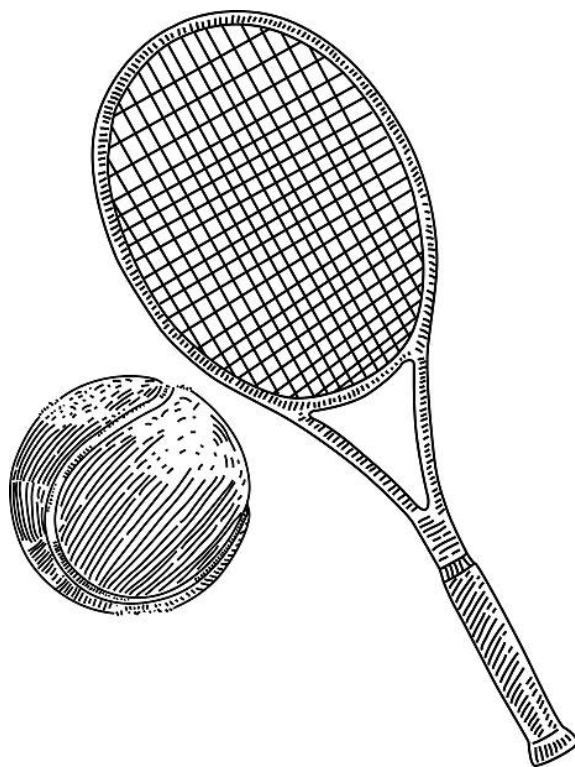


IROB Onderzoek

Joey de Ruiter & Sergi Philipsen

7 juli 2020



Wat is de beste manier om een robot efficiënt en snel tennisballen op te laten rapen?

Inhoudsopgave

1	Inleiding	3
2	Probleemanalyse	4
2.1	Aanleiding en context	4
2.2	Probleemstelling	4
2.3	Doelstelling	4
3	Theoretisch kader	5
3.1	Zumo-robot	5
3.2	Machine-learning	5
4	Onderzoekopzet en uitvoering	6
5	Resultaten	7
6	Onderzoek & Discussie	11
7	Conclusie	15

1 Inleiding

In dit onderzoek worden de mogelijkheden over het design en technische eigenschappen van de tennisballen opraaprobot aan de proef gesteld. Uit dit onderzoek volgt hoe het projectteam de best mogelijke technieken kiest voor het bouwen van een tennis bal opraap robot.

Er wordt als eerste gekeken naar het theoretisch kader, dat een wetenschappelijke basis vormt voor dit project. Hierin worden worden gebruikte begrippen gedefinieerd om zo duidelijk mogelijk een beeld te schappen over de verschillende termen die uiteindelijk gebruikt zullen worden als mogelijk referentie materiaal gebruikt kan worden tijdens het ontwikkelen van de robot.

Met de hoofdvraag en deelvragen die opgesteld zijn zal uiteindelijk een antwoord worden gevonden op de centrale vraag: *“Wat is de beste manier om een robot efficiënt en snel tennisballen op laten rapen?”*

Het uiteindelijke antwoord zal worden opgenomen in de conclusie en hierop zal een aanbeveling worden gedaan.

2 Probleemanalyse

2.1 Aanleiding en context

Voor de minor *Robotica, domotics en industriële automatisering* gaan we een robot maken die tennisballen op kan rapen. Om zeker te zijn hoe we dit product voor ons zien en aan gaan pakken, doen we dit onderzoek.

2.2 Probleemstelling

Het oprapen van tennisballen is een tijdsintensieve klus die de tennisspeler weerhoudt om door te gaan met sporten.

2.3 Doelstelling

De doelstelling is om onderzoek naar de beste mogelijkheden die wij kunnen toepassen om de robot te realiseren.

3 Theoretisch kader

3.1 Zumo-robot

De Zumo-robot of Zumobot is een robot gemaakt door Pololu (Pololu, g.d.). De robot bestaat uit een klein chassis gekoppeld met rupsbanden aan beide kanten en een kleine schans aan de voorkant. Tevens is de robot klein genoeg om mee te doen met mini sumo competities. Deze robot is goed geschikt als basis voor je eigen robot ontwerp.



