egy ilyen eljárás a jelenlegi módszerekkel ellentétben a gyártási folyamat közben képes lenne a lehető legkorábban detektálni a viszkozitás érték megváltozását, így csökkentve a selejtes termékek mennyiségét. Megoldásunkban a laboratóriumban korábban elkészített megoldásokból indultunk ki és azokat fejlesztettük tovább [1]. Fejlesztéseink céljaul az eddigi megoldás valós időjüregyorsítását, új mérhető paraméterekkel való kiegészítését és viszkozitás becslési módszer megalkotását tűztük ki.

A dolgozatban először bemutatjuk az elterjedten használt viszkozitás mérési technikákat, azoknak az előnyeit és a hátrányait. Ismertetjük a mérési beállításunkat, és az ezzel végzett méréseket. Röviden értékeljük a korábban mérésre megalkotott programot és osztályozási kísérletet. Továbbiakban részletezzük a képfeldolgozó program gyorsításához rendelkezésre álló módszereket, ismertetjük az általunk kiválasztott technikákkal elért eredményeket. Bemutatjuk a tanításhoz felhasznált adathalmazt, majd a hibás adatok kiszűrésének módszerét. Végezetül kísérletet teszünk osztályozásra és viszkozitás becslésre a kinyert adatok felhasználásával.

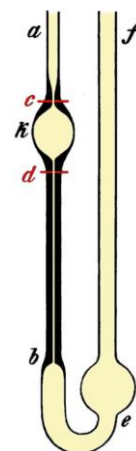
1 Jelenlegi megoldások ismertetése

1.1 Jelenleg ipari körülmények között használatos módszerek

A továbbiakban néhány elterjedt módszert mutatunk be folyadékok viszkozitás mérésére. [2]

1.1.1 Kapilláris viszkoziméter

Az egyik legerjedtebb és legolcsóbb módja a viszkozitás mérésének a kapilláris viszkoziméterek használata. Ez az eszköz leggyakrabban egy U alakú üveg csőből áll. Ezzel az eszközzel történő mérési eljárás során az ábrán *e* és *d* betűkkel jelölt piros vonal közötti átjutási sebességét mérik meg a folyadéknak, ebből az értékből és a mérőeszközre jellemző konstans felhasználásával állapítják meg a kinematikai viszkozitását. Elsődlegesen alacsony és közepes mértékű viszkozitással rendelkező folyadékok vizsgálatára alkalmas.



1. ábra:
kapilláris
viszkoziméter

1.1.2 Eső golyós viszkoziméter



2. ábra: eső golyós viszkoziméter

Eső golyós viszkoziméter használata során egy kis méretű fém golyót engednek el egy kapillárisban. Ennek következtében a golyó egyenes vonalú egyenletes mozgásának ideje kerül megmérésre. Az így kapott időből, valamint a golyó, a kapilláris és a folyadék fizikai tulajdonságainak ismeretében a Stokes törvény [3] segítségével lehet kiszámítani a folyadék viszkozitását. (Különböző fajtájú berendezések érhetőek el jelenleg, a ábrán egy modernebb, félig automatizált eszköz látható.) A folyadék viszkozitásának növekedésével a méréshez szükséges időtartam is jelentős mértékben növekszik.

1.1.2.1 Stokes törvény:

$$F_d = 6\pi\mu Rv$$

Ahol:

- F_d : a golyó és a folyadék közötti súrlódási ellenállás.
- μ : a dinamikai viszkozitás.
- R : a golyó sugara.
- v : a golyóhoz viszonyított áramlási sebesség.

1.1.3 Rotációs viszkoziméterek

Többféle rotációs viszkozitás mérési eljárás létezik [4]. Ezen technikák alapja a mért folyadékban elhelyezett eszköz forgatása során fellépő nyomaték mérése. Az eszköz és a folyadék fizikai tulajdonságainak ismeretében kerül kiszámításra viszkozitás mértéke. Legelterjedtebb rotációs viszkoziméterek: *“Cup and bob viscometer”* és a *“Cone and plate”*. Mindkét eszköz a fent leírt módszeren alapszik. Előnye az előzőekben bemutatott technikákkal szemben, hogy kevésbé időigényes a mérési folyamat azonban nagyobb mennyiségű folyadék minta szükséges hozzá.

1.2 Jelenlegi mérőeszközök értékelése, saját mérési módszer és annak előnyeinek bemutatása

A fent bemutatott három elterjedt mérési módszer mind hasonló, de különböző mértékben fellelhető hátrányokkal rendelkeznek. Mindegyik módszer rendkívül időigényes és ezen mérések elvégzése csak az adott folyadék gyártását követően lehetséges. Az utóbbi talán a legjelentősebb probléma. Mivel mire kiderül a selejtes termékekről, hogy nem megfelelő minőségűek már nagyobb mennyiségben le lettek gyártva, ezzel anyagi és időbeli kárt okozva a gyártó vállalatnak, valamint a környezetet is terheli a selejtes termékek gyártása. Ezen felül minden esetben szükséges emberi közreműködés, ami növeli a mérési hiba előfordulásának valószínűségét.

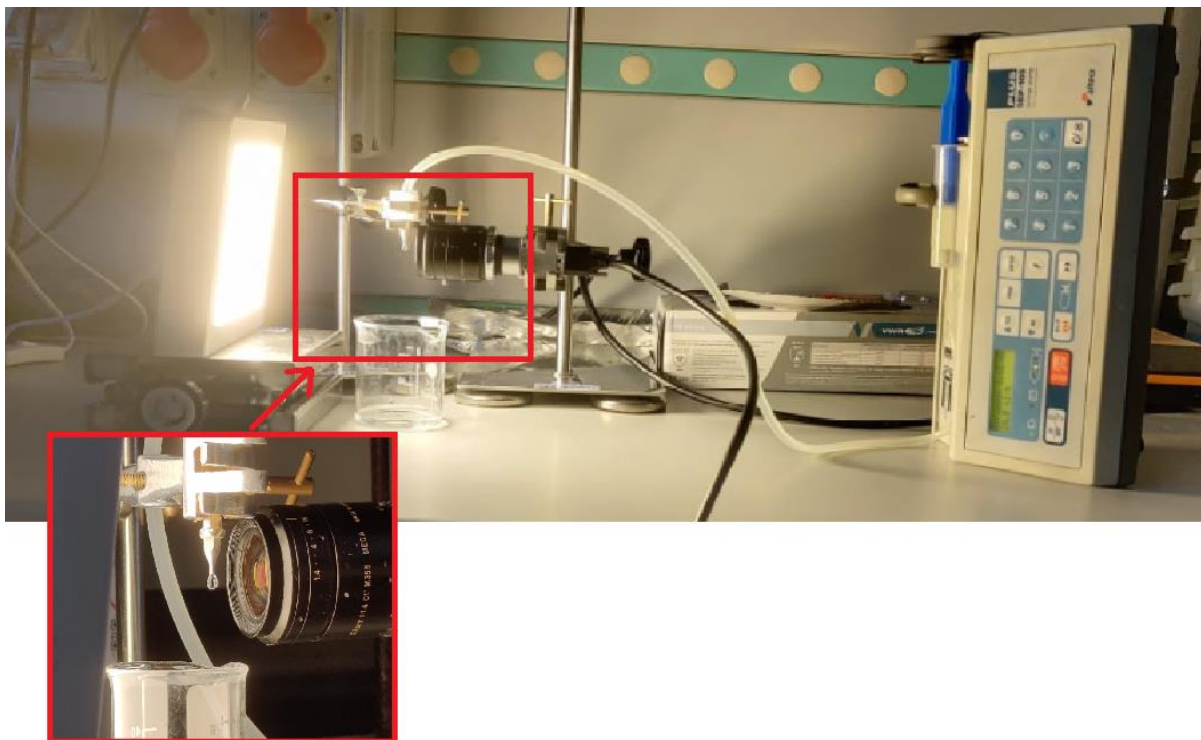
Ezen hátrányok kiküszöbölését magába foglaló mérési módszer kifejlesztését céloztuk meg, mely egy összetett gyártási folyamatba integrálható és valós idejű viszkozitás becslést tesz lehetővé.

2 Mérési módszer ismertetése

A gyártás során a gyártott folyadékot egy csepegtetőn keresztül engedünk lecsöpögni, amiről egy viszonylag alacsony felbontású, de magas képkockasebességű (FPS) videó előállítására alkalmas ipari kamerával készítünk felvételt. Kiemelten fontos, hogy jól megvilágított körülmények között történjen a felvétel, hogy a cseppek körvonala élesen elkülönüljön a háttértől. Az így előállított felvételeket képkockánként dolgozzuk fel.

A feldolgozási folyamat több fázisból tevődik össze. Először is az ipari körülményekhez való alkalmazkodás miatt előtisztításon megy át minden egyes képkocka. Ezt követően kerül sor magára a feldolgozásra, ahol a képkockákból egyesével, valamint az egy lecseppenéshez tartozó összes képkockából kinyerésre kerülnek a tanításhoz szükséges adatok. Ezt követően végrehajtunk egy szűrést, hogy a hibás cseppeket eltávolítsuk az adathalmazokból. Végezetül egy már betanított modellel végrehajtjuk a becslést.

A mérési módszerünk egyedisége miatt nem rendelkezünk már előre elkészített tanító halmazzal vagy előre betanított hálóval ezért szükséges volt saját tanító halmaz előállítására. A tanításhoz szükséges felvételek elkészítéséhez a Pharmatech laboratóriumában létrehoztunk a fentiekben leírt mérési módszerhez nagyon hasonló környezetet.



3. ábra: mérőberendezés

Ahol állványok segítségével rögzítettük a csepegtetőt, az ipari kamerát és a fényforrást. Valamint a gyártott terméket modellező különböző összetételű folyadékokat kevertünk ki és egy automatizált fecskendő segítségével csepegtettünk. Automatizált fecskendő használatára azért volt szükség, hogy majd a gyárban alkalmazott módszerhez hasonlóan állandó nyomás legyen a fecskendőt és a csepegtetőt összekötő csőben. Ellenkező esetben nem lenne stabilan becsülhető a viszkozitás mivel a nyomáskülönbség befolyásolná a lecseppenési sebességet.

