Novos métodos para rastreamento de olhar baseados na razão cruzada com compensação de movimentos da cabeça

Flávio Luiz Coutinho

Tese apresentada
ao
Instituto de Matemática e Estatística
da
Universidade de São Paulo
para
obtenção do título
de
Doutor em Ciências

Programa: Ciência da Computação

Orientador: Prof. Dr. Carlos Hitoshi Morimoto

Durante o desenvolvimento deste trabalho o autor recebeu auxílio financeiro da CAPES

São Paulo, dezembro de 2011

Novos métodos para rastreamento de olhar baseados na razão cruzada com compensação de movimentos da cabeça

Esta tese contém as correções e alterações
sugeridas pela Comissão Julgadora durante a defesa
realizada por Flávio Luiz Coutinho em 14/10/2011.
O original encontra-se disponível no Instituto de
Matemática e Estatística da Universidade de São Paulo.

Comissão Julgadora:

- Prof. Dr. Carlos Hitoshi Morimoto (orientador) - IME-USP
- Prof. Dr. Roberto Hirata Junior - IME-USP
- Prof. Dr. Romero Tori - EP-USP
- Prof. Dr. Ronald Dennis Paul Kenneth Clive Ranvaud - ICB-USP
- Prof. Dr. Dan Witzner Hansen - IT-UC

Agradecimentos

A Deus, por todos os dias me iluminar.

Ao meu orientador, Carlos Hitoshi Morimoto, pela oportunidade de realizar este trabalho e também pela paciência, em especial nos momentos de dificuldade.

Aos amigos e colegas de IME, pelas conversas, troca de ideias e momentos de descontração.

A todos os voluntários que participaram dos experimentos apresentados neste trabalho.

Por fim, agradeço a meus pais, Liane e Walter, aos meus irmãos Audrey e Walter, à minha noiva Monete, e a toda minha família pelo apoio e incentivo durante a realização deste trabalho.

Resumo

O método para rastreamento de olhar baseado na razão cruzada – em inglês *cross-ratio* (**CR**) – mostrou-se bastante promissor ao empregar uma configuração de equipamento simples (similar à usada pelo método tradicional que faz o rastreamento do vetor brilho-pupila) sem possuir, contudo, as desvantagens da baixa tolerância a movimentos de cabeça e da necessidade frequente de recalibração (duas características indesejáveis quando se pensa no uso de rastreadores de olhar em aplicações interativas).

A modelagem geométrica do problema de estimação do olhar adotada pelo método da razão cruzada não exige o conhecimento prévio de qualquer tipo parâmetro e não impõe qualquer restrição ao posicionamento do olho. Isso torna este método, portanto, livre de calibração e tolerante a movimentos de cabeça.

Observou-se na prática, contudo, que os resultados de estimação do olhar para este método apresentam baixa precisão, o que é explicado por duas hipóteses simplificadoras consideradas na modelagem do problema e que são, de fato, as principais fontes de erro do método. Algumas extensões deste método foram desenvolvidas com o objetivo de compensar estas fontes de erro através de parâmetros de correção obtidos por calibração. Entretanto, os parâmetros estimados ainda mostram certo grau de dependência em relação à posição do usuário na qual é feita a calibração, o que acaba restringindo movimentos de cabeça.

Mesmo com essa restrição, dado o potencial do método da razão cruzada em ser naturalmente mais tolerante a movimentos de cabeça (em comparação a métodos mais tradicionais como o que faz o rastreamento do vetor brilho-pupila), estudamos mais profundamente este método a fim de investigar possíveis extensões que melhor tolerassem movimentos de cabeça. O fruto deste estudo são dois novos métodos baseados na razão cruzada que atingiram este objetivo: o método *cross-ratio with dynamic displacement vector correction* (**CR-DD**) e o método *planarization of CR features* (**PL-CR**). Enquanto o método **CR-DD** é melhor adaptado para os casos em que a movimentação do usuário é perpendicular ao plano da tela, o método **PL-CR** compensa todos os tipos de movimentos de cabeça.

