Master Thesis

# Internal and external spatial attention re-examined with lateralized power spectra

Ivania Jahangier

Faculty of Behavioural, Management and Social sciences
Human Factors and Engineering Psychology
University of Twente

First supervisor: Dr. Rob H.J. van der Lubbe
Second supervisor: Dr. Russell W. Chan

# Abstract

Attention can be oriented internally towards the mind (internal attention) or externally towards the environment (external attention). Strong evidence suggests that both internal and external attention share their underlying mechanisms. However, this view is still being questioned. One possible explanation for this opposing view could be due to potential issues in the experimental design used by previous studies. This is visible in the research on the relationship between internal and external spatial attention by Van der Lubbe et al. (2014) and the subsequent replication of this study by Willems (2020). Both studies found vastly different results regarding the underlying mechanisms of internal and external attention using similar methodologies. We propose that this variation in results was caused by the experimental design that allowed for internal attention, specifically, to not be measured as intended. We aimed to conceptually replicate both studies by modifying their experimental design to appropriately reflect internal attention. The cueing paradigm used in this study consisted of two only conditions; a pre-cue condition that reflects external attention and a post-cue that reflects internal attention.  Behavioural results (n = 29) revealed a Simon effect across both conditions for both reaction time and accuracy. The EEG analysis revealed an increased ipsilateral power in the alpha band at parieto-occipital sites during both conditions. These findings add support to the view that internal and external spatial attention share their underlying mechanisms. The design has also led to more pronounced behavioural and EEG results compared to the studies it tried to replicate. This indicates that the modifications have improved the robustness of this experimental design.

# Introduction

For many decades the relationship between the process of viewing outward items (external spatial attention) and retrieving items from working memory (internal spatial attention) has been a focal point in neuroscientific research, with the current literature still debating on the exact nature of this relationship**.** Strong evidence supports the view that internal and external spatial attention share similar processes (Kuo et al., 2009; Nobre et al., 2004; Oberauer, 2019; Van der Lubbe et al., 2014). However, due to the use of vastly different experimental designs used in the literature (e.g.  Lepsien et al., 2005; Mössing and Busch, 2020), and the limited number of studies that are being replicated (Pavlov et al., 2021), the reliability of this view is still being questioned.

## The relationship between internal and external spatial attention

Whether the mechanisms used in both internal and external spatial attention are comparable has been widely studied (Kuo et al., 2009; Lepsien et al., 2005; Nobre et al., 2004; Scheibner et al., 2017; Van der Lubbe et al., 2014; Willems, 2020). While the exact nature of this relationship is debated, there is a broad agreement that attention and working memory are closely related (for a review see Oberauer, 2019), as attention can be influenced by the contents of working memory (Foerster and Schneider, 2018). Working memory is a limited-capacity storage system used to maintain items relevant to the current task (Oberauer, 2019; Ma et al., 2014). This limited capacity applies to both information that is stored in the working memory (Luck and Vogel, 2013), and information that is still externally visible (Tsubomi et al., 2013; but see Lepsien et al., 2005). Chun (2011) has also suggested that visual working memory and visual attention are so closely related that visual memory itself can be considered as internal attention.

Neurophysiological markers of attention shifts, such as alpha (α) band desynchronizations have also been associated with processes in both internal and external attention. These oscillations are negatively linked to visual processing and attention, with increases in α power leading to decreases in visual processing (for a review see Clayton et al., 2017). It follows that α power may be involved in suppressing or regulating visual perception. This can be observed in lateralized α power at parieto-occipital sites, which has been associated with selective orienting of attention toward relevant stimuli and the inhibition of irrelevant stimuli (Kelly et al., 2006; Worden, 2000). These lateralizations reflect increases in α power in areas ipsilateral to the location of the attended stimuli (Sauseng et al., 2005; Van der Lubbe et al., 2014), and decreases in α power contralateral to the attended location (Kelly et al., 2009). These lateralizations are comparable between internal (Worden, 2000) and external attention (Lepsien et al., 2005; Nobre et al., 2004; Wallis et al., 2015). However, α band lateralizations during internal attention tasks could also indicate automatic shifts in attention unrelated to the attended stimuli (Mössing and Busch, 2020).

Multiple studies have also examined the effect of behavioural tasks on attention. An important behavioural measure paired with internal and external attention is the Simon task (Simon 1969). This task examines if reaction times towards relevant stimulus are shorter when a response is given on the same side as the visually presented stimulus than when the response side differs from the stimulus location. Hommel (2002) studied this effect and found that response times were indeed faster and more accurate if the stimulus and response side corresponded with each other, while response times were slower and less accurate when the stimulus and response side did not correspond. These effects were observed in both internal and external attention tasks.

## Reliability and validity in the literature

Van der Lubbe et al. (2014) suggested that one explanation for the differing results in the literature surrounding the mechanisms of internal and external attention could be the experimental design employed in certain studies (e.g. Kuo et al., 2009; Nobre et al., 2004), where effects such as repetition suppression (Mayrhauser et al., 2014) and automatic orienting (Tipples, 2002) might have been present. To test the mechanisms of internal and external attention, while minimizing the presence of these effects, Van der Lubbe et al. (2014) utilized a visual search task without stimulus repetition, which avoids the possibility of repetition suppression. In their experimental design, four different coloured stimuli (two circles and two squares) were presented in a frame. After a fixed amount of time, the frame changed into a colour that matched one of the stimuli. Depending on the condition the frame would appear before (pre-cue), during (simultaneous cue), or after (post-cue) the stimuli were displayed. Participants were expected to respond with either their right or left hand to indicate the form of the target. To analyse the data, the authors expanded traditional ERP analyses with the lateralized power spectra (LPS; Van der Lubbe and Utzerath, 2013). This method allowed them to avoid the shortcomings of the ERP method and assess induced lateralized activity that is not strongly bound to a time-locked event. The authors found increased ipsilateral α power at parieto-occipital areas for a brief duration during the pre-cue (~160 ms) and post-cue (~80 ms) conditions. A Simon effect for reaction time was also observed for all conditions. However, not all conditions displayed a Simon effect for accuracy. Their results provided support for the view that internal and external spatial attention share neural commonalities.

Questioning the stability of these effects over time, and the replicability of these effects Willems (2020) performed a follow-up study. They removed the simultaneous cue condition and added a post-cue condition with a larger array-cue interval, which allowed them to examine if the neural commonalities between internal and external attention are visible over longer periods. Willems also applied the LPS method to investigate induced activity. They found increased ipsilateral α power in parieto-occipital areas for a brief duration during the pre-cue condition (~160 ms). No effects for α lateralizations were found in the post-cue condition. The author observed a Simon effect for reaction time and accuracy in the pre-cue condition, but no effects were observed in the post-cue condition. While mechanisms in the brain are difficult to extract, the question as to why the experimental design used by Van der Lubbe et al. (2014) and Willems (2020) resulted in differing conclusions still remains.

In this study, it is proposed that the experimental design used in both studies (Van der Lubbe et al., 2014; Willems, 2020) still contained experimental noise that contributed to two different interpretations of the relationship between internal and external attention. The design used in both studies allowed participants to accurately perform post-cue trials by only memorising part of the stimulus array. Because the experiment only had two response options (square x circle), participants could easily memorise only one item per visual field and still accurately perform the trials, without having to fully employ internal attentional processes. For example, if the participants only memorised stimuli that had a square shape, and the cue indicated a circle as the target, they would have been able to respond that the shape of the target stimuli was not square. Using this method they would have been able to perform the post-cue tasks without the need to memorise the entire stimulus array. This allowed them to flexibly discard non-relevant items and reduce memory load (Souza and Oberauer, 2016).

The current paper will use an experimental design comparable to Van der Lubbe et al., (2014) and Willems (2020) to re-examine the results found by both studies. Some key changes in this design should ensure that internal attention specifically, and not any additional strategies, are being measured. Firstly, this study removed any irrelevant conditions by only employing the pre-cue and post-cue conditions, which measure external and internal attention respectively. The stimulus array was reduced to two stimuli, allowing for only one stimulus to be visible on either side of the fixation point. As an increase in stimulus set size appears to decrease the probability that all items will be retained in the working memory (Luck and Vogel, 2013; Zhang and Luck, 2008). This change ensured that participants memorise the full stimulus array.

Finally, the number of response options was increased to four. This change ensured that task difficulty will not become too low (Kraft et al., 2005).

## Goals and predictions

The main goal of this study is to re-examine the relationship between internal and external spatial attention by performing a successful conceptual replication with a modified experimental design. This study's main question is whether the use of a cueing task with this modified experimental design would show effects that support the view that internal and external spatial attention share their underlying mechanisms.

To answer this question, the current study used a modified experimental design based on the ones used in the study by Van der Lubbe et al., (2014) and Willems (2020). It is expected that the changes in the experimental design will lead to both behavioural and EEG results reflected in the α band that indicate commonalities between internal and external spatial attention. A clear Simon effect for both reaction time and response accuracy in both the pre-cue and post-cue conditions will also be expected**.** Since α activity tends to decrease contralateral to the focus of attention (Kelly et al., 2009; Thut et al., 2006; Worden et al., 2000), it is also expected to see increased ipsilateral power when compared to contralateral power in the α band in the pre-cue condition similar to the results found in the previous studies (Van der Lubbe et al., 2014; Willems, 2020). Additionally, it is also expected to see a similar increased ipsilateral versus contralateral power during the post-cue condition, similar to the effects observed by Van der Lubbe et al. (2014).