A folyadékok cseréje során a csőbe levegő tud kerülni, az így keletkezett buborékok laboratóriumi mérés során és ipari körülmények között is problémát okozhatnak. A nyomáskülönbség mellett a cseppekben megjelenő buborékok is meghosszabbítják a csepp fejlődési fázisát, ezzel mérési hibát eredményezve. Ezért ezen hibák kiszűrése is szükséges egy megbízható megoldás létrehozása során.

2.1 A gyártott folyadékot modellező folyadék előállítása



4. ábra: telített oldat előállítása

A terméket szimuláló folyadék előállítása során Polyvinylpyrrolidone-t (PVP) [5] használtunk mivel jól oldódik vízben és az ezzel az anyaggal készített telített oldat nagy viszkozitású. Ennek köszönhetően lehetőségünk van a nagyon magas és a nagyon alacsony viszkozitású folyadékok előállítására is víz adagolásával.

A tanító adat létrehozásához szükséges oldatok előállításához első lépésként egy majdnem telített oldatot állítottunk elő a PVP folyamatos adagolásával és keverésével. Ezt követően a táblázatban látható összetételű fokozatosan híguló oldatokat hoztuk létre.

KEVERÉSI ARÁNY (PVP/ DESZTILLÁLT VÍZ)	PVP MENNYISÉG (ml)	DESZTILLÁLT VÍZ MENNYISÉG (ml)
0 / 7	0.0	300.0
1 / 7	50.0	300.0
2 / 7	50.0	125.0
3 / 7	50.0	66.0
4 / 7	50.0	37.5
5 / 7	50.0	20.0
6 / 7	50.0	8.33
7 / 7	50.0	0.0

1. táblázat: a minták keverési arányai



5. ábra: az előállított minták

2.2 Viszkozitás mérés

Mivel regressziós megoldással is kísérletezünk, szükséges volt az elkészített folyadékok viszkozitásának megmérése is a neurális háló tanításához. Ehhez a fentiekben ismertetett hagyományos viszkozitás mérési módszerek egyikével, egy eső golyós viszkoziméterrel megmértük egyenként az elkészített oldatok viszkozitását. Ezen mérés végrehajtása során megtapasztaltuk, hogy milyen időigényes is ez a fajta eljárás és hogy nagyobb viszkozitású folyadékok méréséhez nagyobb kapillárist és golyót kell alkalmazni, ami rontja a mérési pontosságot. Az eredeti folyadékok, amikről a felvételek készültek, a viszkozitás mérés időpontjában már nem álltak rendelkezésünkre, így a különböző keverési arányokkal rendelkező mintákat újra el kellett készítenünk.

3 Korábbi eredmények ismertetése

3.1 Korábbi eredmények

A Pharmatech laboratóriumnak ez egy folyamatban lévő kutatási témája ezért szükséges a korábbi eredmények bemutatása. Rendelkezésünkre állt egy korábbi, szakdolgozat keretén belül megvalósított képfeldolgozó program, amely képes volt a cseppekről elkészített képkockákból a csepp bizonyos paramétereinek mérésére. Jó minőségű videófelvevővel megbízhatóan, stabilan képes az adatok kinyerésére. Viszont a program futási ideje igen hosszú volt, a valós időtől jelentősen lemaradva volt csak képes feldolgozni a videó anyagokat.

A program egy .NET framework-kel C# nyelven lett fejlesztve, az OpenCV [6] nyílt forráskódú képfeldolgozó könyvtár egy C#-os OpenCVsharp [7] nevű wrapper könyvtárának felhasználásával.

A szakdolgozat keretein belül kísérlet történt egy konvolúciós neurális háló alapú osztályozásra a cseppekről készült képek alapján, ez azonban nem járt különösebb sikerrel, 25% százalékos körüli átlagos pontosságot ért el. Ennek a megoldásnak egyik legnagyobb limitációja volt a szükséges erőforrások hiánya. A cseppek képeinek felhasználásának egyik legnagyobb hátránya, hogy azokon a képeken nem láthatóak jelentős különbségek, így a konvolúciós rétegeknek nehezebbé eshet olyan jellemvonásokat találni, amik megkülönböztetik a különböző folyadékokból származó cseppeket, ezért is gondoltuk úgy, hogy más irányból közelítjük meg a becslési feladatot.

3.2 Továbbfejlesztési célok

A fentiekben bemutatott szoftverből indultunk ki és az alábbiakban ismertetett célokat tűztük ki:

Az ipari körülmények közötti alkalmazhatóság eléréséhez szükséges, hogy a program képes legyen az elvártaknak megfelelően működni az ideálistól eltérő körülmények között is. Ezért szükséges a kép előfeldolgozása, hogy az ipari körülmények között előforduló problémák hatását minimalizáljuk a lehető legpontosabb mérés érdekében.

A képfeldolgozási folyamat közben kinyert paraméterek bővítése a predikció pontosságának növelésének érdekében. Valamint az eddigiektől eltérő típusú paraméterek megalkotása, melyek nem csak egy adott képkockán szereplő csepp jellemzőit képesek reprezentálni, hanem az időben folytonos cseppenési folyamat tulajdonságait is.

A valós idejű képfeldolgozás eléréséhez használható módszerek feltérképezése és ezekkel a rendelkezésre álló szoftver továbbfejlesztése.

A képfeldolgozással létrehozott paraméterekkel különböző gépi tanulási eljárások kipróbálása és ezen eljárások optimalizálása.

4 Képfeldolgozás

4.1 Új csepp tulajdonságok

4.1.1 Képkockánkénti új paraméterek

A cseppek pontosabb leírásának érdekében, a korábban implementált paramétereket, kiegészítettük újakkal is. Úgy véltük, hogy a cseppek alakjának a különböző görbületei rendelkezhetnek még olyan jellegzetes tulajdonságokkal, amik korábban nem kerültek lefedésre.

4.1.1.1 A csepp aljának a görbülete

Egy csepp görbületének elemzésekor, a legjellegzetesebb a kontúr “alsó részének” az alakulása az idő előrehaladtával. Ehhez a részhez tartozó kontúr részlet kezdő és végpontját a kontúr x tengelybeli minimum és maximum értékkel rendelkező pontja jelöli ki. A görbület kiszámításához szükséges harmadik pontot, a maximum y tengelybeli értékkel rendelkező pont adja. Így ennek a szakasznak a görbületét Menger görbület segítségével meg tudjuk határozni.

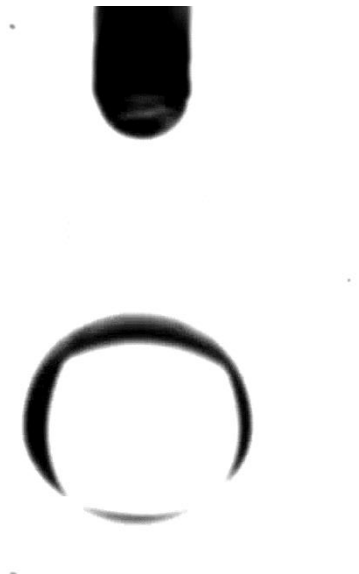
$$c(x, y, z) = \frac{4A}{|x - y||y - z||z - x|},$$

ahol x, y, z a felhasznált 3 pont, A pedig a három pont által kijelölt háromszög területe.

4.1.1.2 A visszamaradó folyadék görbülete

Amikor egy csepp elválik a csepegtetőtől jellemzően visszamarad valamennyi folyadék a csepegtető hegyén, ennek a visszamaradt folyadéknak a hossza mérésre kerül már a programban, viszont a görbülete nem. Ennek a paraméternek a kiértékelése csak akkor történik meg, amikor a program azt állapítja meg, hogy a csepegtetőtől való elválás megtörtént.

Az előbb bemutatott paraméterhez hasonlóan történik itt is a görbület számítása, a két szélső pontot a kontúr x tengelybeli minimum és maximum pontja, a harmadik pontot pedig az y tengelybeli maximum pontja jelöli ki.



6. ábra: a csepegtetőn visszamaradt folyadék

4.1.1.3 A csepp alakjának a görbülete

Egy kontúrral rendelkező alakot képeken lehetséges a kontúr görbületével is leírni. Ehhez először normalizálni kell a kontúr hosszát, majd ezen egy 3 hosszúságú ablak végig tolásával lehetséges kiszámolni a kontúr részletekhez tartozó görbületet.

Jelen esetben a csepp kontúrját 16 pontra normalizáltuk és a görbületet az alábbi képlettel számítottuk ki, ahol a pontok koordinátáinak véges térbeli deriváltját a 3 pont segítségével lehetséges megállapítani.

$$k = \frac{x'y'' - y'x''}{(x'^2 + y'^2)^{\frac{3}{2}}}, \text{ ahol } k \text{ a görbület és } x, y \text{ a koordináták}$$

4.1.2 Egy csepp összefoglaló paraméterei

4.1.2.1 Lecseppenési idő

Minden egyes csepről megállapítjuk, hogy a csepp fejlődési folyamatának kezdetétől egészen a lecseppenés végéig mennyi idő telik el.