Avaliações dos métodos **CR-DD** e **PL-CR** também são apresentadas neste trabalho através tanto de simulações quanto experimentos com um grupo de usuários, confirmando uma significativa melhora de performance destes métodos quando comparados a outros métodos que não modelam a movimentação da cabeça de forma explicita.

Palavras-chave: rastreamento de olhar, livre movimentação da cabeça, razão cruzada, homografia.

Abstract

The cross-ratio (**CR**) method for remote gaze tracking showed great potential by employing a simple hardware setup (similar to the one used by the traditional pupil-corneal reflection method) without having the disadvantages of low tolerance to head movements and frequent need for system recalibration (two unwanted features when we think about using eye trackers for interactive applications).

The geometrical modeling of the gaze estimation problem considered by the cross-ratio method does not require previous knowledge of any kind of parameter and also does not restrict the position of the eye. This way, this method is both calibration free and robust to head movements.

It was observed, however, that estimation results for this method have low accuracy. This can be explained by two simplifying assumptions assumed in the modeling of the problem which are, in fact, the method's main sources of error. Some extensions to this method were developed with the goal of compensating these sources of error by the use of correction parameters obtained via calibration procedures. However, the parameters estimated by these procedures are still bound, to some extent, to the user position where calibration is performed, which limits the freedom of head movements.

Besides this restriction, given the potential of the cross-ratio method of being more tolerant to head movements (compared to the traditional pupil-corneal reflection method), we studied it in greater detail in order to investigate possible extensions that better tolerates head movements. As a result of this study two new methods based on the cross-ratio were proposed: the *cross-ratio with dynamic displacement vector correction* (**CR-DD**) method and the *planarization of CR features* (**PL-CR**) method. While the **CR-DD** method is adapted to the particular case of depth movements (when the user moves in the direction perpendicular to the screen), the **PL-CR** method compensates all types of head movements.

Evaluation of the **CR-DD** and **PL-CR** methods are presented with simulation and user experiments, confirming a significant performance improvement when compared to the methods that do not explicitly model head motion.

Keywords: eye tracking, gaze tracking, free head motion, cross ratio, homography.

Sumário

Capítulo 1

Introdução

Assim como outras partes do corpo, o olho, mesmo que de uma forma não explícita, pode transmitir e comunicar uma série de informações sobre uma pessoa. Através do seu olhar podemos inferir seu estado emocional e também saber para onde está voltada sua atenção. Evidentemente, é possível que uma pessoa esteja olhando para um determinado lugar (ou um objeto/pessoa), mas sua atenção esteja voltada para algo que não tenha relação alguma com aquilo sendo observado, como um pensamento. Contudo, em muitas situações, aquilo que é observado está diretamente relacionado com o foco de nossa atenção, principalmente quando estamos executando uma tarefa que depende do sentido da visão para ser realizada [JC80].

Embora nosso campo de visão seja bastante amplo, a percepção dos detalhes é restrita a uma pequena região da retina chamada foveola onde há uma maior concentração de células foto receptoras e que cobre apenas 1.3^o do campo visual [Duc03]. Assim, para que seja possível perceber detalhes de um objeto em particular, é preciso que o globo ocular seja orientado de modo que o objeto de interesse seja projetado exatamente sobre a foveola. Como consequência desta característica anatômica do olho humano, temos que a orientação do globo ocular está, portanto, diretamente relacionada a quais elementos do ambiente uma pessoa está dirigindo sua atenção. Em outras palavras, a orientação do olho é o que define a linha da visão de uma pessoa.

Nós, seres humanos, temos uma habilidade inata de inferir a linha da visão de uma pessoa pela observação dos seu olhos. De uma forma instintiva percebemos se uma pessoa está ou não

olhando diretamente para nós durante uma conversa, e prontamente percebemos se nosso interlocutor desvia o olhar para algo que lhe chame a atenção, sendo ainda capazes que estimar qual é esse novo ponto de interesse. O olho transmite, desta forma, uma informação similar àquela que é passada através de um gesto feito por alguém que aponta o braço para uma determinada direção ou alvo.