# Methods

## Participants

This study included 29 participants, consisting of 8 men and 21 women in the age range of 17-55 (*M= 22.7, SD = 7.2).* All participants were right-handed. Most participants (n=25) were University of Twente students studying Psychology or Communication Science, who received course credits for participating in this experiment. The remaining participants (n=4) were recruited using convenience sampling and participated voluntarily. None of the participants had visual impairments, and none reported a history of psychiatric or neurological impairments. The participants were also asked to wash their hair the night before the experiment to ensure an optimal EEG recording. The EOG data from two participants were not measured properly and their data were excluded from further analysis. People with pacemakers, dreadlocks, prior brain surgery, sleep problems, and drug and/or alcohol problems were excluded from participating. The study was approved by the Ethics Committee of the Faculty of Behavioural, Management, and Social Sciences at the University of Twente (request number 210676).

## Covid measurements

Due to the COVID-19 pandemic, the Dutch government implemented strict measures to minimize the risk of spreading the virus. Because it was necessary to be in close contact with the participants during the experiments, several rules were put in place to ensure the safety of everyone involved in this experiment. Participants were asked to disinfect their hands and were given FFP2 face masks when entering the EEG lab. The participants and researchers kept their distance whenever possible and the windows were regularly opened, allowing the air to circulate. Participants were explicitly asked not to come to if they were experiencing any COVID-19 symptoms or had been in close contact with someone infected.

## Stimuli, task and Procedure

The experiment took place in the RecogNice Lab on the Campus of the University of Twente. The participants received and signed the informed consent form and the BMS corona form. The BMS corona form contained questions to ensure that the COVID-19 measurements in the lab were being followed by the participants and researchers. The participants performed the Handedness questionnaire (Cohen, 2008) to determine handedness, the Ishihara test to check for colour-blindness, and the Freiburg Vision Test (Version 3.10.5) to check the visual acuity. After these pre-tests, the standard procedures as described by the EEG operations protocol (2021) were followed to prepare the participants for the EEG recording.

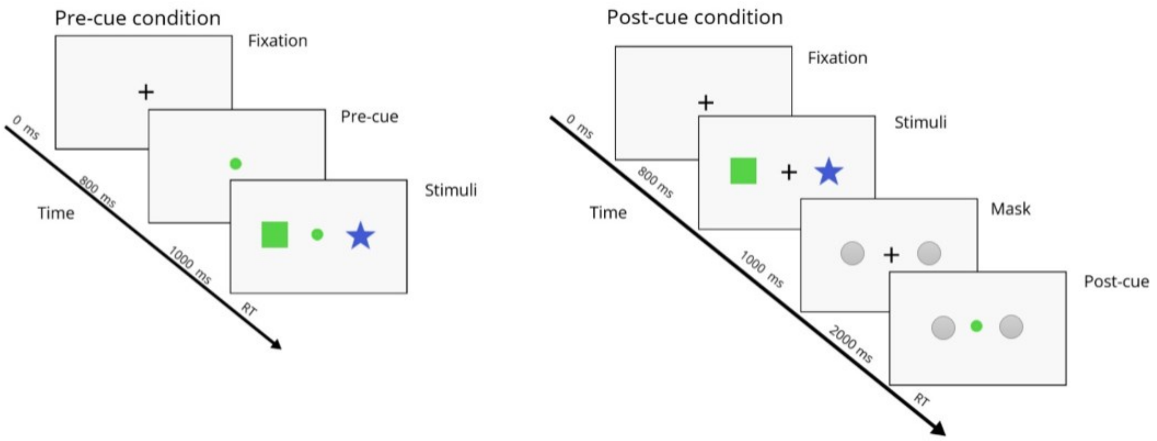

**Figure 1**. Experimental design based on the cueing paradigm task. An example of the experiment with the pre-cue condition (left) and the post-cue condition (right). Both trials depict two stimuli, a green square-shaped target and a blue star-shaped distractor with a cue indicating the target that the subjects had to respond to. Grey masks were used to cover the stimuli in the post-cue condition.

The experiment (See Figure 1) consisted of two conditions; pre-cue and post-cue. Half of the participants started with the pre-cue condition and the other half started with the post-cue condition. There were a total of 960 trials, and 64 practice trials. These were divided into 10 blocks of 96 trials and 2 practice blocks of 32 trials. Each experimental condition consisted of

5 real blocks and one practice block. At the beginning of each trial, the participants were presented with a fixation point. During the pre-cue condition, the cue appeared 800 ms after the fixation point was presented. Two stimuli (one target and one distractor) were randomly chosen from four potential targets and were presented 1000 ms after cue onset. The cue was visible until the response. During the post-cue condition, the stimuli appeared 800 ms after the fixation point was presented. They were visible for 1000 ms and masked afterwards. The cue appeared 2000 ms after the stimuli were masked.

During each trial, the participants had to react as fast and accurately as possible by identifying the target as signalled by the cue. The participants received instructions on the task they would be performing. The potential targets were connected to four keyboard buttons. To correct for any effects stemming from the position of the buttons, two buttons were located on the right side of the keyboard, and two were located on the left side. The practice trials during each condition allowed the participants to familiarize themselves with the task and the position of the relevant response buttons. During the practice trials, participants also received immediate feedback on their responses. After every block and between conditions, the participants were able to take short breaks. After the experiment participants were also asked if they used any specific strategy.

## Apparatus

The pre-tests were performed on an Acer Aspire laptop that ran on Windows 10. The EEG experiment was performed on a desktop computer that ran on Windows 7 in lean mode using a 22-inch LED monitor and a standard QWERTY keyboard. The stimuli were presented using Presentation software (Neurobehavioural Systems, Inc, Version 20.1). The EEG was recorded using 64 active electrodes (excluding the ground and EOG electrodes) attached using the actiCAP snap EEG system (Brain Products). The electrodes were placed using the standard 10-20 system at locations Fp1, Fp2, AF7, AF3, AFz, AF4, AF8, F7, F5, F3, F1, Fz, F2, F4, F6, F8, FT9, FT7, FC5, FC3, FC1, FCz, FC2, FC4, FC6, FT8, FT10, T7, C5, C3, C1, Cz, C2, C4, C6, T8, TP9, TP7, CP5, CP3, CP1, CPz, CP2, CP4, CP6, TP8, TP10, P7, P5, P3, P1, Pz, P2, P4, P6, P8, PO7, PO3, POz, PO4, PO8, O1, Oz and O2. Electrode TP8 was used as the online reference electrode. Eye movements were recorded using EOG electrodes. Vertical EOG was recorded by placing electrodes above and below the left eye. Horizontal EOG was recorded by placing electrodes at the outer canthi on both eyes. The signals from the EEG and EOG were amplified using an actiCHamp amplifier (Brain Products). The signals were registered on a separate desktop computer that ran on Windows 10 using BrainVision Recorder (Brain Products, Version 1.21.0403). The impedance was kept below 10 kΩ.

## Data analysis

*Behavioural measures*

Python (version 3.9)  was used to analyse the behavioural measures. Premature (<150 ms) and delayed reactions (>2000 ms) were removed. Average reaction times (RT) and the proportion of correct responses (PC) were computed as a function of Cue Condition (pre-cue/post-cue) and Correspondence (corresponding/non-corresponding). A 2x2 repeated measures ANOVA with the factors Cue Condition and  Correspondence was used to analyse the results. Mauchly's test for sphericity was applied.

*EEG measures*

The EEG recordings were pre-processed and analysed using a Python script (See Appendix A), and the MNE-python package (version 1.0.2). The EEG data from every participant was processed separately before being combined for further analysis. Every data file was montaged with the standard 10-20 montage to indicate the positions of the electrodes, while TP8 served as the reference electrode. The data received visual inspection to account for channels that contained a large amount of noise. Any subsequent bad channels were marked as 'bad' channels. Only channel TP10 from participant seven during the pre-cue condition was marked as 'bad'. Next, the independent component analysis (ICA) was used to clean the raw

EEG data. This included multiple steps. Since the ICA can change the properties of the raw EEG data, a copy of the raw data was made that was used for the ICA. A finite impulse response filter (0.1-30 Hz) was applied to the raw data. The ICA was applied using the FastICA algorithm. The extracted components were visually inspected. Then all the components related to eye movements were removed. On average 2.5 components (SD = 0.66) were extracted from each participant and removed. In the final step of the ICA, the ICA solution for each participant was applied to the copy of the data. The data was segmented into epochs from -500 to 2000 ms intervals relative to the event markers that signalled the relevant stimulus. Baseline correction was applied with a time interval of -500 ms to 0 ms (cue onset). Epochs that exceeded a peak-to-peak signal amplitude of 150e-6 were rejected. On average 8.25% of all epochs had to be dropped for every participant.

The LPS was constructed using the LPS method (Van der Lubbe & Utzerath, 2013). A Morlet wavelet (c=5) was used on the epochs to extract the power from the upper and lower α frequency bands. The following bands were specified: α1 (8-11 Hz) and α2 (11-14 Hz). After extracting the values, the individual averages for the left and right cues were computed. The resulting estimates were used to compute the lateralized indices [ipsilateral – contralateral] / [ipsilateral +contralateral] for both alpha bands and each cue side. These indices were created using one symmetrical electrode pair comprised of two electrodes (PO7 and PO8) located above the parieto-occipital areas. Finally, an average was computed across indices for both relevant sides [left index+ right index]/2, thereby creating the LPS. These values range from -1 to 1. Here positive values indicate that the power within a frequency band was larger above the hemisphere ipsilateral to the cued side compared to the contralateral side, while a negative value indicates the opposite. A value of zero indicates the absence of any hemispherical differences. The obtained indices were evaluated using one-sample *t*-tests. Both alpha bands were evaluated in 40 ms second intervals between 0 and 1000 ms after cue onset in the pre-cue condition, and after array onset in the post-cue condition, as the processing of the trials during the retro-cue condition can continue up to 1000 ms (Souza and Oberauer, 2016). A one-tailed $p<0.05$ criterion was used as the significance level, with p-values being adjusted using the Bonferroni method for multiple corrections.