3.2 Machine-learning

Machine learning is de studie van computeralgoritmen die automatisch verbeteren door ervaring. (Mitchell, 1997) Het wordt gezien als een subset van kunstmatige intelligentie. Machine learning-algoritmen bouwen een wiskundig model op basis van voorbeeldgegevens, ook wel 'training data' genoemd, om voorspellingen of beslissingen te nemen zonder hiervoor expliciet te zijn geprogrammeerd. (Koza, 1996)

4 Onderzoekopzet en uitvoering

Hoofdvraag

Wat is de beste manier om een robot efficiënt en snel tennisballen op te laten rapen?

De hoofdvraag zal worden beantwoord middels de antwoorden op de deelvragen en wordt vastgesteld in de conclusie.

Deelvragen

1. Wanneer moeten de tennisballen opgeraapt worden?
2. Waar op het tennisveld moeten de tennisballen worden opgeraapt?
3. Wat is de beste verzamelmethode die de Zumobot kan gebruiken zonder belast te raken?
4. Hoe kan de gebruiker op een eenvoudige wijze de robot bedienen?
5. Welke software en protocollen kunnen het best gebruikt worden voor de bediening van de robot?
6. Hoe weet de robot waar die zich bevind op het tennisveld en wat is hier voor nodig?
7. Hoe worden de tennisballen gedetecteerd?
8. Welke machine-learning kan het best gebruikt worden voor de tennisballendetectie?
9. Hoe worden de detecties omgezet in acties?

De deelvragen zullen worden beantwoord door te discussiëren over de resultaten die zijn gevonden.

5 Resultaten

1. Wanneer moeten de tennisballen opgeraapt worden?

Uit eigen bevindingen is het mogelijk om verschillende manieren te gebruiken om te detecteren wanneer de tennisbal opgeraapt moet worden.

Het zou mogelijk zijn om de machine alle ballen aan het eind van een tennis oefening op te laten ruimen, maar het zou ook een mogelijkheid zijn om de bal tussendoor op te halen als er een fluitje af gaat, een druk op de knop van een mobiele app of afstandsbediening.

Ook zou de robot door de wedstrijd door de ballen op kunnen ruimen, als die niet in de weg zit.

2. Waar op het tennisveld moeten de tennisballen worden opgeraapt?

In de oefen en wedstrijd scenario's zijn op de volgende plekken tennisballen te vinden.

- Voor en achter het net
- Achter de zijlijn
- Achter de achterlijn
- Op het speelveld

Echter zal in beide gevallen de maten waarin dit is verschillen. Hierdoor moet er misschien worden gehouden aan hoe vaak de robot langs deze plekken gaat.

3. Wat is de beste verzamelmethode die de Zumobot kan gebruiken zonder belast te raken?

Om te kijken wat de beste verzamelmethode is moeten we eerst kijken naar hoe we de ballen willen opslaan. Hierbij komen de volgende ideeën naar boven:

- De ballen bewaren in een hoekje van het veld zodat ze later gemakkelijk opgeraapt kunnen worden.
- De ballen bewaren in een interne opslag van de robot zelf. Zodat die deze gelijk naar de spelers terug gebracht kan worden.
- De ballen in een vat stoppen zodat deze gemakkelijk opgeruimd kunnen worden.

Nu kunnen we het gaan houden over de mogelijke verzamelmethodes voor de tennisballen.

- De robot heeft een u-vormige ram aan de voorkant gemonteerd waarmee die de ballen over het veld kan duwen.
- De robot heeft een bus waarin die meerdere tennisballen mee kan nemen en ergens anders af kan leveren.
- De robot heeft een net waarin die meerdere tennisballen mee kan slepen en bij zich kan houden.
- De robot heeft een springveer op zich waarmee hij een tennisbal over het speelveld naar de juiste hoek kan schieten.

4. Hoe kan de gebruiker op een eenvoudige wijze de robot bedienen?

Met een afstandsbediening kan worden aangegeven welke “zone” van het veld opgeruimd moet worden. Tevens kan er ook met een app of een website worden bepaald waar opgeruimd moet worden, of kan de robot worden bestuurd.

5. Welke software en protocollen kunnen het best gebruikt worden voor de bediening van de robot?

Voor een afstandsbediening kan het beste via infra rood of 433Mhz de commando's worden verstuurd. Dit kan op de Arduino zonder dat er extra dingen nodig zijn, naast de sensoren zelf. Het zou ook mogelijk zijn om een website te gebruiken maar dan moet er gebruikt gemaakt worden van een Wi-Fi enabled chip, zoals een ESP8266 of Raspberry Pi.