4.1.2.2 Lecseppenés előtti állapot

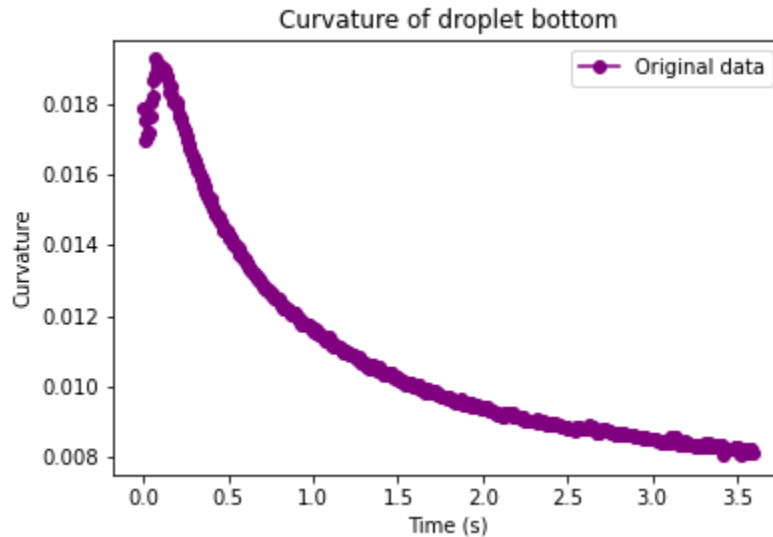
A csepp fejlődési folyamatának legvégén, mielőtt megszűnik a kapcsolat a cseppentő és a csepp közötti állapotának számos paraméterét eltároljuk. A lecseppenést megelőző pillanatbeli hosszúságot és maximális szélességet. A lecseppenés megelőző kerületet és területet. A csepp tömegközéppontjának cseppentőhöz viszonyított x és y koordinátáját. A cseppbe illeszthető deltoid alsó és felső háromszögének területét és a deltoid oldalai által bezárt szöget. Valamint a csepp aljának görbületét.

4.1.2.3 A cseppentőn visszamaradó folyadék állapota

Miután a csepp elválna a cseppentőtől egy kicsi, de nem elhanyagolható folyadékmennyiség fennmarad a cseppentőn. Ennek a folyadék mennyiségnek számos paraméterét kinyertük és rögzítettük. A rögzített paraméterek a fennmaradó folyadék hosszának és görbületének a minimum és maximum értékei.

4.1.2.4 Maximum érték és időpontja

A cseppek átmérőjének és göbületének az időben ábrázolt függvényei tapasztalataink szerint rendelkeztek olyan maximum értékekkel, amik nem a nulladik időpillanathoz köthetőek, így ezeknek a maximum értékét és azoknak az időpontjait külön eltároljuk.



7. ábra: a csepp alsó részének a görbülete az idő függvényében

4.1.2.5 Görbeillesztés

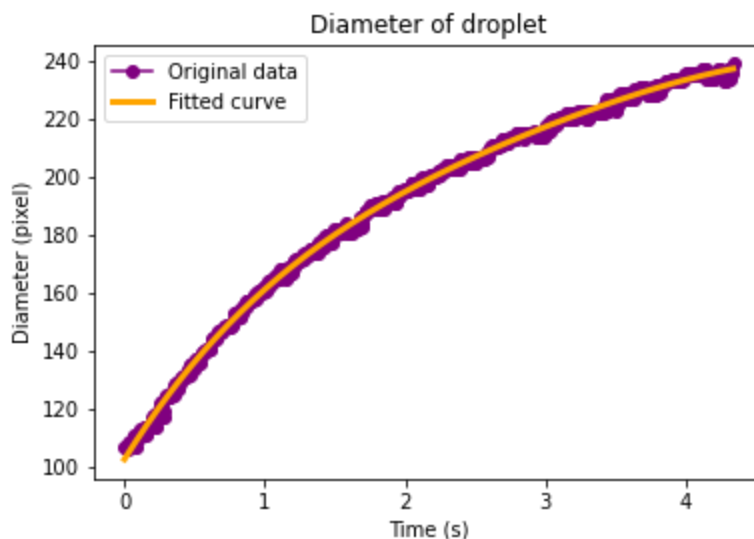
A cseppek hosszúságának és átmérőjének az időbeli alakulására görbeillesztést alkalmaztunk negyedfokú függvénnyel, majd ezeknek az együtthatóit eltároljuk. Ez lehetőséget ad a csepp időbeli alakulásának egyszerűbb leírására.

4.1.2.6 Inflexiós pontok száma

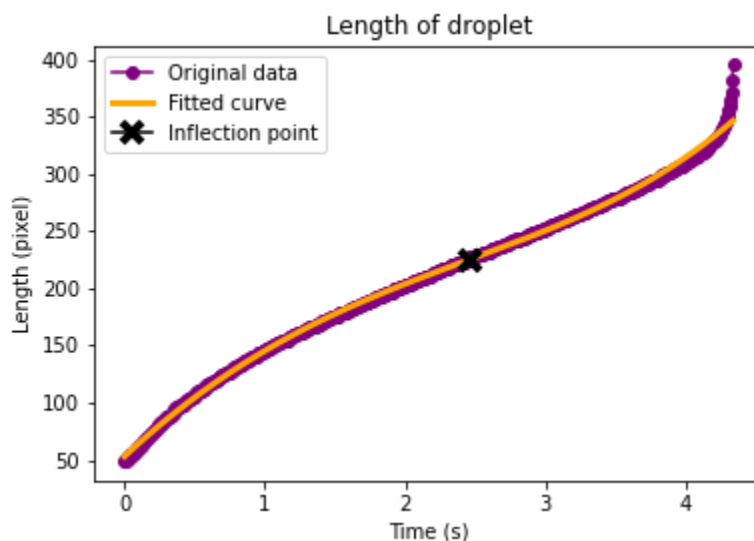
A görbeillesztés által meghatározott együtthatók alapján, meg lehet találni az inflexiós pontokat. Az egyes cseppekre vonatkozóan eltároljuk az inflexiós pontok számát és azoknak az időpontjait.

4.1.2.7 Függvény "konvexitás"

Az előzőleg bemutatott görbeillesztés alapján meghatározott függvény az inflexiós pontok mentén részekre osztható, majd ezeknek a részeknek meghatározható a konvexitása, ehhez értékként, az adott függvény részlet második deriváltjának az átlagát rendeltük.



8. ábra: a csepp átmérője az idő függvényében és az illesztett függvénye

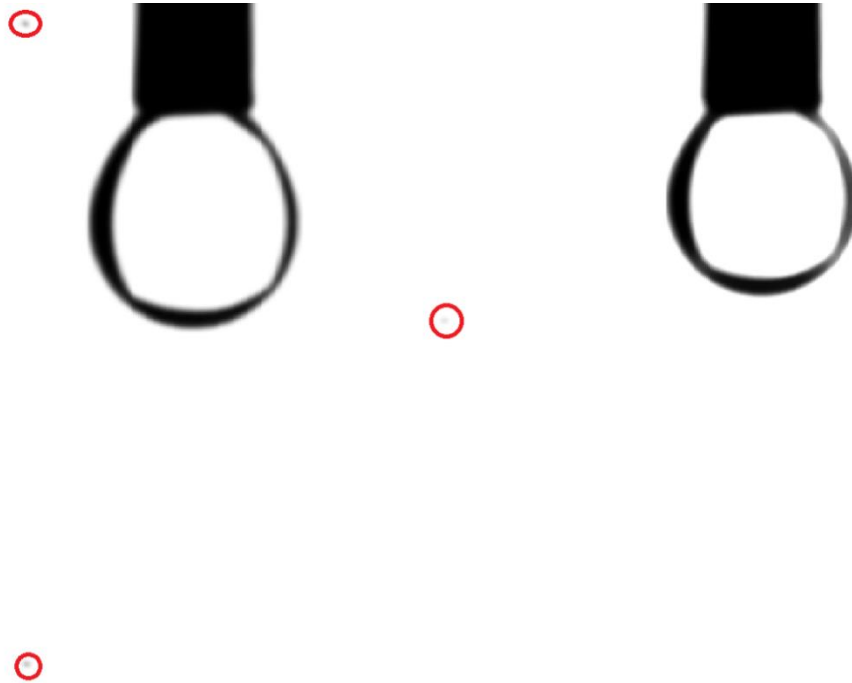


9. ábra: a csepp hossza az idő függvényében, illesztett függvénye és az inflexiós pontja

4.2 Ipari körülményekhez való alkalmazkodás

A program kiinduló állapotában szerepelt a képkocka homályosítása, elforgatása és az éldetekció. A korábban alkalmazott homályosítási algoritmus azonban nem volt elégséges kevésbé jó minőségű felvételek esetén. Az ábra bal oldalán látható, hogy enyhén koszos kamera kép esetén megjelennek foltok, amelyeket az éldetekciós algoritmus érzékel ezzel

hibás képfeldolgozást eredményezve. Ezért szükséges volt a homályosítási algoritmus módosításár. Medián filter [8] alkalmazására átállva megszűntek az ilyen jellegű problémák.