## Results

### Behavioural results

The mean reaction times (RT) and the proportion of correct responses (PC) are displayed in Table 1. An ANOVA analysis on the RTs with factors Cue Condition and Correspondence showed that responses during both conditions were faster during corresponding than non-corresponding trials (904 vs 937 ms; $F(1,26)=25.9$, $p=<0.001$, $\eta_p^2$ $=0.004$). Responses were also faster during the pre-cue condition (908 ms) when compared to the post-cue condition (934 ms), although no interaction effects between RT and Cue Condition were observed ($F(1,26)=0.4$, $p=0.55$, $\eta_p^2=0.002$). The factors Correspondence and Cue Condition also showed no additional interaction effects ($F(1,26)=3.2$, $p=0.08$, $\eta_p^2$ $=0.001$). Separate t-test per cue condition confirmed the presence of a Simon effect (correspondence effect) in both the pre-cue ($t(26) = 41.2$, $p =<0.0125$) and post-cue ($t(26) = 30.8$, $p = <0.001$)

An ANOVA on PC with the factors Cue Condition and Correspondence showed that responses in the pre-cue condition were more often correct when compared to the post-cue condition ($F(1,26)=14.7$, $p=<0.001$, $\eta_p^2$ $=0.081$). Responses to corresponding trials were also more often correct when compared to non-corresponding trials ($F(1,26)=21.6$, $p=<0.001$, $\eta_p^2$ $=0.011$). No additional effects between the factors Correspondence and Cue Condition were observed ($F(1,26)=3.9$, $p=0.057$, $\eta_p^2$ $=0.001$). Separate t-test per cue condition confirmed the presence of a Simon effect in both the pre-cue ($t(26) = 63.8$, $p =<0.001$) and post-cue ($t(26) = 58.5$, $p = <0.001$).

**Table 1.**

Mean reaction time (RT) and proportion of correct answers (PC) as a function of Condition and Correspondence

| Condition | RT(ms) | | PC(%) | |
|---|---|---|---|---|
| | Corr | Non-corr | Corr | Non-corr |
| Pre-cue | 901(30) | 914(32) | 92(2) | 90.6(2.1) |
| Post-cue | 908(46) | 959(50) | 86.5(2) | 83.5(2.2) |

*Note*. Standard errors are in brackets. Corr = corresponding, Non-corr = non-corresponding.

**EEG results**

An overview of the significant time windows for the electrode pair PO7/8 is shown in Table 2 The analysis using a one-sample t-test showed that both the pre-cue and post-cue conditions exhibited significant deviations from zero. During the pre-cue condition, the lower alpha band revealed a posterior effect, showing larger ipsilateral than contralateral power (140-980 ms; (t(26)=51.3, p= <0.001) ). A comparable effect was observed in the upper alpha band (140-820 ms; (t(26)=70.4, p= <0.001)). Similar to the pre-cue condition, the lower alpha band during the post-cue condition also showed a similar, longer-lasting posterior effect(100-980 ms; ((t(26)=38.4, p= <0.001)), showing stronger ipsilateral versus contralateral power. The upper alpha band during this condition revealed two windows where a posterior effect was visible. Effects are reflected in both an early (140-260 ms; (t(26)=7.8, p= <0.001) and later (340-980 ms; (t(26)=54.5, p= <0.001) time window. Both effects also reflected a stronger ipsilateral versus contralateral power.

**Table 2**

LPS results

| Condition | Band(site) | LPS time window | Polarity |
|---|---|---|---|
| Pre-cue | α1(PO7/8) | 140-980 | Positive |
| | α2(PO7/8) | 140-820 | Positive |
| Post-cue | α1(PO7/8) | 100-980 | Positive |
| | α2(PO7/8) | 140-260 | Positive |
| | α2(PO7/8) | 340-980 | Positive |

*Note*. Positive indicates a stronger ipsilateral versus contralateral power

Figure 2 displays the LPS indices for the lower and upper alpha bands during both conditions. For the pre-cue condition, there are effects in both the lower and upper alpha band, which indicate a stronger ipsilateral versus contralateral power after cue onset. In both bands, this effect is followed by an inverse effect around 1000 ms, where the contralateral power is stronger compared to ipsilateral. The LPS power in the upper alpha band also appears to be stronger compared to the lower alpha band. For the post-cue condition, the effects also show an increased ipsilateral versus contralateral power in both the lower and upper alpha bands. This effect is mostly sustained throughout the examined time window in the lower alpha band. Similar to the pre-cue condition, the power in the alpha band also seems to be stronger in the upper alpha band compared to the lower alpha band.

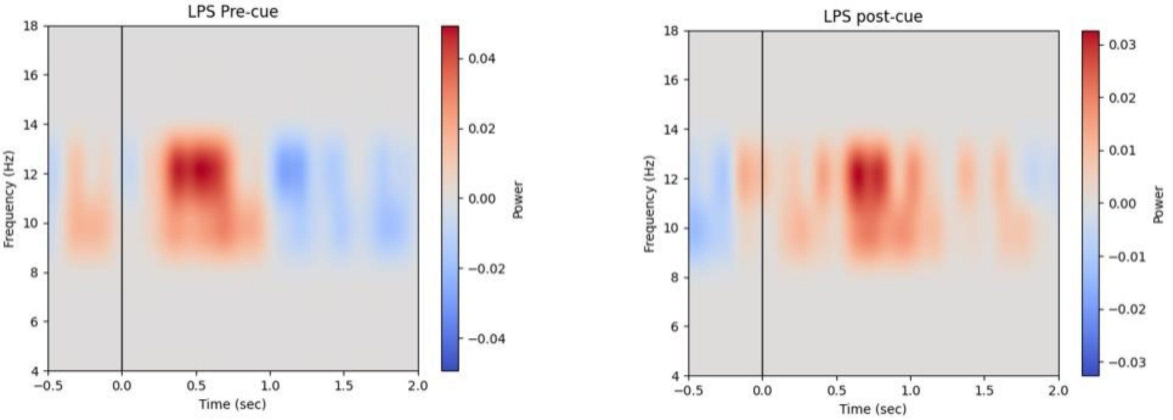

**Figure 2.** LPS indices displaying the lower (~8-11 Hz) and upper (~11-14 Hz) alpha bands during the pre-cue (left) and post-cue (condition). The line on the x-axis indicates cue onset in the pre-cue condition and stimuli-onset in the post-cue condition. Positivity indicates a stronger ipsilateral power when compared to contralateral, while negativity indicates the opposite.

# Discussion

## General discussion

This study set out to re-examine the relationship between internal and external spatial attention. The aim was to investigate whether a modified experimental design based on previous studies (Van der Lubbe et al., 2014; Willems, 2020) would show results that support the view that internal and external attention share underlying mechanisms.

The behavioural results indicate the presence of a clear Simon effect in both the pre-cue and post-cue conditions. Reaction times during both conditions were faster when the stimuli appeared on the corresponding side and slower when the stimuli appeared on the non-corresponding side. Trials with corresponding stimuli were also more often correct compared to non-corresponding stimuli. These results are comparable with previous findings (Hommel, 2002). This suggests that the process of internally retrieving an item from memory facilitates a response that corresponds to the original location of that item. A process that is comparable to externally viewing an item.

The LPS analysis showed an increased ipsilateral versus contralateral power in both the lower and upper posterior α bands. This effect was observed during both the pre-cue and post-cue conditions. These findings suggest a reduced inhibition of the contralateral hemisphere and/or an increased inhibition of the ipsilateral hemisphere (Kelly, 2009; Sauseng et al., 2005; Van der Lubbe and Utzerath, 2013). This implies that when targets and distractors are presented in separate hemifields, α synchronizes during the maintenance interval over the hemisphere that is processing the distractor and desynchronizes over the hemisphere that is processing the targets (Kelly et al., 2006). This suggests that externally viewing an item and internally retrieving an item from memory are processes where α power is involved in suppressing or regulating visual perception, even in the absence of visual stimuli.

The visual results (See Figure 2) indicate that the lateralized α power is stronger in the $\alpha_2$ band compared to the $\alpha_1$ band.  One explanation for the discrepancy in results between these bands could be related to a distinction in task processing in the α band. Studies have suggested that the $\alpha_1$ band is more likely to reflect general tasks, such as attentional processes, whereas the $\alpha_2$ band reflects specific tasks, such as semantic processing (Neubauer and Fink, 2009).

An inverse effect is visible when comparing the pre-stimulus LPS power between both conditions. A contralateral versus ipsilateral power is displayed during the pre-cue condition, while the post-cue condition shows the opposite. However, rather than being directly linked to spatial attention, these fluctuations in pre-stimulus α power could involve global changes in visual processing as suggested by Foster and Awh (2019). When comparing the two conditions after stimulus onset, it seems that the pre-cue condition shows an earlier deviation in lateralized α power compared to the post-cue condition. This difference could be the result of variations in memory load and perceptual selection (Tas et al.,2016). Cues measuring external attention were always presented at the beginning of a trial and did not require memorisation, while internal attention cues were presented after the stimuli had to be memorised. Indicating that internal attention occurs in the context of working memory (Tas et al., 2016).

Considering the conceptual replication, this study was able to observe that internal and external spatial attention share similar mechanisms. This study observed a Simon effect across conditions for both reaction time and accuracy, while previous studies (Van der Lubbe et al. 2014; Willems, 2020) only observed this effect in some conditions. This suggests that the discrepancies in behavioural results from these studies could have been due to the additional noise in the post-cue condition. The lateralized α power during both conditions is more pronounced and visible over a longer time window when compared to the results of Van der Lubbe et al. (2014) and Willems (2020). It follows that the changes made in the experimental design do not only measure internal and external attention as intended, but have also led to a more robust paradigm that still adds support for the view of shared mechanics regarding internal and external attention.