6. Hoe weet de robot waar die zich bevind op het tennisveld en wat is hier voor nodig?

De robot kan zichzelf kalibreren door eerst langs de randen van het veld te rijden. Eventueel kan de robot ook gebruik maken van de lichtbalk sensoren die onderaan de robot is gemonteerd. Op deze manier heeft de robot begrip voor waar de “zones” zich bevinden.

Met een hoog gemonteerde camera kan het volledige veld en de robot worden gedetecteerd. Dan is ten alle tijden duidelijk waar de robot zich bevind. Wellicht is het ook mogelijk om te berekenen in welk veld de Zumo robot zich bevindt doormiddel van een Raspberry Pi camera en wat ruimte bepaling.

Een alternatief zou zijn om een afstandssensor op de voorkant van de robot plaatsen, hiermee geef je hem de mogelijkheid om muren en andere obstakels en muren te ontwijken maar verliezen we een manier om accuraat bij te houden in welke zone we zitten.

Het is mogelijk om met BLE (Bluetooth Low Energy) beacons een verbinding

te maken met de Raspberry Pi. Door gebruik te maken van signaalsterkte en driehoeksmetingen kan de Raspberry Pi zichzelf localiseren (Leverge, 2019). Er zijn ook Bluetooth 5.1 oplossingen die kunnen meten wat de afstand en hoek van een verbinding is. Dit zou heel erg handig zijn aangezien er dan maar één beacon hoeft te worden geplaatst.

Een andere optie (die gesuggereerd werd door Vincent) is dat we een combinatie van gps of BLE en object detectie gebruiken om zo tot een relatief accuraat resultaat te komen. Hier zouden de palen van het net kunnen worden gebruikt als herkenningspunt.

7. Hoe worden de tennisballen gedetecteerd?

De tennisballen kunnen gevonden worden met de camera die gemonteerd wordt op de robot. Met de output van deze camera is het mogelijk om object detectie uit te voeren met behulp van machine learning. Hieruit hopen we de tennisballen te kunnen te identificeren.

Ook kan er gebruik worden gemaakt van een afstandssensor in combinatie met een kleursensor om te kijken of er groen-gele objecten op het pad van de robot zijn.

8. Welke machine-learning kan het best gebruikt worden voor de tennisballendetectie?

De tennisballen kunnen snel worden gedetecteerd met behulp van een CNN (Ren'o, 2018). Met edge detectie kunnen de ronde vormen makkelijk herkend worden. Ook kunnen de ballen herkend worden aan de duidelijke geel-groene kleur. Dit kan heel goed met OpenCV (Heywood, 2017)

Er kan ook gebruik worden gemaakt van een object detection classifier. Deze heeft het voordeel dat andere objecten als het net, de speler en andere obstakels ook herkend kunnen worden. (Rosebrock, 2020)

9. Hoe worden de detecties omgezet in acties?

Het zou mogelijk zijn om de robot meerdere statussen te geven, bijvoorbeeld eentje die zoekt naar een bal, eentje die een bal aan het tracken is en eentje die de bal aan het wegbrengen is.

Bij deze statussen is het belangrijk dat de robot zich op 1 onderdeel kan focussen en dat er niet andere acties van andere statussen “lekken” in de huidige status.

Het zou ook mogelijk zijn om continue de dichtsbijzijnde bal te zoeken en deze op te halen en weg brengen. Een ander idee zou zijn om met een punten systeem te werken, die rekening houdt met de afstand en de hoeveelheid ballen. Zo kan

de robot zo efficiënt mogelijk te werk gaan.

De communicatie die op de robot zal plaatsvinden tussen de micro-controllers zal via de fysieke draden lopen. Voor het protocol hebben we dan veel verschillende keuzes, de meest voor de handliggende keuze lijkt ons UART (TI, 2010), dit komt omdat het een simpel te implementeren protocol is die weinig extra hardware nodig heeft. Verder hebben wij volledige controle over het dataformaat en baud-rate, hierdoor zijn de min-punten van UART eigenlijk een non-issue. Ook al zeggen we dat UART de meest voor de handliggende methode is moeten we nog steeds in ons achterhoofd houden dat andere protocollen ook een goede fit zouden zijn.