10. ábra: bal oldalt a Gaussian blur-rel ellátott kép, jobb oldalt a Median filter-rel ellátott kép

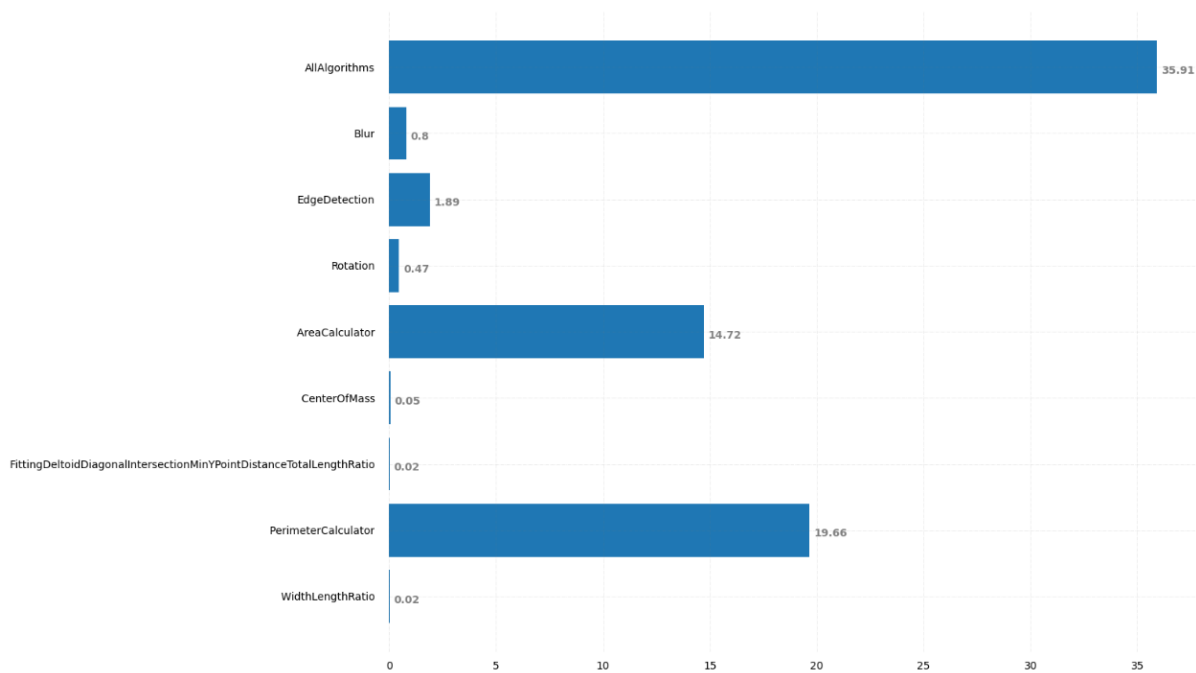
4.3 A képfeldolgozás sebességének bemutatása

A következőkben ismertetett tesztek, eredmények egy RTX 3060 grafikus feldolgozó egységgel és egy AMD Ryzen 5 3600XT processzorral felszerelt számítógépen kerültek kiértékelésre. A számítógép hardverének fejlesztésével is lehet jelentősen javítani az elért eredményeket. A feldolgozott videók 150 képkockasebességűek voltak és 720-szor 536 pixel méretű képkockákból álltak.

4.3.1 A jelenlegi szoftver sebességének demonstrációja

Az ábrán látható, hogy a feldolgozás egyes lépései átlagosan milyen hosszúságú idő (milliszekundum) alatt kerültek kiértékelésre. A feldolgozás lépéseit két fő csoportra

oszthatjuk, az előfeldolgozásra és az algoritmusok futtatására.



11. ábra: egyes algoritmusok futási ideje (ms)

Az előfeldolgozás a program jelenlegi állapotában három lépésből tevődik össze: homályosítás (Blur), ami gaussian blur-rel volt megvalósítva az esetleges foltok megszüntetésére, forgatás ha a kamera nem álló helyzetben került elhelyezésre és az éldetekció, amivel kinyerjük a cseppek körvonalát. A körvonalat használjuk a további paraméterek kiszámítása során. Az összes előfeldolgozási lépés összesen 3.16 milliszekundumban telik.

A négy paraméter kinyerő algoritmus összesen 35.91 milliszekundumban telik összesen melyből a csepp tömegközéppontjának és a cseppbe illeszhető deltoid tulajdonságainak meghatározása elhanyagolhatóan kis mértékű időt vesz igénybe. A csepp területének és kerületének meghatározása azonban jelentős számítási igénnyel rendelkező feladat, 14.72 és 19.66 milliszekundum mire egy képkockára kiszámításra kerül ezen két paraméter.

A fent leírt lépésekhez szükséges idő együttesen adja egy képkocka feldolgozása során eltelő időt. 3.16 milliszekundum az előfeldolgozás és 35.91 az algoritmusok végrehajtása, ami összesen azt jelenti, hogy egy képkocka feldolgozása 39.07 milliszekundumba telik. Ebből következik, hogy egy másodperc alatt 25.6 képkockát tudott feldolgozni a program.

4.3.2 Az új paraméterek és robusztusabb előfeldolgozás implementációját követő eredmények

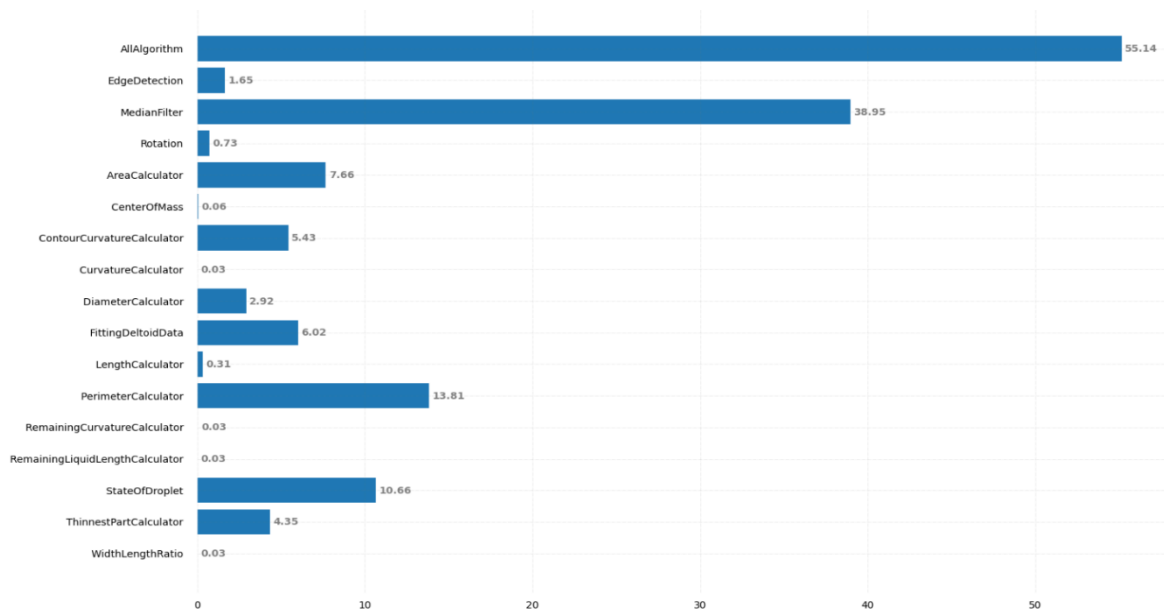
4.3.2.1 Az előfeldolgozás módosítása

Az új, kevésbé tiszta videófelvevételekhez való alkalmazkodás miatt módosításra került a homályosító eljárás, mely jelentős mértékben meghosszabbította a futási időt. A medián filter alkalmazásához szükséges idő 37.16 milliszekundum. Így egy képkocka teljes feldolgozása 73.07 milliszekundum, ami képkockasebesség tekintetében 13.69 képkocka/másodperc-es sebességet jelent.

4.3.2.2 Új paraméterek hatása

A fentiekben ismertetett új paraméterek előállításához szükséges algoritmusokkal való bővítést követően a feldolgozó program sebessége tovább lassult. Összesen egy képkocka feldolgozásához szükséges idő 96.5 milliszekundumra nőtt, ami másodpercenként átlagosan 10 darab feldolgozott képkockát eredményezett. Mivel egy csepp az oldat arányának és az oldott anyagtól függően körülbelül 4-5 másodpercig fejlődik és ereszkedik le ez a feldolgozási sebesség azt jelenti, hogy cseppenként 40-50 feldolgozott kép alapján kellene végrehajtani a becslést, ami nem bizonyult elegendőnek.

Tíz képkocka feldolgozása másodpercenként nem elégséges valós idejű becslés megvalósításához. A helyzetet tovább rontja, hogy az algoritmusok futási ideje a csepp méretének növekedésével (hosszabb körvonalak elemzésével) jelentősen magasabb lehet, mint az átlagos érték. Ez azonban nem kardinális probléma mivel a jelenlegi megoldás lecseppenést követően hajtja végre a becslést a nagyobb méretű cseppeket ábrázoló képkockák feldolgozásának ideje behozhatóvá válik a kisebb cseppek feldolgozása alatt. Ez egy elhanyagolható mértékű késleltetést ad minden egyes becsült viszkozitás értékhez.



12. ábra: egyes algoritmusok futási ideje (ms), az új paraméterekkel bővítve

4.4 Futás idő gyorsítása, valós idejű becslés eléréséhez.

Mivel a jelenleg megszokott és a fentiekben már ismertetett hagyományos viszkozitás mérési eljárások egyik legnagyobb hátránya, hogy igen lassú folyamat során képesek csak meghatározni a különböző folyadékok viszkozitását, ezért láttuk szükségességét annak, hogy a jelenlegi szoftvert ebbe az irányba fejlesszük tovább. Ez a fejlesztési irány az, amellyel a hagyományos módszerekkel szemben a legnagyobb pozitívumot tudja a mi módszerünk nyújtani.

A korábban ismertetett módon kibővített program képfeldolgozási kapacitása nem elégséges a valós idejű becslés megvalósításához. Szükséges volt ezen a téren fejleszteni a szoftverünket. Ezen fejlesztés elvégzéséhez számos keretrendszert, módszert megvizsgáltunk a legjobb megoldás kiválasztásának érdekében.

4.4.1 Párhuzamosítás processzoron

A kiinduló program teljes mértékben egy szál felhasználásával futott ezért az első logikus futási idő csökkentő megoldási ötlet a szoftver többszálúsítása volt. Itt két fő irányt határoztunk meg a különböző képkockánkénti paraméterek kinyerésének párhuzamosításának lehetőségét, valamint a kinyerési algoritmusokon belüli párhuzamosítás lehetőségét.

4.4.1.1 Algoritmusok végrehajtásának párhuzamosítása

Gondos mérlegelést követően megállapításra került, hogy mely algoritmusok azok, melyek egymástól függetlenek és ezért lehetséges a párhuzamos végrehajtásuk. A párhuzamosításhoz a .NET6 System.Threading.Tasks névtérben Található Parallel osztály [9] párhuzamos végrehajtású ciklusait használtuk. Ezzel jelentős eredményt elérve, hiszen amennyiben párhuzamosan hajtódnak végre az algoritmusok nem az egyes algoritmusok futási idejének összege adja a paraméterek kinyeréshez szükséges időt, hanem a leghosszabb ideig futó algoritmus futási ideje, amennyiben rendelkezésre áll megfelelő számú szál.