## Conclusion

The results show that the changes made to the experimental design have led to the presence of a clear Simon effect, and pronounced LPS effects in both conditions. It also highlights that the modified experimental design has resulted in more pronounced effects compared to the studies it tried to replicate. These results add further support to the view that internal and external spatial attention share their underlying mechanisms. Lastly, this study also supports the suggestion that the experimental design played an important factor as to why both studies resulted in varying conclusions about the relationship between internal and external attention.

## Limitations and future studies

One of the main limitations of this study is that it only examined the effects in the α band in the posterior parieto-occipital area, while disregarding potential activity at other sites and bands. Studies have shown that posterior α desynchronizations are coupled to frontal oscillations in the delta (e.g. de Vries et al., 2019) and theta band (Reinhart and Woodman, 2014). This suggests that the modulations in the α band when processing spatial attention may only occur because another region directs it (Woodman, 2022).

It can be interesting for future studies to consider using this paradigm in multimodal settings. The presence of only two visual stimuli that need to be retained in the working memory allows for more opportunities to combine this cueing task with other for example, auditory tasks.

## References

Chun, M.M. (2011). Visual working memory as visual attention sustained internally over time. *Neuropsychologia,* 49, 1407-1409. https://doi.org/10.1016/j.neuropsychologia.2011.01.029

Clayton, M. S., Yeung, N., & Cohen Kadosh, R. (2018). The many characters of visual alpha oscillations. *European Journal of Neuroscience*, *48*(7), 2498–2508. https://doi.org/10.1111/ejn.13747

Cohen, M.S. (2008). *Handedness Questionnaire.* Brainmapping. https://www.Brainmapping. org/shared/ Edinburgh.php

de Vries, I., Savran, E., van Driel, J., & Olivers, C. (2019). Oscillatory Mechanisms of Preparing for Visual Distraction. *Journal of Cognitive Neuroscience*. 31. 1-22. 10.1162/jocn_a_01460.

Foerster, R. M., & Schneider, W. X. (2018). Involuntary top-down control by search-irrelevant features: Visual working memory biases attention in an object-based manner. *Cognition*, *172*(December 2017), 37–45. https://doi.org/10.1016/j.cognition.2017.12.002

Foster, J. J., & Awh, E. (2019). The role of alpha oscillations in spatial attention: limited evidence for a suppression account. *Current Opinion in Psychology*, *29*, 34–40. https://doi.org/10.1016/j.copsyc.2018.11.001

Hommel, B. (2002). Responding to object files: Automatic integration of spatial information revealed by stimulus-response compatibility effects. *The Quarterly Journal of Experimental Psychology Section A*, *55*(2), 567-580. doi:10.1080/02724980143000361

Hörsting, J., & Titsing, D. (2021). *EEG operations protocol: Set-up and recording.* University of Twente. https://bmslab.utwente.nl/wp-content/uploads/2021/06/EEG-Manual.pdf

Kelly, S. P., Lalor, E. C., Reilly, R. B., & Foxe, J. J. (2006). Increases in alpha oscillatory power reflect an active retinotopic mechanism for distracter suppression during sustained visuospatial attention. *Journal of Neurophysiology*, *95*(6), 3844–3851. https://doi.org/10.1152/jn.01234.2005

Kraft, A., Müller, N. G., Hagendorf, H., Schira, M. M., Dick, S., Fendrich, R. M., & Brandt, S. A. (2005). Interactions between task difficulty and hemispheric distribution of attended locations: Implications for the splitting attention debate. *Cognitive Brain Research*, *24*(1), 19–32. https://doi.org/10.1016/j.cogbrainres.2004.12.004

Kuo, B. C., Rao, A., Lepsien, J., & Nobre, A. C. (2009). Searching for targets within the spatial layout of visual short-term memory. *Journal of Neuroscience*, *29*(25), 8032–8038. https://doi.org/10.1523/JNEUROSCI.0952-09.2009

Lepsien, J., Griffin, T. I. C., Devlin, J. T., & Nobre, A. C. (2005). *Directing spatial attention in mental representations : Interactions between attentional orienting and working-memory load*. *26*, 733–743. https://doi.org/10.1016/j.neuroimage.2005.02.026

Luck, S. J., & Vogel, E. K. (2013). Visual working memory capacity: From psychophysics and neurobiology to individual differences. *Trends in Cognitive Sciences*, *17*(8), 391–400. https://doi.org/10.1016/j.tics.2013.06.006

Ma, W. J., Husain, M., & Bays, P. M. (2014). Changing concepts of working memory. *Nature Neuroscience*, *17*(3), 347–356. https://doi.org/10.1038/nn.3655

Mayrhauser, L., Bergmann, J., Crone, J., & Kronbichler, M. (2014). Neural repetition suppression: Evidence for perceptual expectation in object-selective regions. *Frontiers in Human Neuroscience*, *8*(1 APR), 1–8. https://doi.org/10.3389/fnhum.2014.00225

Mössing, W. A., & Busch, N. A. (2020). Lateralized alpha oscillations are irrelevant for the behavioral retro-cueing benefit in visual working memory. In *PeerJ* (Vol. 2020, Issue 6). https://doi.org/10.7717/peerj.9398

Neubauer, A. C., & Fink, A. (2009). Intelligence and neural efficiency. *Neuroscience and biobehavioral reviews*, *33*(7), 1004–1023. https://doi.org/10.1016/j.neubiorev.2009.04.001

Nobre, A. C., Coull, J. T., Maquet, P., Frith, C. D., Vandenberghe, R., & Mesulam, M. M. (2004). Orienting attention to locations in perceptual versus mental representations. *Journal of Cognitive Neuroscience*, *16*(3), 363-373. doi:10.1162/089892904322926700

Oberauer, K. (2019). Working memory and attention - A conceptual analysis and review. *Journal of Cognition*, *2*(1), 1–23. https://doi.org/10.5334/joc.58

Pavlov, Y. G., Adamian, N., Appelhoff, S., Arvaneh, M., Benwell, C. S. Y., Beste, C., Bland, A. R., Bradford, D. E., Bublatzky, F., Busch, N. A., Clayson, P. E., Cruse, D., Czeszumski, A., Dreber, A., Dumas, G., Ehinger, B., Ganis, G., He, X., Hinojosa, J. A., … Mushtaq, F. (2021). #EEGManyLabs: Investigating the replicability of influential EEG experiments. *Cortex*, *144*, 213–229. https://doi.org/10.1016/j.cortex.2021.03.013

Reinhart, R.M.G. & Woodman, G.F. (2014). Oscillatory Coupling Reveals the Dynamic Reorganization of Large-scale Neural Networks as Cognitive Demands Change. *Journal of Cognitive Neuroscience.* 26 (1): 175–188. doi: https://doi.org/10.1162/jocn_a_00470

Sauseng, P., Klimesch, W., Stadler, W., Schabus, M., Doppelmayr, M., Hanslmayr, S., Gruber, W. R., & Birbaumer, N. (2005). A shift of visual spatial attention is selectively associated with human EEG alpha activity. *European Journal of Neuroscience*, *22*(11), 2917–2926. https://doi.org/10.1111/j.1460-9568.2005.04482.x

Scheibner, H. J., Bogler, C., Gleich, T., Haynes, J. D., & Bermpohl, F. (2017). Internal and external attention and the default mode network. *NeuroImage*, *148*(January), 381–389. https://doi.org/10.1016/j.neuroimage.2017.01.044

Simon, J. R. (1969). Reactions toward the source of stimulation. *Journal of Experimental Psychology*, *81*(1), 174-176. doi:10.1037/h0027448

Souza, A. S., & Oberauer, K. (2016). In search of the focus of attention in working memory : 13 years of the retro-cue effect. *Attention, Perception, & Psychophysics*, 1839–1860. https://doi.org/10.3758/s13414-016-1108-5

Tas, A. C., Luck, S. J., & Hollingworth, A. (2016). The relationship between visual attention and visual working memory encoding: A dissociation between covert and overt orienting. In *Journal of Experimental Psychology: Human Perception and Performance* (Vol. 42, Issue 8, pp. 1121–1138). https://doi.org/10.1037/xhp0000212

Thut, G., Nietzel, A., Brandt, S. A., & Pascual-Leone, A. (2006). α-Band electroencephalographic activity over occipital cortex indexes visuospatial attention bias and predicts visual target detection. *Journal of Neuroscience*, *26*(37), 9494–9502. https://doi.org/10.1523/JNEUROSCI.0875-06.2006

Tipples, J. Eye gaze is not unique: Automatic orienting in response to uninformative arrows. *Psychonomic Bulletin & Review* 9, 314–318 (2002). https://doi.org/10.3758/BF03196287

Tsubomi, H., Fukuda, K., Watanabe, K., & Vogel, E. K. (2013). Neural limits to representing objects still within view. *The Journal of Neuroscience, 33*(19), 8257–8263. https://doi.org/10.1523/JNEUROSCI.5348-12.2013

Van Der Lubbe, R. H. J., Bundt, C., & Abrahamse, E. L. (2014). Internal and external spatial attention examined with lateralized EEG power spectra. *Brain Research*, *1583*(1), 179–192. https://doi.org/10.1016/j.brainres.2014.08.007

Van der Lubbe, R., & Utzerath, C. (2013). Lateralized power spectra of the EEG as an index of visuospatial attention. *Advances in Cognitive Psychology*, *9*(4), 184-201. doi:10.5709/acp-0144-7

Wallis, G., Stokes, M., Cousijn, H., Woolrich, M., & Nobre, A. C. (2015). Frontoparietal and Cingulo-opercular Networks Play Dissociable Roles in Control of Working Memory. *Journal of cognitive neuroscience*, *27*(10), 2019–2034. https://doi.org/10.1162/jocn_a_00838

Willems, N. N. (2020). *Overlap between internal and external spatial attention over longer time intervals*. [Master's thesis, University of Twente]. University of Twente Repository. https://essay.utwente.nl/80365/