1.1 Rastreadores de olhar

Dispositivos que possuem esta mesma capacidade de estimar a linha da visão de uma pessoa são chamados rastreadores de olhar. Da mesma forma que nossa capacidade de interpretar o olhar de uma pessoa tem sua função ao complementar a comunicação de uma forma não verbal, sistemas que empregam o uso de rastreadores de olhar tem a sua disposição um canal extra de comunicação com o usuário que pode ser utilizado de diferentes formas. Em um levantamento de aplicações beneficiadas pelo uso de rastreadores de olhar, Duchowski as classificou em dois grupos principais: aplicações de diagnóstico e aplicações interativas [Duc02]. As aplicações de diagnóstico utilizam os dados fornecidos por um rastreador de olhar como evidência do comportamento visual e da atenção de um indivíduo. Já as aplicações interativas utilizam esses dados para responder ou interagir com o usuário de acordo com o movimento observado do olhar.

Entre as as aplicações de diagnóstico podemos citar: estudos neurológicos, que relacionam dados provenientes do rastreamento do olhar com a atividade cerebral; estudos psicológicos, que analisam os padrões de movimento do olhar durante a execução de determinadas tarefas, como a leitura, por exemplo; aplicações em marketing, como a avaliação da eficácia de anúncios; e testes de usabilidade de softwares e páginas de internet. Os pontos em que uma pessoa fixa o olhar, a duração das fixações, a quantidade delas e a distribuição espacial constituem algumas das informações importantes relacionadas ao movimento dos olhos que um rastreador de olhar permite coletar, daí sua utilidade e importância para as aplicações de diagnóstico.

Quando se pensa em aplicações interativas, a primeira ideia que costuma vir à mente é a utilização de um rastreador de olhar como um dispositivo apontador alternativo. Neste

caso, o olhar pode ser utilizado para controlar um cursor ou selecionar elementos de uma interface gráfica. Tal uso é de grande importância para pessoas que possuem deficiências físicas ou dificuldades motoras que as impedem de usar os dispositivos típicos para estes fins como mouses, joysticks e teclados. Ao fornecer a este público acesso aos computadores e suas aplicações, os rastreadores de olhar ganham destaque como um elemento capaz de melhorar a qualidade de vida destas pessoas. Dentro deste cenário podemos destacar o COGAIN (**co**mmunication by **ga**ze **in**teraction - http://www.cogain.org/), uma rede de colaboração que tem por objetivo pesquisar, desenvolver e disseminar tecnologias que possibilitem a comunicação através da interação com o olhar.

Além do uso de rastreadores como dispositivos apontadores alternativos, eles também podem ser empregados de uma forma não tão direta para o desenvolvimento de interfaces mais naturais e intuitivas, visando o público geral. O simples reconhecimento de gestos do olho como piscadas, fixações e movimentação (sem estimar o ponto observado) constitui uma valiosa informação que pode ser utilizada por uma interface prestativa [Sel04]. Até mesmo o simples reconhecimento da presença de olhos pode ser utilizado para promover a interação de usuários com aplicativos [SVS03]. O trabalho desenvolvido por Zhai *et al.* [ZMI99] e estendido por Barcelos [Bar05] são exemplos onde um rastreador de olhar é aplicado de forma indireta. Nestes trabalhos, um rastreador de olhar é utilizado em conjunto com um mouse convencional para controlar o cursor de uma interface gráfica. Enquanto o rastreador de olhar é usado para mover o cursor entre regiões distantes da tela, acompanhando o olhar do usuário, o mouse é utilizado para manipulações precisas do cursor dentro de uma região.

1.2 Técnicas para rastreamento do olhar

Diversos métodos e tecnologias foram criadas para resolver o problema do rastreamento de olhar. Algumas técnicas são bastante intrusivas como aquelas que exigem o uso de lentes de contato especiais [Rob63] ou eletrodos posicionados ao redor dos olhos [AKS93]. Embora aceitáveis para o uso em laboratórios ou ambientes controlados, tais técnicas não são interessantes para aplicações interativas pois exigem preparo antes do uso e o uso continuado pode ser bastante desconfortável.