4.4.2 GPU felhasználása futási idő csökkentésére

Az Nvidia által gyártott grafikus kártyák (GPU) több ezer CUDA [10] maggal rendelkeznek, a processzoron elérhető limitált számú maggal szemben. Egy CUDA mag képességeit tekintve korlátozottabb egy processzor maghoz képest. Nagymennyiségű, jól párhuzamosítható repetitív feladat végrehajtásakor viszont jelentősen gyorsabbak. [11]

Mint már korábban ismertettük, kiindulásként a képfeldolgozáshoz az OpenCVSharp nevű OpenCV csomagoló keretrendszer (wrapper) lett használva. Habár az OpenCV-ben számos eljárás hagyományos processzor implementációja mellett CUDA implementációja [12] is megtalálható, azonban az OpenCVSharp nem támogatja azokat. Mivel az általunk az előfeldolgozás során használt eljárások mindegyike rendelkezik CUDA-s implementációval is mindenképpen ki akartuk használni ezt a lehetőséget. Ezért két lehetséges alternatívát vizsgáltunk meg saját, OpenCv keretrendszer fordítása CUDA támogatással és a szoftver újra írása C++ nyelven vagy CUDA támogatással rendelkező C# csomagoló osztály használata.

4.4.2.1 OpenCv keretrendszer fordítása CUDA támogatással

Amennyiben szeretnénk élni az OpenCv CUDA-s implementációk előnyével, szükséges saját magunknak lefordítani azt. Ez a folyamat Windows-os környezetben CMAKE keretrendszer segítségével generált Visual Studio megoldás (solution) futtatásával történik, jelentős idő alatt. Mivel az előfeldolgozás során használjuk ki az így létrehozott GPU-n futó eljárásokat ezért ennek C++ nyelven megírásával tudtuk tesztelni, hogy milyen sebességbeli különbség érhető el ennek a módszernek az alkalmazásával és hogy érdemes-e emiatt újraírni a teljes programot. Összességében elhanyagolható mértékű sebesség különbséggel rendelkezik

az így létrehozott előfeldolgozás, a következőkben bemutatott EmguCV [13] nevű OpenCV csomagoló keretrendszer felhasználáshoz képest. Ezért ezt a fejlesztési irányt elvetettük.

4.4.2.2 EmguCV

Az EmguCV egy platform független .Net nyelvekkel kompatibilis OpenCv csomagoló keretrendszer mely magába foglalja a számunkra fontos eljárások GPU-ra implementált [14] változatát is és lehetővé teszi minden OpenCV-ben található metódus hívását. Mivel ennek a megoldásnak a használatával csak a jelenlegi programon kellett módosítani, újra írni az előfeldolgozást végző modult. Ezzel lényegében a C++-os változattal megegyező sebességet elérve. Nem volt kérdéses számunkra, hogy ez lesz az optimális megoldás.

A 13. ábrán látható, hogy a Medián filter, elforgatás és éldetektlás GPU-n való végrehajtása jelentős mértékben csökkenti az előfeldolgozáshoz szükséges időt.

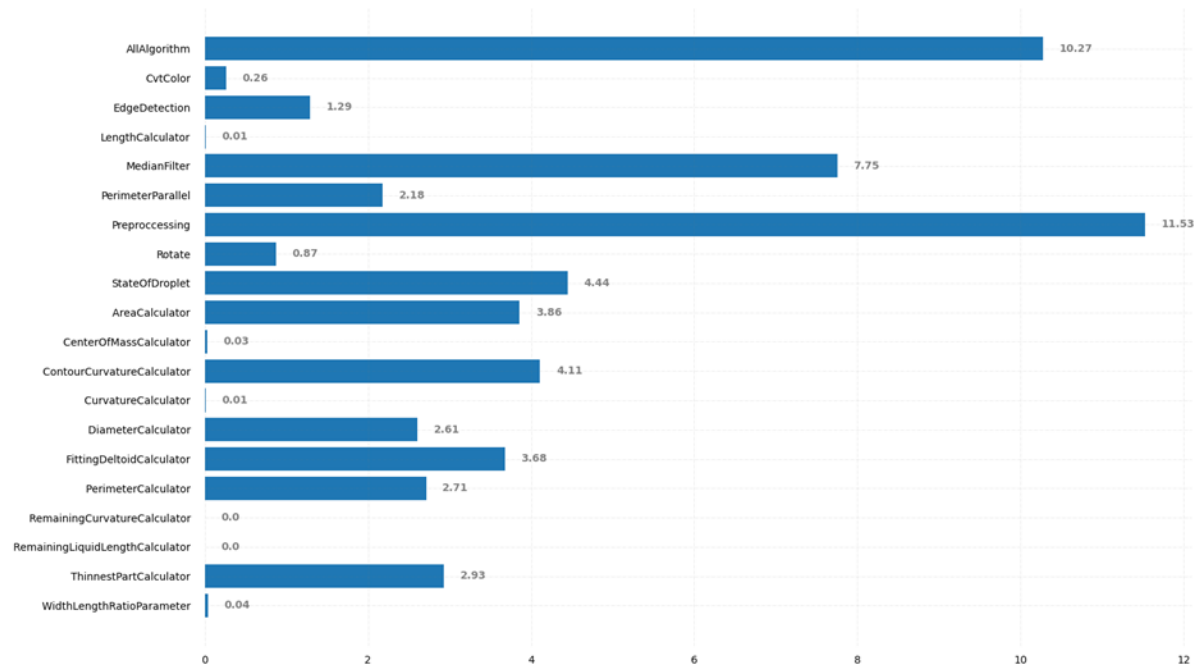
4.4.3 Az egyes algoritmusok párhuzamosítása

Az egyes algoritmusok párhuzamosításánál egy fontossági sorrendet határoztunk meg, a legtovább tartó algoritmus futásával kezdtük, mert a már korábban leírt okok miatt ez határozza meg a feldolgozás hosszát.

4.4.3.1 Kerület kiszámításának párhuzamosítása

A kerület kiszámítás az egyik legszámítás és egyben legidőigényesebb eljárás. Az éldetekcióval meghatározott kontúr pontokon pixel soronként végighaladva határozzuk meg az egyes cseppek kerületét. Mivel az ezt végrehajtó ciklusban egyik iteráció eredménye sem függ az azt megelőző vagy követő iteráció eredményétől, ezért ennek a ciklusnak a párhuzamos végrehajtásával jelentős sebességnövekedést értünk el.

A kerületszámítás végrehajtása 720 pixel sorból álló képkockán még párhuzamosítva is hosszú ideig tart a korlátos számban elérhető CPU magok miatt. GPU-ra implementálva azonban az elérhető magok száma nem jelentene korlátozó tényezőt, ezzel minimálisra csökkentve a kerület számításához szükséges időt. Ez a fejlesztés jelenleg még nem került megvalósításra, azonban további gyorsítás eléréséhez ez lenne a következő lépés.



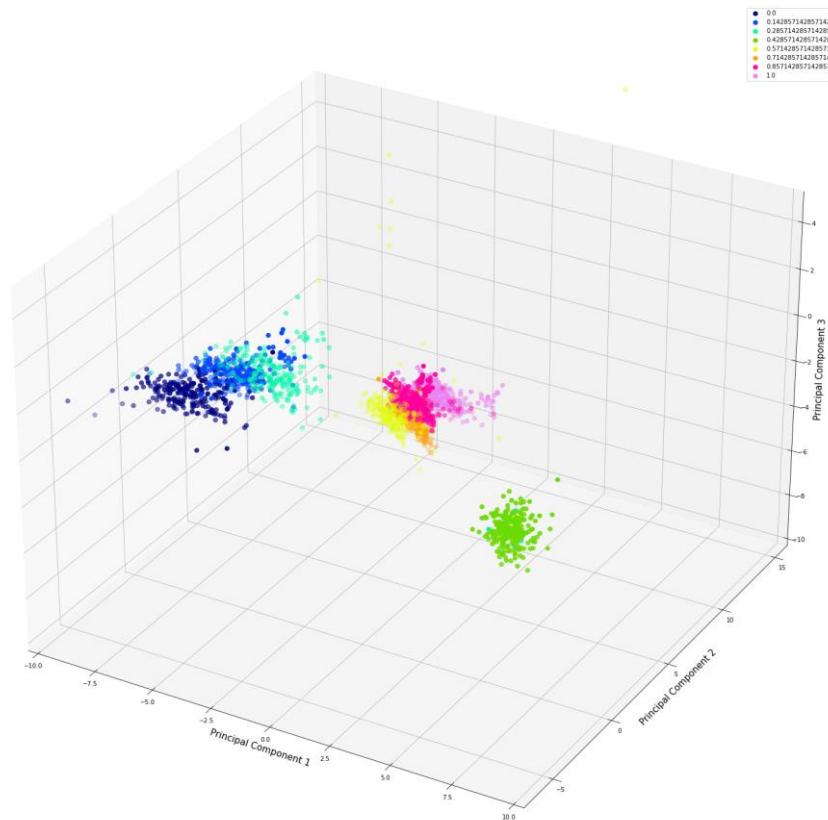
13. ábra: futásidő optimalizálást követő futási sebesség

5 Adathalmaz bemutatása, előkészítése

A cseppekről kinyert adataink két részre bonthatóak, rendelkezünk a csepp élettartama alatt minden időpillanathoz lemért adatokkal, illetve egy az élettartam egészét összefoglaló adatsorral.