Woodman, G. F., Wang, S., Sutterer, D. W., Reinhart, R. M. G., & Fukuda, K. (2022). Alpha suppression indexes a spotlight of visual-spatial attention that can shine on both perceptual and memory representations. *Psychonomic Bulletin and Review*, *29*(3), 681–698. https://doi.org/10.3758/s13423-021-02034-4

Worden, M. S., Foxe, J. J., Wang, N., & Simpson, G. V. (2000). Anticipatory biasing of visuospatial attention indexed by retinotopically specific alpha-band electroencephalography increases over occipital cortex. In *The Journal of neuroscience :*

*the official journal of the Society for Neuroscience* (Vol. 20, Issue 6). https://doi.org/10.1523/jneurosci.20-06-j0002.2000

Zhang, W., & Luck, S. J. (2008). Discrete fixed-resolution representations in visual working memory. *Nature*, *453*(7192), 233–235. https://doi.org/10.1038/nature06860

## Appendix A

## Python script

This appendix contains the full script that was used to process the EEG data. This script can also be found in the online Github repository at https://github.com/IvaniaJahangier/Internal_external_attention_analysis.

```
#!/usr/bin/env python
# -*- coding: utf-8 -*-
#------------------------------------------------------------------------
# @Author: Ivania Jahangier
# @Date: 12-10-2021
# ------------------------------------------------------------------------

#LOADING PACKAGES
import pandas as pd
import seaborn as sns
import glob
import os as os
import sklearn
from picard import picard
from turtle import title
import numpy
import numpy as np
from scipy.special import logsumexp
import pathlib
from pathlib import Path
import PyQt5
import matplotlib.pyplot
import matplotlib
matplotlib.use('Qt5Agg')
import matplotlib.pyplot as plt
from tabulate import tabulate
import mne
from mne.time_frequency import tfr_morlet, tfr_array_morlet
from mne import event
from mne.io import concatenate_raws, read_raw_edf
from mne import concatenate_events, find_events
from mne.preprocessing import ICA, create_eog_epochs, create_ecg_epochs, corrmap
from mne.baseline import rescale
from mne.stats import bootstrap_confidence_interval
from mne.viz import plot_topomap
from mne.event import define_target_events
from matplotlib.collections import PatchCollection
from matplotlib.patches import Rectangle
import matplotlib.patches as patches
```

```
from numpy import unravel_index
from statsmodels.stats.anova import AnovaRM
import pingouin as pg
import scipy.stats as stats
import statsmodels.stats.multitest as smm

# GLOBAL VARIABLES
reject_criteria = dict(eeg=150e-6)  # 100uV
flat_criteria = dict(eeg=1e-7)  # 1uV
interval = (-0.5, 0)
sfreq = 1000
tmin = 0
tmax = 3
fill_na = 99
wave_cycles = 5
frequenciesa1 = np.arange(8, 11, 1) # a1 band
frequenciesa2 = np.arange(11, 14, 1) # a2 band
frequenciesa = np.arange(4, 19, 1) # plot
picks = ['PO7', 'PO8']
number_of_trials = 240 #for every RVF/LVF per condition (240x4=960 trials)
shape = (0,2501)
dtype = float

#LISTS
#These lists are used during both conditions
PRO_files = []
RETRO_files = []
corr_list = []
corresponding = []
subjectnumber = []
condition = []
pc_list = []
p_correspondence = []

# These lists will be used for the pre-cue condition.
a1_PRO_L_PO7 = np.empty(shape = shape, dtype = dtype)
a1_PRO_L_PO8 = np.empty(shape = shape, dtype = dtype)
a1_PRO_R_PO7 = np.empty(shape = shape, dtype = dtype)
a1_PRO_R_PO8 = np.empty(shape = shape, dtype = dtype)
a2_PRO_L_PO7 = np.empty(shape = shape, dtype = dtype)
a2_PRO_L_PO8 = np.empty(shape = shape, dtype = dtype)
a2_PRO_R_PO7 = np.empty(shape = shape, dtype = dtype)
a2_PRO_R_PO8 = np.empty(shape = shape, dtype = dtype)

# These lists will be used for the post-cue condition.
a1_RETRO_L_PO7 = np.empty(shape = shape, dtype = dtype)
a1_RETRO_L_PO8 = np.empty(shape = shape, dtype = dtype)
a1_RETRO_R_PO7 = np.empty(shape = shape, dtype = dtype)
```

```
 a1_RETRO_R_PO8 = np.empty(shape = shape, dtype = dtype)
 a2_RETRO_L_PO7 = np.empty(shape = shape, dtype = dtype)
 a2_RETRO_L_PO8 = np.empty(shape = shape, dtype = dtype)
 a2_RETRO_R_PO7 = np.empty(shape = shape, dtype = dtype)
 a2_RETRO_R_PO8 = np.empty(shape = shape, dtype = dtype)


#LPS FUNCTION
#Here we are creating a function that calculates the LPS indices
#the formula is adapted from Van der Lubbe and Utzerath (2013)

def Left_cues (L_ipsi, L_contra):
    LPS_Left_cues = (L_ipsi-L_contra)/(L_ipsi+L_contra)
    return LPS_Left_cues

def Right_cues (R_ipsi, R_contra):
    LPS_Right_cues = (R_ipsi-R_contra)/(R_ipsi+R_contra)
    return LPS_Right_cues

def N_LPS(L_ipsi, L_contra, R_ipsi, R_contra):
    LPS_Left_cues = (L_ipsi-L_contra)/(L_ipsi+L_contra)
    LPS_Right_cues = (R_ipsi-R_contra)/(R_ipsi+R_contra)
    final_LPS = (LPS_Left_cues+LPS_Right_cues)/2
    return final_LPS


ICA ANALYSIS
# Setting the ICA to False. This way the code will not go through the ICA process repeatedly.
completeICA = False

#Change the working directory to open the raw data
working_directory = os.chdir("C:/Users/ivani/Desktop/dir_eeg_analysis/EEG.data")

#Create an empty list to store the raw files
ICA_solutions = [] # move to the lists section
for item in glob.glob("*.eeg"):
    ICA_solutions.append(item)

for item in ICA_solutions:
    if completeICA:
        raw = mne.io.read_raw_brainvision(item, preload=True)
        anno = mne.read_annotations(item, sfreq='auto', uint16_codec=None)

#plotting the data to check for bad channels
        raw_RETRO.plot()
        raw_RETRO.plot_psd(fmax=500)
```

```
# Setting the montage to the extended 10-20system, which is how we indicate how the electr
odes were positioned
# On the head. The data was recorded by setting the channel TP8 as the reference electrode.

    raw.set_montage(mne.channels.make_standard_montage('standard_1020'))
    raw = mne.add_reference_channels(raw_PRO, ref_channels=['TP8'])
    raw_PRO.set_eeg_reference(ref_channels='average')

# Plot to show the waves and their source on the head
    raw_PRO.plot_sensors(kind='topomap', show_names=True, title='ProPlot')

# We are filtering all major frequencies to enhance the quality of the data.
# Since these frequency drifts can make it hard to create an ICA solution.
    raw.load_data().filter(l_freq=0.1, h_freq=30)
    print('step 1 preparation complete')

# Starting the ICA
# Since the ICA can change our raw data We start of with creating a copy of our data
    raw_copy= raw.copy()
    raw_copy.load_data().filter(l_freq=0.1, h_freq=30)

# Variables that are used in the ICA
    nu_components = 63
    algorithm = 'fastica'
    random_seed = 91 #Setting a random state ensures that we get the same random value
for every train and test datatesets. Otherwise it would set to non everytime and generate diff
erent values each time.
    ica = mne.preprocessing.ICA( n_components= nu_components, method= algorithm, ran
dom_state= random_seed )
    ica.fit(raw_copy)

# Instead of manually selecting which ICs to exclude, we use dedicated EOG sensors as a "pat
tern" to check the ICs against
    ica.exclude =[] #first we make an empty exclude list
    eog_indices, eog_scores = ica.find_bads_eog(raw_PRO_copy, ['hEOG', 'vEOG'])#automat
ically find the ICs that best match the EOG signal
    ica.exclude = eog_indices#excludes artefacts matching eog signals that are added to the
exclude list

# Barpolt of ICA component "EOG" match scores
    fig_1 = ica.plot_scores(eog_scores)
    plt.close(fig_1)
    fig_1.savefig(r'C:/Users/ivani/Desktop/Resultseeg/ICA_PRO_component_Score.png', ov
erwrite=False)

# Plot diagnostics
    fig_2 = ica.plot_properties(raw_PRO_copy, picks=eog_indices) #save this image
```

```
# Visual presentation ICA components on head
        fig_4 = ica.plot_components(ch_type = 'eeg')


# Plot ICs applied to raw data, with EOG matches highlighted
        fig_3 = ica.plot_sources(raw_PRO_copy)
        fig_3.figure.savefig(r'C:/Users/ivani/Desktop/Resultseeg/ICA_PRO_plot_sources.png', ov
erwrite=False)

# Apply the ICA
        ica.apply(raw_copy)

# Final check of raw data, here the data should be full cleaned
        fig_5 = raw_PRO_copy.plot(title='Final check')
        fig_5.figure.savefig(r'C:/Users/ivani/Desktop/Resultseeg/raw_PRO_final_check.png', ov
erwrite=True)

# If the data is fully cleaned, the ICA solution can be saved
        raw_copy.save(r'C:/Users/ivani/Desktop/Resultseeg/raw_cleaned.fif', overwrite=True)
    else:
        print('Ica is completed')

#PRE-CUE ANALYSIS

# This loop consist of two main parts
# the first part involved creating epochs, TFRs and saving the data to arrays
# This conderns the PO7/PO8 electrodes for the a1 and a2 band
# This data will be used to calculate the LPS later on
# The second part involves the data analysis of the behavioral data.
# The events are merged and saved to lists that will be used in a pandas dataframe

#Change the working directory to find the cleaned files
working_directory = os.chdir("C:/Users/ivani/Desktop/dir_eeg_analysis/ICA_Solutions/PRO")

# Save the data for this condition in one list.
for item in glob.glob("*.fif"):
    PRO_files.append(item)

# Variables for this section
subject = 1
current_condition = 'pre-cue'

for item in PRO_files:
    PRO_raw = mne.io.read_raw_fif(item, preload=True)

    events_PRO, event_dict_pro = mne.events_from_annotations(PRO_raw)
```

```
   events_PRO, notNeeded = mne.events_from_annotations(PRO_raw)

   event_dict_PRO = {"fixation start": 2, 'cue': 3,
              'target': 4, 'correctresponse/1': 101,
              'correctresponse/2': 102, 'correctresponse/3': 103,
              'correctresponse/4': 104, 'incorrectresponse': 105,
              'PRO_LVF/L': 11, 'PRO_LVF/R': 12,
              'PRO_RVF/L': 21, 'PRO_RVF/R': 22}


#Create epochs
   epochs_PRO = mne.Epochs(PRO_raw, events_PRO, event_id=event_dict_PRO, tmin=-0.5,
                           tmax=2.0, preload=True)
   epochs_PRO.drop_bad(reject=reject_criteria, flat=flat_criteria)
   epochs_PRO.apply_baseline(interval)

# The TFR analysis is performed twice, one for the a1 and one for the a2 band.
# We start with the manipulations in the a1 band. Using a morlet wavelet
   L_tfr_a1 = tfr_morlet(epochs_PRO['PRO_LVF'], freqs=frequenciesa1, picks=picks,
              return_itc=False, n_cycles=wave_cycles, average = True, output= 'power')

   R_tfr_a1 = tfr_morlet(epochs_PRO['PRO_RVF'], freqs=frequenciesa1, picks=picks,
              return_itc=False, n_cycles=wave_cycles, average = True, output= 'power')


# Save the data from the TFR into separate arrays
   a1_PO7 = L_tfr_a1.data[0,:,:]
   a1_PO8 = L_tfr_a1.data[1,:,:]
   a1_PO72 = R_tfr_a1.data[0,:,:]
   a1_PO82 = R_tfr_a1.data[1,:,:]


# Take the average from the 4 frequencies for the band per time point (2501 points) and
compute the mean.
# Creating one mean for the entire frequency band
   m_a1_PO7 = np.mean(a1_PO7, axis=0, keepdims=True)
   m_a1_PO8 = np.mean(a1_PO8, axis=0, keepdims=True)
   m_a1_PO72 = np.mean(a1_PO72, axis=0, keepdims=True)