O avanço do poder computacional permitiu, já há alguns anos, o surgimento de uma nova categoria de rastreadores que utilizam câmeras de vídeo, viabilizando o aparecimento dos rastreadores de olhar remotos. Estes dispositivos não empregam nenhum tipo de equipamento em contato físico com o usuário, o que os torna especialmente atraentes para uso em aplicações interativas. De uma forma geral, os rastreadores de olhar remotos capturam e processam imagens de um ou ambos os olhos do usuário. Durante o processamento das imagens, características relevantes do olho são detectadas e utilizadas para estimar o ponto observado sobre um plano de interesse (tipicamente a tela do monitor), ou ainda a direção do olhar no espaço 3D. Características típicas do olho que são usadas por técnicas de rastreamento de olhar baseadas em câmeras incluem os contornos da pupila e da íris, os cantos dos olhos e ainda reflexos sobre a córnea gerados por fontes de luz (iluminação ativa) [VCP+08].

Apesar de os rastreadores de olhar remotos serem mais apropriados para aplicações interativas, as técnicas tradicionais apresentam duas limitações principais que constituem importantes obstáculos para uma adoção em larga escala destes dispositivos: a necessidade de calibração do dispositivo a cada sessão de uso e a pouca tolerância a movimentos de cabeça. Técnicas mais recentes, que empregam modelos geométricos mais avançados, foram bem sucedidas em contornar essas limitações, substituindo a calibração a cada sessão de uso por uma única calibração por usuário, e permitindo uma maior movimentação dos mesmos. Por outro lado, os rastreadores que empregam tais técnicas possuem configurações mais complexas, exigindo calibração da(s) câmera(s) e da geometria do ambiente. Essas configurações mais complexas acabam pesando contra a usabilidade, sem contar o maior custo, o que não é interessante quando se tem em mente aplicações interativas.

1.3 Rastreamento de olhar baseado na razão cruzada

Yoo *et al.* [YKLC02] propuseram uma técnica bastante elegante para rastreamento do olhar baseado na razão cruzada, uma invariante da geometria projetiva. A técnica proposta conseguiu resolver as limitações dos métodos tradicionais e ainda manter um sistema bastante simples composto por apenas uma câmera não calibrada. O princípio de funcionamento deste método consiste no uso de um padrão planar composto por 4 de fontes de luz, nor-

malmente posicionadas nos cantos de um monitor, que geram 4 reflexões sobre a córnea. A partir da posição relativa entre estes reflexos e o centro da pupila pode-se usar a razão cruzada para estimar o ponto observado por uma pessoa.

Essa técnica, embora bastante interessante, considera duas hipóteses simplificadoras em sua formulação, que não são válidas na prática, e que constituem as principais fontes de erro para este método [GE08]. A primeira destas hipóteses é que o padrão dos reflexos gerados sobre a córnea e o centro da pupila são coplanares. A segunda é que a linha da visão é definida pelo eixo óptico do olho (eixo que passa pelos centros de curvatura das estruturas do olho, como o centro da pupila e o centro de curvatura da córnea) quando, de fato, a linha da visão é definida pelo eixo visual, que apresenta um desvio angular entre 4 a 8 graus em relação ao eixo óptico [WS82].

Alguns trabalhos posteriores foram desenvolvidos para corrigir estas fontes de erro. Yoo e Chung [YC05] propuseram um método para compensar a primeira fonte de erro, através de uma escala dos pontos na imagem referente as reflexões especulares. Eles não consideraram, contudo, uma solução para corrigir o desvio angular entre os eixos do olho. Coutinho e Morimoto [CM06] estenderam o trabalho de Yoo e Chung para corrigir também a segunda fonte de erro através da adição de um vetor bidimensional ao ponto estimado no plano da tela. Kang *et al.* [KGME07] e Hansen *et al.* [HAV10] propuseram ainda uma forma de se corrigir tais fontes de erro pelo mapeamento dos pontos estimados (com erro) para os pontos corretos através de homografias.