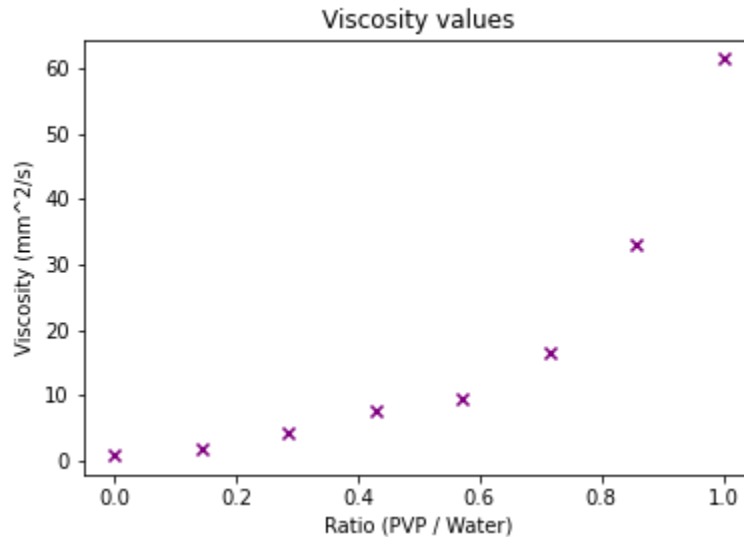
5.1 Főkomponens analízis

A cseppek adatsorainak jellemzésére létrehozott összefoglaló adatokkal főkomponens analízist [15] végeztünk. Az első 3 legnagyobb variancia lefedettséggel rendelkező főkomponenst ábráztuk. Az ábrán látszik, hogy különböző arányban kevert folyadékok elkülönülnek egymástól, a felhasznált jellemzők alapján. Ebből az következik, hogy osztályozási technikákkal lehetséges lehet a keverési arány meghatározása ezekkel az adatokkal.



14. ábra: főkomponens analízis a cseppek tulajdonságain

5.2 Viszkozitások



15. ábra: a keverési arányokhoz tartozó mért viszkozitások

A mérés alapján megállapított viszkozitás értékekből az látszik, hogy a PVP és desztillált víz keverési arányai és a viszkozitásai között exponenciális összefüggés van. Ez a nagy lépésköz a keverési arányok között feltehetően egyszerűbbé teszi, hogy megkülönböztessük a folyadékokat, mert így lényegesen más viszkozitású cseppek keletkeznek, amik jellemzői is jelentősen eltérőek lehetnek.

5.3 Tisztítás

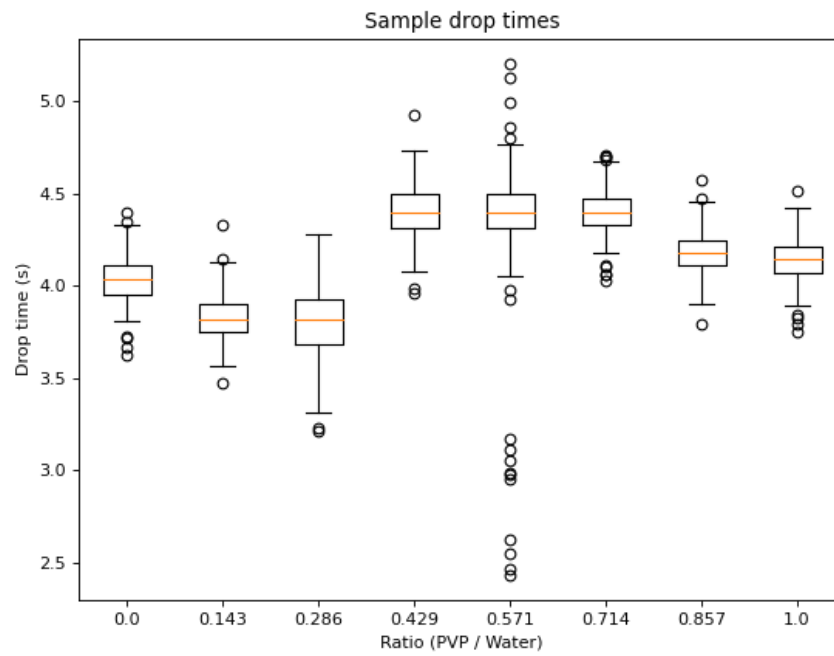
A mérési beállításunkban előfordulhat, hogy a csőbe, ami az adagolót a csepegtetővel köti össze, buborék kerül. Ilyenkor a folyadék vagy beragad és nem képes leválni vagy nagyon gyorsan leválik a csepegtetőtől.

Ezeknek a hibáknak a detektálásra a lecseppenési időt lehet használni. Amennyiben egy csepp ezen tulajdonsága jelentősen eltér a többi mintától, eltávolítjuk az adathalmazból. Az kivülesők (outlier) megtalálására az interquartile range-t [16] használjuk. Az elterjedt eljárás, hogy a $Q1 - 1.5 *$



16. ábra: buborékos csepp

IQR és a $Q3 + 1.5 * IQR$ tartományt őrizzük meg és a többi elemet kívülesőnek minősítjük.



17. ábra: lecseppenési idők

6 Becslés

A viszkozitás becslését két irányból lehet megközelíteni, amennyiben csak arra van szükség, hogy lehetséges legyen meghatározni, hogy a kívánt viszkozitás értékhez képest jelentős eltérés van-e, akkor klasszifikációs feladatként lehet értelmezni. Viszont amennyiben ténylegesen szeretnénk megbecsülni a viszkozitás értékét, akkor regressziós feladatról beszélünk.

Először, az osztályozási problémával foglalkoztunk, erre a logistic regression módszere jó eredményekkel szolgált, az idősoros adatok összefoglalására szolgáló adatok felhasználásával.

Utána a regressziós feladat megoldásával foglalkoztunk, amire kihasználva, hogy idősoros adatunk van konvolúciós neurális hálózatokkal kísérleteztünk. A konvolúciós neurális hálózatok elterjedtek olyan feladatokra, ahol az adatok idősorosak, hiszen a konvolúciós filterek segítségével lehetőség nyílik az azokban megtalálható mintázatok megtanulására.

6.1 Osztályozás

6.1.1 Adathalmaz

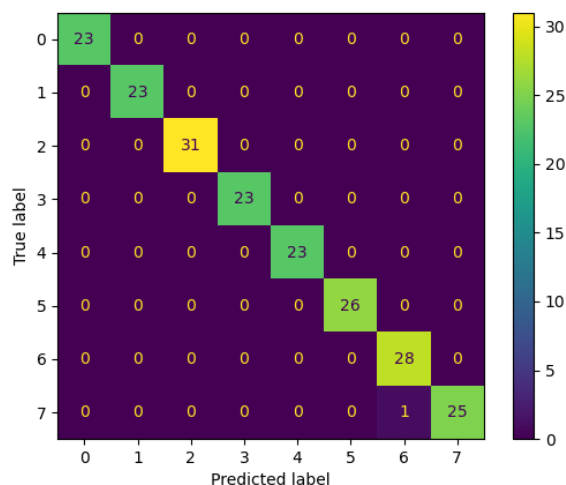
Az osztályozáshoz adathalmazként a cseppek lecseppenésének összefoglalására szolgáló adatokat használtuk. Tapasztalataink alapján ezek is elég információt tartalmaznak, hogy osztályozni lehessen őket a keverési arányaik szerint.

6.1.2 Multinomial logistic regression

PCA segítségével megállapítottuk, hogy a minták elkülöníthetőek az összefoglaló adataik alapján. Az osztályozásra Multinomial logistic regression-t használtunk Python-ban Sickit learn [17] könyvtár segítségével. Az algoritmust 100 iteráción keresztül futtattuk és Limited-memory BFGS [18] algoritmust használtunk az optimalizálásra.

6.1.3 Módszer értékelése

A fent bemutatott megoldás 99%-os pontosságot tudott elérni a teszt adathalmazon. Ebből megállapítható, hogy lehetséges annak a megállapítása, hogy az adott anyag víz-PVP keverési aránya lényegesen megváltozott, jelen esetben legalább 1/7-el, így az eltérés detektálása esetén lehetséges a hiba jelzése.



18. ábra: a teszt adathalmaz kiértékelésére szolgáló konfúziós mátrix

6.2 Regresszió

A regressziós megoldásra egy több bemenetű konvolúciós neurális háló [19] került implementálásra. Mivel a videofelvételeket képkockáról képkockára dolgozzuk fel, az egyes képkockákból kinyert adatok sorozata idősoros adat lesz. Ilyen idősoros adatok feldolgozására elterjedt módszer a konvolúciós hálók alkalmazása, mivel képes kinyerni azokat a jellemvonásokat, amik hasznosnak bizonyulnak a tanítás során.

6.2.1 Adathalmaz

A laboratóriumi mérés során készített videofelvételekből, majd azokból a korábban bemutatott feldolgozó programmal lement képkockánkénti adatokból állt össze a háló tanításához szükséges adathalmaz. A videófeldolgozása után a korábban bemutatott módszerrel a hibás cseppek el lettek távolítva az adatsorokból. A tanításhoz minden csepp 150 képkocka hosszúra lett normalizálva, úgy hogy az utolsó időpillanat mindig a lecseppenés előtti képkockához tartozó adat legyen.

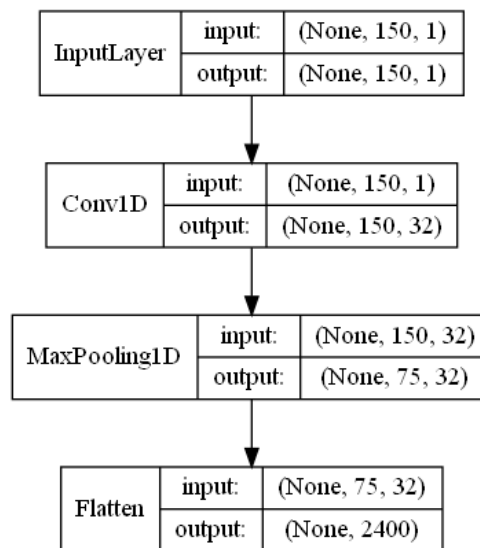
A viszkozitás értékekhez Gauss-eloszlás szerint néhány százaléknyi hibát rendeltünk, hogy a címkék olyanok legyenek, mintha minden egyes csepp viszkozitását külön lemértük volna és a mérés rendelkezett volna egy kis hibával. Minden címkét egyesével megszoroztunk egy 1 várható értékkel és 0.007 varianciával parametrizált eloszlással. Ezzel célunk a túlilleszkedés nehezítése volt.