   m_a1_PO82 = np.mean(a1_PO82, axis=0, keepdims=True)

#save the mean values to arrays
   a1_PRO_L_PO7 = np.append(a1_PRO_L_PO7, m_a1_PO7, axis=0)
   a1_PRO_L_PO8 = np.append(a1_PRO_L_PO8, m_a1_PO8, axis=0)
   a1_PRO_R_PO7 = np.append(a1_PRO_R_PO7, m_a1_PO72, axis=0)
   a1_PRO_R_PO8 = np.append(a1_PRO_R_PO8, m_a1_PO82, axis=0)

# Manipulations in the a2 band. Using a morlet wavelet
```

```
   L_tfr_a2 = tfr_morlet(epochs_PRO['PRO_LVF'], freqs=frequenciesa2, picks=picks, return_it
c=False,
                     n_cycles=wave_cycles, average = True, output= 'power')  # power ipv complex

   R_tfr_a2= tfr_morlet(epochs_PRO['PRO_RVF'], freqs=frequenciesa2, picks=picks, return_it
c=False,
                     n_cycles=wave_cycles, average = True, output= 'power')

# Save the data from the TFR into separate arrays
   a2_PO7 = L_tfr_a2.data[0,:,:]
   a2_PO8 = L_tfr_a2.data[1,:,:]
   a2_PO72 = R_tfr_a2.data[0,:,:]
   a2_PO82 = R_tfr_a2.data[1,:,:]

# Takes the average from the 4 frequencies for the band per time point (2501 points) and
compute the mean.
# Creating one mean for the entire frequency band
   m_a2_PO7 = np.mean(a2_PO7, axis=0, keepdims=True)
   m_a2_PO8 = np.mean(a2_PO8, axis=0, keepdims=True)
   m_a2_PO72 = np.mean(a2_PO72, axis=0, keepdims=True)
   m_a2_PO82 = np.mean(a2_PO82, axis=0, keepdims=True)


# Save the mean values to arrays
   a2_PRO_L_PO7 = np.append(a2_PRO_L_PO7, m_a2_PO7, axis=0)
   a2_PRO_L_PO8 = np.append(a2_PRO_L_PO8, m_a2_PO8, axis=0)
   a2_PRO_R_PO7 = np.append(a2_PRO_R_PO7, m_a2_PO72, axis=0)
   a2_PRO_R_PO8 = np.append(a2_PRO_R_PO8, m_a2_PO82, axis=0)


# Now we start with creating the data for the pandas dataframe
# This will be done for both the corresponding and non-corresponding trials
# First the correct answers are all merged
# Then the corresponding trials (visual fieldxhandedness) are merged
# Lastly the non-corresponding trials are merged

   events_PRO = mne.merge_events(events_PRO, [101,102,103,104], 80,
                                           replace_events=False)
   events_PRO = mne.merge_events(events_PRO, [11,22], 13, replace_events=False)
   events_PRO = mne.merge_events(events_PRO, [12,21], 14, replace_events=False)


# These events sets up the reaction times for the corresponding correct responses
# Setting up the RTs. Which starts with the define-target function.
# This allows us to define new targets based on co-occurring events and the time (in ms)
#between  the target and the reference id events
```

```
   c1_events, nan_rt1 = define_target_events(events = events_PRO, reference_id = 13,
                        target_id = 80, sfreq = sfreq,
                        tmin = tmin, tmax=3, new_id = 25, fill_na=fill_na)

# remove the nan values
   rt1 = nan_rt1[np.logical_not(np.isnan(nan_rt1))]

# Calculate the percentage of correct answers (pc)
   pc = len(rt1)/number_of_trials*100

# Appending the RT, corresponding side, participantnumber, condition, and pc
   corr_list.append(np.mean(rt1, axis=0))
   corresponding.append('corr')
   subjectnumber.append(subject)
   condition.append(current_condition)
   pc_list.append(pc)

# These events set up the reaction times for the non-corresponding correct responses
# Setting up the RTs. Which starts with the define-target function.
# This allows us to define new targets based on co-occurring events and the time (in ms)
# between  the target and the reference id events

   c2_events, nan_rt2 = define_target_events(events = events_PRO, reference_id = 14,
                         target_id = 80, sfreq = sfreq, tmin = tmin, tmax=3, new_id = 15,
                         fill_na=fill_na)


# Remove nan-values
   rt2 = nan_rt2[np.logical_not(np.isnan(nan_rt2))]

# Calculate the percentage of correct answers (pc)
   pc = len(rt2)/number_of_trials*100

# Appending the RT, non-corresponding side, participant number, condition,and pc
   corr_list.append(np.mean(rt2, axis=0))
   corresponding.append('noncorr')
   subjectnumber.append(subject)
   condition.append(current_condition)
   pc_list.append(pc)


   subject = subject+1


#POST-CUE ANALYSIS
# This loop consist of two main parts
# he first part involved creating epochs, TFRs and saving the data to arrays
# This conderns the PO7/PO8 electrodes for the a1 and a2 band
# This data will be used to calculate the LPS later on
# The second part involves the data analysis of the behavioral data.
# The events are merged and saved to lists that will be used in a pandas dataframe

#Change the working directory to find the cleaned files
working_directory = os.chdir("C:/Users/ivani/Desktop/dir_eeg_analysis/ICA_Solutions/
RETRO")

# Save the data for this condition in one list.
for item in glob.glob("*.fif"):
   RETRO_files.append(item)

# Variables needed for this section
subject = 1
current_condition = 'post-cue'

for item in RETRO_files:
   RETRO_raw = mne.io.read_raw_fif(item, preload=True)

   event_dict_RETRO = {'fixation start': 2, 'Cue': 3,
            'target': 4, 'correctresponse/1': 101,
            'correctresponse/2': 102, 'correctresponse/3': 103,
            'correctresponse/4': 104, 'incorrectresponse': 105,
            'RETRO_LVF/L': 111, 'RETRO_LVF/R': 112,
            'RETRO_RVF/L': 121, 'RETRO_RVF/R': 122}


   events_RETRO, event_dict_retro = mne.events_from_annotations(RETRO_raw)

   events_RETRO, notNeeded = mne.events_from_annotations(RETRO_raw)
   print(events_RETRO)

#Create epochs
   epochs_RETRO = mne.Epochs(RETRO_raw, events_RETRO, event_id=event_dict_RETRO,
                        tmin=-0.5, tmax=2.0)
   epochs_RETRO.drop_bad(reject=reject_criteria, flat=flat_criteria)
   epochs_RETRO.apply_baseline(interval)

# The TFR analysis is performed twice, one for the a1 and one for the a2 band.
# We start with the manipulations in the a1 band. Using a Morlet wavelet
   L_tfr_a1 = tfr_morlet(epochs_RETRO['RETRO_LVF'], freqs=frequenciesa1, picks=picks,
                              return_itc=False, n_cycles=wave_cycles, average = True,
                              output= 'power')
   R_tfr_a1= tfr_morlet(epochs_RETRO['RETRO_RVF'], freqs=frequenciesa1, picks=picks,
                                return_itc=False, n_cycles=wave_cycles, average = True,
                             output= 'power')


# Save the data from the TFR into separate arrays
   a1_PO7 = L_tfr_a1.data[0,:,:]
   a1_PO8 = L_tfr_a1.data[1,:,:]
   a1_PO72 = R_tfr_a1.data[0,:,:]
   a1_PO82 = R_tfr_a1.data[1,:,:]


# Take the average from the 4 frequencies for the band per time point (2501 points) and
# compute the means.
# Creating one mean for the entire frequency band
   m_a1_PO7 = np.mean(a1_PO7, axis=0, keepdims=True)
   m_a1_PO8 = np.mean(a1_PO8, axis=0, keepdims=True)
   m_a1_PO72 = np.mean(a1_PO72, axis=0, keepdims=True)
   m_a1_PO82 = np.mean(a1_PO82, axis=0, keepdims=True)


#save the mean values to arrays
   a1_RETRO_L_PO7 = np.append(a1_RETRO_L_PO7, m_a1_PO7, axis=0)
   a1_RETRO_L_PO8 = np.append(a1_RETRO_L_PO8, m_a1_PO8, axis=0)
   a1_RETRO_R_PO7 = np.append(a1_RETRO_R_PO7, m_a1_PO72, axis=0)
   a1_RETRO_R_PO8 = np.append(a1_RETRO_R_PO8, m_a1_PO82, axis=0)

# Manipulations the a2 band. Using a morlet wavelet
   L_tfr_a2 = tfr_morlet(epochs_RETRO['RETRO_LVF'], freqs=frequenciesa2, picks=picks,
                                 return_itc=False, n_cycles=wave_cycles, average = True,
                                 output= 'power')

   R_tfr_a2= tfr_morlet(epochs_RETRO['RETRO_RVF'], freqs=frequenciesa2, picks=picks,
                                return_itc=False, n_cycles=wave_cycles, average = True,
                                output= 'power')

# Save the data from the TFR into separate arrays
   a2_PO7 = L_tfr_a2.data[0,:,:]
   a2_PO8 = L_tfr_a2.data[1,:,:]
   a2_PO72 = R_tfr_a2.data[0,:,:]
   a2_PO82 = R_tfr_a2.data[1,:,:]


# Take the average from the 4 frequencies for the band per time point (2501 points) and com
putes the mean.
# Creating one mean for the entire frequency band
```

```
   m_a2_PO7 = np.mean(a2_PO7, axis=0, keepdims=True)
   m_a2_PO8 = np.mean(a2_PO8, axis=0, keepdims=True)
   m_a2_PO72 = np.mean(a2_PO72, axis=0, keepdims=True)
   m_a2_PO82 = np.mean(a2_PO82, axis=0, keepdims=True)


# Save the mean values to arrays
   a2_RETRO_L_PO7 = np.append(a2_RETRO_L_PO7, m_a2_PO7, axis=0)
   a2_RETRO_L_PO8 = np.append(a2_RETRO_L_PO8, m_a2_PO8, axis=0)
   a2_RETRO_R_PO7 = np.append(a2_RETRO_R_PO7, m_a2_PO72, axis=0)
   a2_RETRO_R_PO8 = np.append(a2_RETRO_R_PO8, m_a2_PO82, axis=0)



# Now we start with creating the data for the pandas dataframe
# This will be done for both the corresponding and non-corresponding trials
# First the correct answers are all merged
# Then the corresponding trials (visual fieldxhandedness) are merged
# Lastly the non-corresponding trials are merged

   events_RETRO = mne.merge_events(events_RETRO, [101,102,103,104], 80,
                          replace_events=False)
    events_RETRO = mne.merge_events(events_RETRO, [111,122], 113, replace_events=False)
    events_RETRO = mne.merge_events(events_RETRO, [112,121], 114, replace_events=False)


# These events set up the reaction times for the corresponding correct responses
# Setting up the RTs. Which starts with the define-target function.
# This allows us to define new targets based on co-occurring events and the time (in ms) bet
ween
# the target and the reference id events
   c1_events, nan_rt1 = define_target_events(events = events_RETRO, reference_id = 113,
                              target_id = 80, sfreq = sfreq, tmin = tmin, tmax=tmax, new_id = 25,
                              fill_na=fill_na)

#Remove the nan values
   rt1 = nan_rt1[np.logical_not(np.isnan(nan_rt1))]

# Calculate the percentage of correct answers (pc)
   pc = len(rt1)/number_of_trials*100

# Append the RT, corresponding side, participantnumber, condition,and pc
   corr_list.append(np.nanmean(rt1, axis=0))
   corresponding.append('corr')
   subjectnumber.append(subject)
   condition.append(current_condition)
   pc_list.append(pc)
```

```
# These events set up the reaction times for the non-corresponding correct responses
# Setting up the RTs. Which starts with the define-target function.
# This allows us to define new targets based on co-occurring events and the time (in ms)
# between the target and the reference id events
   c2_events, nan_rt2 = define_target_events(events = events_RETRO, reference_id = 114,
                              target_id = 80, sfreq = sfreq, tmin = tmin, tmax=tmax, new_id = 15,
                              fill_na=fill_na)

# Remove nan-values
   rt2 = nan_rt2[np.logical_not(np.isnan(nan_rt2))]

# Calculate the percentage of correct answers (pc)
   pc = len(rt2)/number_of_trials*100

# Append the RT, non-corresponding side, participantnumber, condition,and pc
   corr_list.append(np.nanmean(rt2, axis=0))
   corresponding.append('noncorr')
   subjectnumber.append(subject)
   condition.append(current_condition)
   pc_list.append(pc)

   subject = subject+1
```