Een ander mogelijk protocol zou I^2C (NXP, 2014) zijn. Dit komt voornamelijk omdat het een heel populair en simpel protocol is waardoor het op vele platformen al een library heeft. Tevens biedt I^2C nog wat fijne features, zoals dat de master de volledige bus beheert, hierdoor hoeven de slaves geen instellingen te delen. Helaas gaat de performance van dit protocol wel erg naar beneden als er veel data over de lijn wordt gestuurd.

6 Onderzoek & Discussie

1. Wanneer moeten de tennisballen opgeraapt worden?

Aangezien grote wedstrijden vaak al ballenjongens hebben en niet het probleem hebben dat er steeds veel ballen op moeten worden geraapt, focussen wij ons op de trainingen en amateurs. Voor de veiligheid van de robot, laten we deze langs de zijlijn wachten totdat deze aangestuurd wordt om de ballen te verzamelen.

2. Waar op het tennisveld moeten de tennisballen worden opgeraapt?

Het is in meerdere gevallen nodig dat de ballen over het gehele veld verzameld zullen worden. Echter is het wel zo dat bij trainingen merendeels van de ballen zich aan de tegengestelde zijde van de speler bevinden. Het lijkt ons handig om te werken met “zones” en aan de robot door te kunnen geven welke “zone” die moet opruimen.

3. Wat is de beste verzamelmethode die de Zumobot kan gebruiken zonder belast te raken?

Door de kleine motoren gemonteerd op de Zumo robot, hebben wij het gevoel dat als we een interne opslag erop monteren we niet genoeg kracht hebben om vooruit te komen. Tevens hebben we hetzelfde gevoel als we met een bal een schans op gaan rijden om deze uiteindelijk in een vat/opslag te krijgen.

Hierdoor blijft er eigenlijk maar één mogelijke oplossing over, en dat is de ballen bewaren in een hoekje van het veld zodat het makkelijk is op te ruimen door de tennisspeler.

4. Hoe kan de gebruiker op een eenvoudige wijze de robot bedienen?

Omdat we denken dat het fijn is voor de gebruiker om feedback te krijgen, lijkt het ons het handigst om dit met een web applicatie te doen. Dit is ook technisch mogelijk zonder dat wij extra hardware hoeven te bestellen.

5. Welke software en protocollen kunnen het best gebruikt worden voor de bediening van de robot?

Om een web applicatie te bouwen moeten we de applicatie natuurlijk bereikbaar maken voor een webbrowser, hiervoor zijn enkele mogelijkheden de meest simpele en meest robuuste techniek zou zijn om de Raspberry Pi zijn eigen Wi-Fi hotspot te laten creëren. Hiermee is het mogelijk om een verbinding te leggen tussen de robot en een extern apparaat zoals een smartphone of een laptop.

Met deze verbinding is het mogelijk om HTTP verkeer op te zetten tussen de webserver zoals een Nginx server en een internetbrowser.

6. Hoe weet de robot waar die zich bevind op het tennisveld en wat is hier voor nodig?

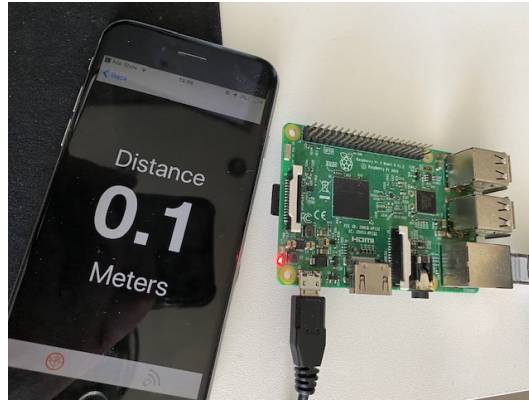
Het kabileren van de robot op het veld zal niet werken, dit komt omdat de motoren van de robot DC motoren zijn en iedere “stap” die de robot zet is een niet een uniformen afstand is. Hierdoor is het niet mogelijk om te vertrouwen op de kalibratie.

Een hoog gemonteerde camera zou hier de technisch correcte oplossing zijn, met de camera is het mogelijk om continue de positie van de robot te weten en in welke zone deze aan het werk is. Helaas moeten we nog een camera ter beschikking hebben en daarnaast is het minder gebruiksvriendelijk om een camera op te moeten hangen iedere keer als een persoon wilt trainen.