6.2.2 Háló architektúrája

Mindegyik jellemvonáshoz (feature) tartozó adatsor külön bemeneten és konvolúciós rétegeken keresztül kerül feldolgozásra, a legtöbb adatsor egydimenziós, így egydimenziós konvolúciós rétegeken mennek keresztül. A feature, ami felosztja a csepp kontúráját 16 egyenlő szakaszra és az ezek görbületét írja le, 2 dimenziós adat hiszen itt az adat időbelisége mellett a helyzete is tartalmazhat lényeges információt. A konvolúciós ágról érkező kimenet összefűzésre kerül, majd az így létrehozott vektor lesz a teljesen összekötött rétegek bemenete.

A modell Python-ban, a Keras keretrendszerrel [20] készült PlaidML backenden [21], mivel AMD chippel rendelkező videókártyákhoz nem használható az elterjedten használt TensorFlow backend [22] és a saját hardware-m azzal rendelkezett. A továbbiakban bemutatásra kerül a legjobb eredményt elérő architektúra felépítése.

6.2.2.1 Egydimenziós konvolúciós ágak

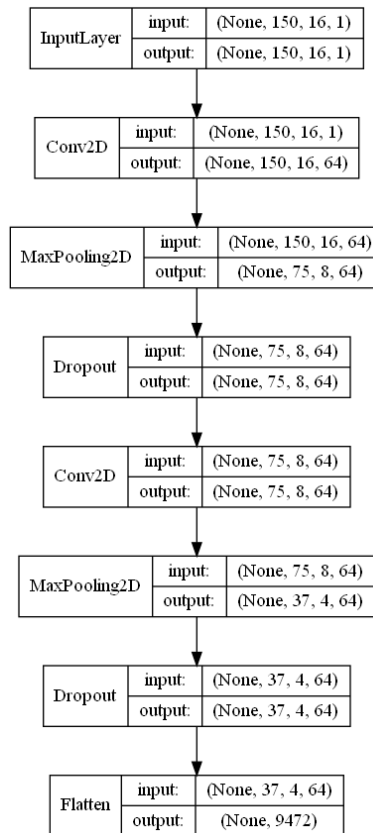


19. ábra: egydimenziós adatsorok feldolgozására szolgáló konvolúciós ág

Az egydimenziós adatsorok először egy konvolúciós rétegen mennek át, ami 32 filterrel és hármass kernel mérettel rendelkezik. Ezután maximum pooling kerül alkalmazásra 2-es kernel mérettel, hogy a bemenet hossza csökkentésre kerüljön, illetve, hogy a legfontosabb információk kiszűrésre kerüljenek. Végül az eredményt kilapítjuk a teljesen összekötött (fully connected) rétegek tanításához.

6.2.2.2 Kétdimenziós konvolúciós ág

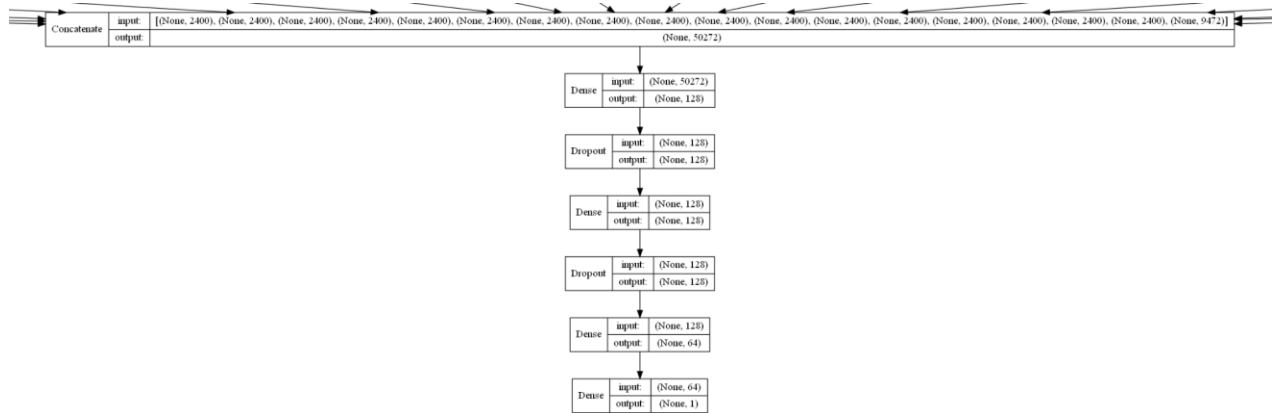
A cseppek kontúrját 16 egyenlő részre való felosztásából és ezeknek a szakaszoknak a görbület értékéből származtatható egy 16x150-es kétdimenziós mátrix. Az eddigi idősorokkal ellentétben itt a szakaszok egymáshoz viszonyított elhelyezkedése is tartalmazhat hasznos jellegzetességeket.



20. ábra: kétdimenziós adatsor feldolgozásra szolgáló konvolúciós ág

A kétdimenziós konvolúciós rétegek mindkét alkalommal 64 filterrel, és 3x3-as kernel mérettel rendelkeznek, a maximum pooling rétegek pedig 2x2-es kernel mérettel. A pooling rétegek után található egy-egy dropout réteg is 0.4-es kiesési értékkel, a dropout rétegek növelik a háló robusztusságát, illetve segítenek elkerülni a túlillesztést.

6.2.2.3 Teljesen összekötött rétegek (Fully connected layer)



21. ábra: fully connected rétegek

A konvolúciós ágakról érkező eredmények először összefűzésre kerülnek, majd 3 dense réteg következik, az első kettőt egy-egy dropout réteg követi 0.4-es dropout aránnyal. A dropout rétegek itt is a túlillesztést hivatottak elkerülni. Az első 2 dense réteg 128 neuronnal rendelkezik, a harmadik 64-gyel, mindegyik réteg aktivációs függvénye tangens hiperbolikus. Végül a kimeneti réteg 1 neuronból áll hiszen egy viszkozitás értéket várunk eredményül, ez a réteg lineáris aktivációt használ.

A korábban bemutatott minden réteg súly inicializására Glorot normal inicializációt [23] használ, hiszen a tangens hiperbolikus aktivációs függvényhez ez az optimális súly inicializáló eljárás.

6.2.3 Tanítás

A tanítás során Adam optimalizációt [24] használtunk 0.001-es tanulási rátával, 96-os batch méretre. Korai leállás módszerét (Early stopping) [25] alkalmaztuk a tanulási iterációk során 40-es türelmi idővel, így ha ezen időszakban nem javul érdemlegesen a validációs loss érték, akkor a tanítás véget ér. Hibafüggvényként a tanítás során az átlagos abszolút hibát használjuk.

Tanítás előtt az adatok standardizálva lesznek, illetve szükséges a viszkozitás értékek átskálázása is, mivel a tanító adatainkon ezek elég széles skálán oszlanak el (0.891-61), ezért 10-es alapú logaritmussal elérjük, hogy egyenletesebben legyen elosztva a viszkozítások, majd erre alkalmazunk min-max skálázást 0 és 1 közé. Enélkül a skálázás nélkül az első néhány kis értéket általában rosszul tanulta meg a háló, mivel úgy is lehetséges volt a veszteség

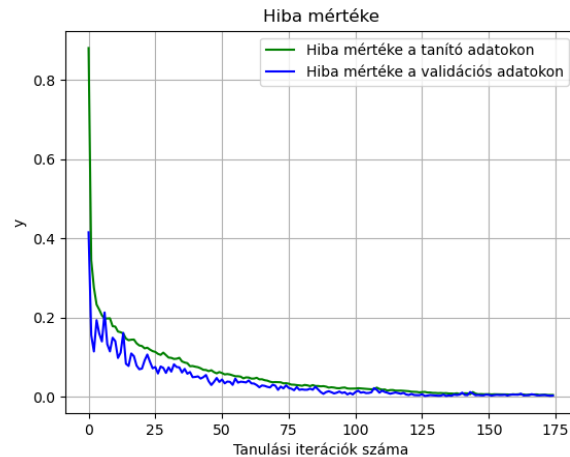
minimalizálása, hogy ezeket nem igazán tanulta meg (0-nak vette az 3 legkisebb viszkozitás értéket), így arányaiban nagyobb hiba keletkezett ezeken az adatokon.

6.2.4 Módszer értékelése

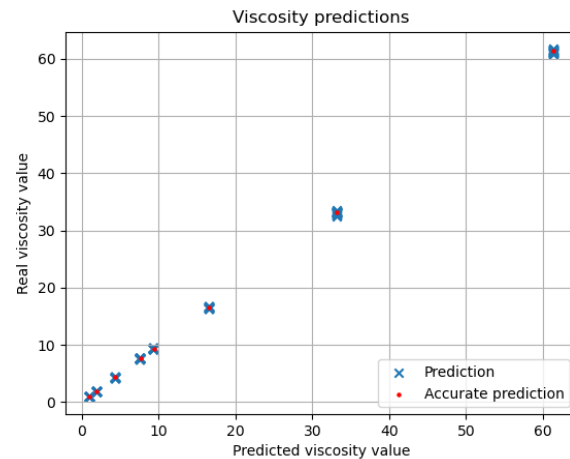
A háló képes volt megkülönböztetni a cseppeket és kis hibával meghatározni a hozzájuk tartozó viszkozitás értékeket is, azonban érdemes megjegyezni, hogy a különböző viszkozitású minták között az anyag keverési arányában és így a viszkozitásukban is jelentős különbségek voltak láthatóak. Amennyiben kisebbek lennének a gélek keverési arányai közötti eltérés, ezzel az architektúrával nem biztos, hogy sikerrel járnánk.