**LPS CONSTRUCTION**

```
# construct the LPS for both conditions

#pre-cue a1
PRO_left_cues_a1 = Left_cues(a1_PRO_L_PO7,a1_PRO_L_PO8)
PRO_right_cues_a1 = Right_cues(a1_PRO_R_PO8, a1_PRO_R_PO7)
PRO_epoch_array_a1 = N_LPS(a1_PRO_L_PO7,a1_PRO_L_PO8,a1_PRO_R_PO8,
                            a1_PRO_R_PO7)

#pre-cue a2
PRO_left_cues_a2 = Left_cues(a2_PRO_L_PO7,a2_PRO_L_PO8)
PRO_right_cues_a2 = Right_cues(a2_PRO_R_PO8, a2_PRO_R_PO7)
PRO_epoch_array_a2 = N_LPS(a2_PRO_L_PO7,a2_PRO_L_PO8,a2_PRO_R_PO8,
                           a2_PRO_R_PO7)

#post-cue a1
RETRO_left_cues_a1 = Left_cues(a1_RETRO_L_PO7,a1_RETRO_L_PO8)
RETRO_right_cues_a1 = Right_cues(a1_RETRO_R_PO8, a1_RETRO_R_PO7,)
RETRO_epoch_array_a1 = N_LPS(a1_RETRO_L_PO7,a1_RETRO_L_PO8,a1_RETRO_R_PO8,
                              a1_RETRO_R_PO7)
```

```
#post-cue a2
RETRO_left_cues_a2 = Left_cues(a2_RETRO_L_PO7,a2_RETRO_L_PO8)
RETRO_right_cues_a2 = Right_cues(a2_RETRO_R_PO8, a2_RETRO_R_PO7)
RETRO_epoch_array_a2 = N_LPS(a2_RETRO_L_PO7,a2_RETRO_L_PO8,a2_RETRO_R_PO8,
                             a2_RETRO_R_PO7)
```

**LPS POWER PLOTS**

**pre-cue plots**

```
#creating the frame to plot both alpha bands in one plot
#Because the values are mapped from the array they are plotted from [0,0]
#we create some empty layers to allow the plot to be scaled more accurately

PRO_epoch_array_a1_np = np.mean(PRO_epoch_array_a1, axis=0, keepdims=True)
PRO_epoch_array_a2_np = np.mean(PRO_epoch_array_a2, axis=0, keepdims=True)
PRO_epoch_array = np.vstack([PRO_epoch_array_a1_np, PRO_epoch_array_a2_np])

empty = np.zeros(shape = (2,2501), dtype = float)
PRO_epoch_array2 = np.vstack ([empty,PRO_epoch_array ])
PRO_epoch_array3 = np.vstack ([PRO_epoch_array2, empty ])

times =epochs_PRO.times
plot_max = np.max(np.max(PRO_epoch_array3))
plot_min = -plot_max

fig, ax = plt.subplots(1)
im = plt.imshow(PRO_epoch_array3,
       extent=[times[0], times[-1], frequenciesa[0], frequenciesa[-1]],
       aspect='auto', origin='lower', cmap='coolwarm', vmin=plot_min, vmax=plot_max)

plt.xlabel('Time (sec)')
plt.ylabel('Frequency (Hz)')
plt.title('LPS Pre-cue')
cb = fig.colorbar(im)
cb.set_label('Power')
plt.axvline(x=0, color = 'k', linewidth = 1)
plt.show()
```

**post-cue plot**

```
#create the frame to plot both alpha bands in one plot
#Because the values are mapped from the array they are plotted from [0,0]
#we create some empty layers to allow the plot to be scaled more accurately


RETRO_epoch_array_a1_np = np.mean(RETRO_epoch_array_a1, axis=0, keepdims=True)
RETRO_epoch_array_a2_np = np.mean(RETRO_epoch_array_a2, axis=0, keepdims=True)
```

```
RETRO_epoch_array = np.vstack([RETRO_epoch_array_a1_np, RETRO_epoch_array_a2_np])


# create an empty list for scaling
empty = np.zeros(shape = (2,2501), dtype = float)
RETRO_epoch_array2 = np.vstack ([empty,RETRO_epoch_array ])
RETRO_epoch_array3 = np.vstack ([RETRO_epoch_array2, empty ])

times =epochs_RETRO.times
plot_max = np.max(np.max(RETRO_epoch_array3))
plot_min = -plot_max

fig, ax = plt.subplots(1)
im = plt.imshow(RETRO_epoch_array3,
        extent=[times[0], times[-1], frequenciesa[0], frequenciesa[-1]],
        aspect='auto', origin='lower', cmap='coolwarm', vmin=plot_min, vmax=plot_max)

plt.xlabel('Time (sec)')
plt.ylabel('Frequency (Hz)')
plt.title('LPS post-cue')
cb = fig.colorbar(im)
cb.set_label('Power')

plt.axvline(x=0, color = 'k', linewidth = 1)
plt.show()
```