De zones kunnen worden bepaald met een afstandssensor, deze kan tot vier en een halve meter detecteren. Zo kunnen we ongeveer bijhouden of de robot niet het veld betreed door niet te ver van de muur te gaan. Ook kan de robot op deze manier lang de zijlijn blijven, al zal het niet heel accuraat zijn. Er is een mogelijkheid dat het veld geen muur heeft, in dit geval zal de robot niet goed kunnen bepalen waar die is.

Het is ook mogelijk om het met een meer “lokale” techniek indoor positioning toe te passen, hiervoor hebben we twee mogelijkheden gevonden die goed kunnen worden toegepast op dit project: een BLE gebaseerde oplossing of een Bluetooth 5.1 oplossing met Angle of Arrival (AoA) en Angle of Departure (AoD) oplossing.

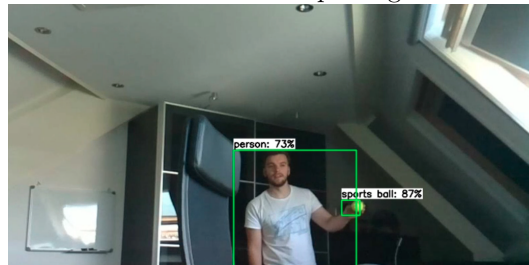
Het bluetooth 5.1 protocol is best nieuw, waardoor er nog weinig goedkope hardware voor beschikbaar is. Aangezien de oplossingen vaak zeldzaam en meer dan €40,- zijn, kunnen we deze manier van lokalisering maar beter achterwegen laten. Om BLE verder te onderzoeken hebben we een klein onderzoek gedaan met het opzetten van een implementatie van de BLE beacon techniek: iBeacon (Apple, 2013), met dit onderzoek kwamen wij er achter dat het mogelijk was om relatief accuraat de afstand tussen de Pi en de het apparaat (in ons geval een telefoon) te berekenen. Helaas werkt het minder als er een object in de weg staat van het signaal, maar we nemen aan dat er geen signaal blokkerende objecten op de tennisbaan staan.



7. Hoe worden de tennisballen gedetecteerd?

Voor de kleurensensor moet het object heel dichtbij komen. Dit is niet zo handig als er veel ballen in een keer moeten worden opgeruimd. Het lijkt ons daarom handiger om een Raspberry Pi met een camera te gebruiken. Het fijne hiervan is dat de Raspberry Pi ook de webserver bevat. Hierdoor kan heel veel logica op de Raspberry Pi worden gedaan, wat net wat fijner werkt. Zo hoeven we niet veel rekening te houden met opslag en kunnen we elke programmeertaal kiezen die we willen.

We hebben ook de getest of het mogelijk is om object detectie te doen op een relatief grote afstand. Het resultaat hiervan was dat een persoon en bal met een goed getrainde dataset (coco dataset) nog op een verre afstand gedetecteerd werd. Hierdoor nemen wij de aannamen dat als we een goed getrainde dataset op tennisballen hebben dat dit ook werkt op een grotere afstand.



8. Welke machine-learning kan het best gebruikt worden voor de tennisballendetectie?

Om een accuraat en efficiënte robot te maken lijkt het ons handig om gebruik te maken van de snelste manier van machine learning. Er zijn twee state of the art modellen die gebruikt kunnen worden voor object detectie op kleine apparaten zoals de Raspberry pi. Dit zijn SSD en YOLO (Gunnarsson, 2019). Uit een vergelijking blijkt dat SSD toch net iets sneller en accurater is dan YOLO

op kleine apparaten. Ook willen wij gebruik maken van de Google Coral die de performance van de Raspberry Pi aanzienlijk kan verbeteren. Deze limiteerd ons enigzins met modellen. Gelukkig is er wel een SSD model voor. (Tensorflow, 2020) Hierom hebben wij besloten om gebruik te maken van het volgende model: *ssd_mobilenet_v2_quantized_coco*.

9. Hoe worden de detecties omgezet in acties?

Het lijkt ons het beste om de robot meerdere statussen te geven, hierdoor kunnen we echt de focus leggen op de toestand die belangrijk is op het huidige moment. Zonder dat het een “spaghetti” wordt met wat de robot op dat moment moet doen.