Amennyiben megpróbálunk eltávolítani egy adott viszkozitásnak a teljes adatsorát a tanító adatok közül és azon értékeljük ki a hálót, akkor azt tapasztaljuk, hogy nem képes meghatározni az ismeretlen viszkozitás értéket, viszont jellemzően az eggyel kisebb viszkozitású mintához hasonló értékekkel látja el a teszt adatokat.

Ebből arra a következtetésre juthatunk, hogy a modellnek nem sikerül megtanulnia olyan jellemvonásokat, amivel sikeresen le tudná írni a csepp általunk kinyert jellemvonásai és a viszkozitásaik közötti összefüggéseket. Ennek a hiányosságnak az egyik oka lehet, hogy bár viszonylag sok mintával rendelkezünk, nagyjából 250 cseppel mindegyik folyadékból, az egyfajta cseppek nagyon hasonlítanak egymásra, így az adathalmazunk diverzitása kicsi lesz,



22. ábra: tanítás során a hiba mértéke



23. ábra: a teszt adathalmazon becsült eredmények

így lényegében a tanítása során egy viszkozitás értékhez mindig nagyjából ugyanazt a mintát adjuk be a modellnek.

Egy jól működő megoldás kidolgozásához a jövőben szükséges lenne új mérések készítésére, ahol a mért folyadékok viszkozitás értékei kisebb lépésközökkel változnak. Ez kivitelezhető kisebb viszkozitású PVP-vel és annak további hígításával. Érdeemes lehet kisebb keverési tartományon (tisztá víztől csak feles keverési arányig), több mintát kikeverni, viszont az egyes mintákhoz elég rövid videókat készíteni, mert az azonos anyagból származó cseppek nem mutatnak lényeges eltérést. Illetve tesztelési szempontból előnyös lehet az egyenletesen elosztott keverési arányok közé újabb mintákat kikeverni, ezen adatok annak kiértékelésére nyújthatnak segítséget, hogy a megoldásunk képes volt-e olyan összefüggéseket találni, amik alapján képes ismeretlen viszkozításokat is megállapítani.

Összefoglalás

A dolgozatban bemutattuk a Pharmatech laboratórium egy aktív kutatását, melynek célja egy új képfeldolgozáson és géptanuláson alapuló viszkozitás mérési eljárás kidolgozása.

Először feltérképeztük a hagyományos viszkozitás mérési technikákat. Összegyűjtöttük ezek előnyeit és hátrányait, majd ezek mentén meghatároztuk fejlesztésünk célját. Olyan technika kidolgozását céloztuk meg, ami képes egy gyártási folyamatban, a folyadék cseppjeiről készült videofelvételek alapján valós időben meghatározni a folyadék viszkozitását.

Bemutattuk a laboratóriumi méréseink során használt mérési módszert a szükséges adatok előállítására, a felhasznált anyag minták előállítását, majd ezeknek a hagyományos módon történő viszkozitás mérését. Áttekintettük és értékeltük a korábban megalkotott képfeldolgozó szoftvert és a konvolúciós neurális hálóval való osztályozási kísérletet. Továbbiakban ismertettük a paramétereket, amikkel kiegészítettük a korábban cseppek képkockáiról mért adatok sorát és az újonnan létrehozott összefoglaló adatokat.

Megvizsgáltuk milyen igényeknek kell megfelelnie ahhoz, hogy ipari körülmények között alkalmazható legyen a megoldásunk és az itt felmerült igényeknek megfelelően módosítottuk a képkockák előfeldolgozását. Demonstráltuk a kiinduló szoftvernek és az új paraméterekkel kibővített változatának a feldolgozási sebességét. Ezt követően bemutattuk a lehetőségeket, melyekkel lehetséges a valós idejű működés elérése. Kiválasztottuk ezek közül a számunkra optimális módszereket és implementáltuk is azokat.

A feldolgozó programmal kinyert csepp tulajdonságokat megvizsgáltuk főkomponens analízissel, bemutattuk a mintáink alapján a keverési arányaink és a mért viszkozitásértékeink közötti összefüggést, majd a mérés során előforduló hibás csepp mintákon szűrést végeztünk.

Végezetül bemutattunk kétféle megközelítést gépi tanulással való viszkozitás becslésre. Osztályozást végeztünk multinomial logistic regression módszerével, ami közel hibátlanul teljesített a teszt adathalmazon. A viszkozitás értékek regresszióval való becslésére, létrehoztunk egy több bemenetű neurális hálózatot, azonban a tanításaink során a modell túlilleszkedett a mintákhoz, így ezen adathalmazzal, nem találtuk lehetségesnek, a feladat megoldását.

Köszönetnyilvánítás

Szeretnénk megköszönni konzulensünknek, Dr. Csorba Kristófnak a segítségét és iránymutatását. Ezen felül szeretnénk köszönetet mondani a Pharmatech laboratórium munkatársainak, hogy biztosították a mérésekhez szükséges eszközöket és segítséget nyújtottak azok elvégzésében.

Az eredmények a 2019-1.1.1-PIACI-KFI-2019-00263 projekt keretében valósultak meg a Nemzeti Kutatási, Fejlesztési és Innovációs alap 2019-1.1 séma szerinti támogatásával.

Irodalomjegyzék

- [1] B. Á. Dr. Jancsó, „Viszkozitás mérésének automatizálása képfeldolgozás segítségével,” 2021. [Online]. Available: <https://diplomaterv.vik.bme.hu/hu/Theses/Viszkozitas-meresenek-automatizalasa>.
- [2] K. A. Abbas, S. M. Abdulkarim, A. M. Saleh, M. Ebrahimian, „Suitability of viscosity measurement methods for liquid food variety and applicability in food industry - A review,” *Journal of Food, Agriculture & Environment*, pp. 100-107, 2010.
- [3] M. Fowler, „Dropping the Ball (Slowly),” [Online]. Available: https://galileo.phys.virginia.edu/classes/152.mf1i.spring02/Stokes_Law.htm. [Hozzáférés dátuma: 22 10 2022].
- [4] Alan S. Morris, Reza Langari, *Measurement and Instrumentation*, 2016.
- [5] Ajit S. Narang, Sherif I.F. Badawy, *Handbook of Pharmaceutical Wet Granulation*, 2019.
- [6] OpenCV team, „OpenCV,” [Online]. Available: <https://opencv.org/>. [Hozzáférés dátuma: 30 10 2022].
- [7] "shimat" felhasználó és mások, „OpenCvSharp,” [Online]. Available: <https://github.com/shimat/opencvsharp>. [Hozzáférés dátuma: 31 10 2022].
- [8] Leonidas Deligiannidis, Hamid R. Arabnia, *Emerging Trends in Image Processing, Computer Vision and Pattern Recognition*, 2015.
- [9] Microsoft, „Parallel Class,” [Online]. Available: <https://learn.microsoft.com/en-us/dotnet/api/system.threading.tasks.parallel?view=net-6.0>. [Hozzáférés dátuma: 12 9 2022].
- [10] NVIDIA, „CUDA Toolkit,” [Online]. Available: <https://developer.nvidia.com/cuda-toolkit>. [Hozzáférés dátuma: 10 10 2022].

- [11] A. Syberfeldt és T. Ekblom, „A Comparative Evaluation of the GPU vs The CPU for Parallelization of Evolutionary Algorithms Through Multiple Independent Runs,” *International Journal of Information Technology and Computer Science*, 2017.
- [12] OpenCV team, „Cuda,” [Online]. Available: <https://opencv.org/platforms/cuda/>. [Hozzáférés dátuma: 30 10 2022].
- [13] Emgu, „Main Page,” [Online]. Available: https://www.emgu.com/wiki/index.php/Main_Page. [Hozzáférés dátuma: 30 10 2022].
- [14] Emgu Corporation, „Emgu.CV.Cuda Namespace,” [Online]. Available: <https://www.emgu.com/wiki/files/4.5.1/document/html/04a0b184-ff75-176f-07c7-7a958e28d860.htm>. [Hozzáférés dátuma: 30 10 2022].
- [15] J. C. Ian T. Jolliffe, „Principal component analysis: a review and recent developments,” *PHILOSOPHICAL TRANSACTIONS OF THE ROYAL SOCIETY A*, 2016.
- [16] H. P. Vinutha, B. Poornima, B. M. Sagar, „Detection of Outliers Using Interquartile Range Technique from Intrusion Dataset,” *Information and Decision Sciences*, pp. 511-518, 2018.
- [17] Sickit learn, „Logistic Regression,” [Online]. Available: https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.LogisticRegression.html. [Hozzáférés dátuma: 30 10 2022].
- [18] D. N. J. Liu, „On the limited memory BFGS method for large scale optimization,” *Mathematical Programming*, p. 503–528, 1989.
- [19] Zewen Li, Fan Liu, Wenjie Yang, Shouheng Peng, Jun Zhou, „A Survey of Convolutional Neural Networks: Analysis, Applications, and Prospects,” *IEEE Transactions on Neural Networks and Learning Systems*, pp. 1-21, 2021.
- [20] Keras, „Keras,” [Online]. Available: <https://keras.io/>. [Hozzáférés dátuma: 10 10 2022].

- [21] Intel, „PlaidML,” [Online]. Available: <https://www.intel.com/content/www/us/en/artificial-intelligence/plaidml.html>. [Hozzáféres dátuma: 10 10 2022].
- [22] TensorFlow, „TensorFlow,” [Online]. Available: <https://www.tensorflow.org/>. [Hozzáféres dátuma: 20 10 2022].
- [23] Keras, „Layer weight initializers,” [Online]. Available: <https://keras.io/api/layers/initializers/>. [Hozzáféres dátuma: 10 10 2022].
- [24] Diederik P. Kingma, Jimmy Ba, „Adam: A Method for Stochastic Optimization,” in *3rd International Conference for Learning Representations*, San Diego, 2015.
- [25] Yao, Y., Rosasco, L. & Caponnetto, „On Early Stopping in Gradient Descent Learning,” *Constructive Approximation*, pp. 289-315, 2007.