**STATISTICAL ANALYSIS**

```
# The statistical analysis consists of two main parts
# First we perform the LPS analysis to look for relevant time windows
# Then we perform a 2x2 repeated measures ANOVA
```

**LPS analysis**

**pre-cue a1**

```
# Creating 40 ms time intervals for the analysis
# Performing one sample one-tailed t-tests to get an overview of the raw sig time windows
chunk_size = 40
sig_time_windows = []
for i in range(0, 2501, chunk_size):
    chunk = PRO_epoch_array_a1[:, i:i+chunk_size]
    chunk = chunk.flatten()
    tstat, pval = stats.ttest_1samp(chunk, popmean=0, alternative = 'greater')
    if pval1 <= 0.05:
        print('time window', + i,'till', + i+chunk_size, 'ms ', 'pval=',+pval)
    sig_time_windows.append(pval)
```

```
#calculating adjusted p-values
chunk_size = 0
reject, pvals, alphacs, alphacb = smm.multipletests(list3, alpha=0.05, method='b',
                                                     is_sorted=False, returnsorted=False)


for i in pvals:
    if i <= 0.05:
        print('time window', + chunk_size,'till', + chunk_size+40, 'ms ', 'pval=',+i )
    chunk_size = chunk_size+40

#calculate p-value of the sig time window
tstat, pval = stats.ttest_1samp(PRO_epoch_array_a1[:, 640:1480].flatten(), popmean=0,
                                alternative = 'greater')
print('time window 140-980 ms', 'pval=',+pval, 'tstat=',+tstat )

#calculate the polarity of the sig time windows
pol_1 = np.mean(PRO_epoch_array_a1[:, 640:1480].flatten())
print('Polarity time window 140-980 ms = '+ str(pol_1))
```

**pre-cue a2**

```
# Create 40 ms time intervals for the analysis
# Perform one sample one-tailed t-tests to get an overview of the raw sig time windows
chunk_size = 40
sig_time_windows = []
for i in range(0, 2501, chunk_size):
    chunk = PRO_epoch_array_a2[:, i:i+chunk_size]
    chunk = chunk.flatten()
    tstat, pval = stats.ttest_1samp(chunk, popmean=0, alternative = 'greater')
    if pval1 <= 0.05:
        print('time window', + i,'till', + i+chunk_size, 'ms ', 'pval=',+pval)
    sig_time_windows.append(pval)

#calculating adjusted p-values
chunk_size = 0
reject, pvals, alphacs, alphacb = smm.multipletests(list3, alpha=0.05, method='b',
                                                     is_sorted=False, returnsorted=False)
for i in pvals:
    if i <= 0.05:
        print('time window', + chunk_size,'till', + chunk_size+40, 'ms ', 'pval=',+i )
    chunk_size = chunk_size+40

#calculate p-value of the sig time window
tstat, pval = stats.ttest_1samp(PRO_epoch_array_a2[:, 640:1320].flatten(), popmean=0,
                                alternative = 'greater')
print('time window 140-820 ms', 'pval=',+pval, 'tstat=',+tstat )

#calculate the polarity of the sig time windows
pol_1 = np.mean(PRO_epoch_array_a2[:, 640:1320].flatten())
print('Polarity time window 140-820 ms = '+ str(pol_1))
```

**post-cue a1**

```
# Creating 40 ms time intervals for the analysis
# Perform one sample one-tailed t-tests to get an overview of the raw sig time windows
chunk_size = 40
sig_time_windows = []
for i in range(0, 2501, chunk_size):
    chunk = RETRO_epoch_array_a1[:, i:i+chunk_size]
    chunk = chunk.flatten()
    tstat, pval = stats.ttest_1samp(chunk, popmean=0, alternative = 'greater')
    if pval1 <= 0.05:
        print('time window', + i,'till', + i+chunk_size, 'ms ', 'pval=',+pval)
    sig_time_windows.append (pval)

#calculate adjusted p-values
chunk_size = 0
reject, pvals, alphacs, alphacb = smm.multipletests(list3, alpha=0.05, method='b',
                                            is_sorted=False,  returnsorted=False)
for i in pvals:
    if i <= 0.05:
        print('time window', + chunk_size,'till', + chunk_size+40, 'ms ', 'pval=',+i )
    chunk_size = chunk_size+40

#calculate p-value of the sig time window
tstat, pval = stats.ttest_1samp(RETRO_epoch_array_a1[:, 600:1480].flatten(), popmean=0,
                        alternative = 'greater')
print('time window 100-980 ms', 'pval=',+pval, 'tstat=',+tstat )

#calculate the polarity of the sig time windows
pol_1 = np.mean(RETRO_epoch_array_a2[:, 840:1480].flatten())
print('Polarity time window 100-980 ms = '+ str(pol_1))
```

**post-cue a2**

```
#variables
chunk_size = 40
sig_time_windows = []

# Create 40 ms time intervals for the analysis
# Perform one sample one-tailed t-tests to get an overview of the raw sig time windows
# Save sig time windows in one list
for i in range(0, 2501, chunk_size):
    chunk = RETRO_epoch_array_a2[:, i:i+chunk_size]
    chunk = chunk.flatten()
    tstat, pval = stats.ttest_1samp(chunk, popmean=0, alternative = 'greater')
    if pval1 <= 0.05:
        print('time window', + i,'till', + i+chunk_size, 'ms ', 'pval=',+pval)
    sig_time_windows.append(pval)
```

```
#calculating adjusted p-values
chunk_size = 0
reject, pvals, alphacs, alphacb  = smm.multipletests(list3, alpha=0.05, method='b',
                                              is_sorted=False,  returnsorted=False)


for i in pvals:
   if i <= 0.05:
      print('time window', + chunk_size,'till', + chunk_size+40, 'ms ', 'pval=',+i )
   chunk_size = chunk_size+40

#calculate p-value of the sig time window
tstat, pval = stats.ttest_1samp(RETRO_epoch_array_a2[:, 640:760].flatten(), popmean=0,
                        alternative = 'greater')
print('time window 140-260 ms', 'pval=',+pval, 'tstat=',+tstat )

#calculate p-value of the sig time window
tstat, pval = stats.ttest_1samp(RETRO_epoch_array_a2[:, 840:1480].flatten(), popmean=0,
                        alternative = 'greater')
print('time window 340-980 ms', 'pval=',+pval, 'tstat=',+tstat )

#calculate the polarity of the sig time windows
pol_1 = np.mean(RETRO_epoch_array_a2[:, 640:760].flatten())
print('Polarity time window 140-760 ms = '+ str(pol_1))

pol_2 = np.mean(RETRO_epoch_array_a2[:, 840:1480].flatten())
print('Polarity time window 340-980 ms = '+ str(pol_2))
```

**ANOVA**

```
#Create the pandas data frame
df = pd.DataFrame({'id': subjectnumber,
             'RT': corr_list,
             'condition': condition,
             'corr_side': corresponding,
             'pc':pc_list})
df

#test the sphericity of the data
pg.sphericity(df, dv='pc', subject='id', within=['condition','corr_side'])
pg.sphericity(df, dv='RT', subject='id', within=['condition','corr_side'])

#calculate the means of the RT and PC
df.groupby(['condition', 'corr_side'])['RT'].mean()
df.groupby(['condition', 'corr_side'])['pc'].mean()

# Repeated measures ANOVA 2x2 factors for RT
pg.rm_anova(dv='RT', within=['condition', 'corr_side'], subject = 'id', data =df, detailed = True
                 )
```

```
# Repeated measures ANOVA 2x2 factors for pc
pg.rm_anova(dv='pc', within=['condition', 'corr_side'], subject = 'id', data =df, detailed = True
)

# PC pre-cue condition
# Filter the dataframe for the conditionxpc t-tests for correspondence
correspondence = df[df["condition"] =='pre-cue']
correspondence = correspondence['RT']

# One sample t test for simon effect(correspondence)
tstat, pval = stats.ttest_1samp(correspondence, popmean=0, alternative = 'greater')
p_correspondence.append(pval)

# PC pre-cue condition
# Filter the dataframe for the conditionxpc t-tests for correspondence
correspondence = df[df["condition"] =='post-cue']
correspondence = correspondence['RT']

# One sample t-test for Simon effect(correspondence)
tstat, pval = stats.ttest_1samp(correspondence, popmean=0, alternative = 'greater')
p_correspondence.append(pval)

# PC pre-cue condition
# Filter the data frame for the conditionxpc t-tests for correspondence
correspondence = df[df["condition"] =='pre-cue']
correspondence = correspondence['pc']

# One sample t-test for Simon effect(correspondence)
tstat, pval = stats.ttest_1samp(correspondence, popmean=0, alternative = 'greater')
p_correspondence.append(pval)

# PC post-cue condition
# Filter the data frame for the conditionxpc t-tests for correspondence
correspondence = df[df["condition"] =='post-cue']
correspondence = correspondence['pc']

# One sample t-test for Simon effect(correspondence)
tstat, pval = stats.ttest_1samp(correspondence, popmean=0, alternative = 'greater')
p_correspondence.append(pval)

# Correct the pvalues
reject, pvals, alphacs, alphacb  = smm.multipletests(p_correspondence, alpha=0.05,
                                   method='b', is_sorted=False, returnsorted=False)
```