Voor communicatie tussen de Arduino en de Raspberry Pi gaan we I^2C gebruiken, de meest voornaamste redenen hiervoor is dat dit de standaard is binnen Wire library van de Arduino en het ook gemakkelijk in te stellen is op de Raspberry Pi. Verder heeft het min-punt van een lage transfer rate bij veel data geen grote invloed, aangezien er maar af en toe kleinen pakketten naar de Arduino wordt gestuurd met alleen maar aansturing commando's.

7 Conclusie

Om antwoord te geven op de hoofdvraag “*Wat is de beste manier om een robot efficiënt en snel tennisballen op te laten rapen?*” hebben wij geconcludeerd dat we voor het verzamelen de u-vormige ram het beste is. Dit komt voornamelijk omdat wij erg gelimiteerd zijn in het gebruik van materialen en hardware. Het verzamelen zal de robot in zones doen, we nemen namelijk aan dat de ballen ongeveer bij elkaar liggen, hierdoor zou de robot iets efficiënter zijn werk kunnen doen. De zone waar de robot zal opruimen, kan worden aangegeven door de gebruiker via een web applicatie. Om te bepalen in welke zone de robot zich bevindt gaan we gebruik maken van BLE lokalisering, ook al is dit niet super accuraat, we gaan er van uit dat dit accuraat genoeg is voor onze toepassing.

In de zone zal de robot gebruik maken van de Raspberry Pi en de camera om de ballen in beeld te brengen. Met object detectie bepalen we waar de ballen zijn en waar de robot naartoe zal rijden. De ballen zullen vervolgens naar een verzamelpunt worden gebracht.

Literatuurlijst

- Apple. (2013). *iBeacon*. Verkregen 29 mei 2020, van <https://developer.apple.com/ibeacon/>
- Gunnarsson, A. (2019). *Real time object detection on a Raspberry Pi*. Verkregen 27 mei 2020, van <https://www.diva-portal.org/smash/get/diva2:1361039/FULLTEXT01.pdf>
- Heywood, M. (2017, december 31). *Object Detection and Tracking with OpenCV and Python*. Verkregen 26 mei 2020, van <https://www.bluetin.io/opencv/object-detection-tracking-opencv-python/>
- Koza, J. R. e. (1996). *Automated Design of Both the Topology and Sizing of Analog Electrical Circuits Using Genetic Programming*. https://doi.org/10.1007/978-94-009-0279-4_9
- Leverge. (2019, januari 4). *Indoor Positioning with Bluetooth Low Energy (BLE)*. Verkregen 28 mei 2020, van <https://www.iotforall.com/indoor-positioning-bluetooth-low-energy-ble/>
- Mitchell, T. (1997). *Machine Learning*. Verkregen 25 mei 2020, van <http://www.cs.cmu.edu/~tom/mlbook.html>
- NXP. (2014, april 4). *I2C-bus specification and user manual*. Verkregen 26 mei 2020, van <https://www.nxp.com/docs/en/user-guide/UM10204.pdf>
- Polulu. (g.d.). *Zumo-bot*. Verkregen 25 mei 2020, van <https://www.pololu.com/category/129/zumo-robots-and-accessories>
- Ren'ō, V. e. (2018). *Convolutional Neural Networks based ball detection in tennis games*. Verkregen 26 mei 2020, van http://openaccess.thecvf.com/content_cvpr_2018_workshops/papers/w34/Reno_Convolutional_Neural_Networks_CVPR_2018_paper.pdf
- Rosebrock, A. (2020, januari 27). *YOLO and Tiny-YOLO object detection on the Raspberry Pi and Movidius NCS*. Verkregen 26 mei 2020, van <https://www.pyimagesearch.com/2020/01/27/yolo-and-tiny-yolo-object-detection-on-the-raspberry-pi-and-movidius-ncs/>
- Tensorflow. (2020). *Tensorflow detection model zoo*. Verkregen 27 mei 2020, van https://github.com/tensorflow/models/blob/master/research/object_detection/g3doc/detection_model_zoo.md
- TI. (2010, november). *Universal Asynchronous Receiver/Transmitter*. Verkregen 26 mei 2020, van <https://www.ti.com/lit/ug/sprugp1/sprugp1.pdf?ts=1590482964077>