Para que as correções propostas por cada um destes métodos possam ser aplicadas, torna-se necessário um processo de calibração onde parâmetros usados nas correções são estimados. Embora tais métodos apresentem melhoras significativas em relação ao método original [YKLC02], os parâmetros estimados pela calibração ainda mostram certo grau de dependência em relação à posição de calibração, o que implica em queda de desempenho para tais métodos quando há movimentos de cabeça para longe desta posição.

Neste trabalho investigamos os métodos baseados na razão cruzada em maiores detalhes com o objetivo de melhorar a tolerância deste método a movimentos de cabeça. Como resultado apresentamos dois novos métodos para rastreamento de olhar baseados na razão cruzada: o método *cross-ratio with dynamic displacement vector correction* (**CR-DD**— em

português método da *razão cruzada com correção por deslocamento dinâmico*) e o método *planarization of CR features* (**PL-CR**— em português *planificação das características do método da razão cruzada*). Ambos se mostraram capazes de lidar melhor com as situações em que o usuário se afasta da posição de calibração em comparação aos métodos atualmente presentes na literatura também baseados no conceito da razão cruzada. Enquanto o método **CR-DD** se mostra mais apropriado para o caso em que o usuário se movimenta na direção perpendicular a tela, o método **PL-CR** é uma solução mais completa, capaz de compensar todos os tipos de movimentos de cabeça.

1.4 Contribuições

Dentre as principais contribuições deste trabalho podemos destacar:

- Revisão bibliográfica dos métodos de rastreamento de olhar baseados na razão cruzada.
- Desenvolvimento e implementação do método **CR-DD**.
- Desenvolvimento e implementação do método **PL-CR**.
- Avaliação de desempenho dos métodos propostos **CR-DD** e **PL-CR** e comparação com outros métodos baseados na razão cruzada, através de simulações.
- Desenvolvimento de ferramentas para simulação de métodos de rastreamento de olhar.
- Realização de experimentos para um grupo de usuários com o objetivo de avaliar os métodos **CR-DD** e **PL-CR**, e também compará-los a outros métodos, com geração de um conjunto de dados que pode ser reaproveitado em novos experimentos.

1.5 Organização do texto

A fim de tornar o texto acessível para um maior número pessoas com interesse na área de rastreamento de olhar, a continuação deste texto é apresentada em inglês sob a forma de um apêndice (apêndice A). O conteúdo deste apêndice é auto contido de modo que leitores que não compreendem português possam iniciar a leitura do texto diretamente pelo apêndice

sem qualquer prejuízo para a compreensão do texto, com exceção de uma introdução mais resumida apresentada na seção A.1.

Dando continuidade ao texto do apêndice, a seção A.2 apresenta uma breve revisão de técnicas remotas para o rastreamento do olhar. Inicialmente descrevemos um método tradicional que ilustra o princípio de funcionamento destes dispositivos, e permite entender o problema da tolerância a movimentos da cabeça. Em seguida apresentamos métodos mais recentes da literatura, que apresentam melhor tolerância a movimentos de cabeça, mas que, por outro, lado requerem configurações mais complexas que acabam afetando a usabilidade de tais dispositivos.

Na seção A.3 descrevemos em detalhes o método da razão cruzada proposto por Yoo *et al.* em [YKLC02], explicando seu princípio de funcionamento e porque ele se mostra mais tolerante a movimentos de cabeça. São discutidas as hipóteses simplificadoras assumidas na formulação do problema que são as principais fontes de erro do método. Alguns métodos mais recentes baseados na razão cruzada que compensam tais fontes de erro [YC05, CM06, KGME07, HAV10] são mostrados e os problemas ainda presentes nestes métodos são discutidos na seção A.4.

Em seguida, nas seções A.5 e A.6, apresentamos os dois novos métodos baseados na razão cruzada desenvolvidos neste trabalho, que tem por objetivo serem mais tolerantes a movimentos de cabeça. Primeiro apresentamos o método **CR-DD**e, em seguida, o método **PL-CR**, descrevendo em detalhes como cada método funciona e porque eles compensam os movimentos de cabeça de uma forma mais eficiente.

Uma avaliação dos dois métodos propostos é apresentada na seção A.7, onde são mostrados resultados tanto de simulações quanto de experimentos com usuários. Nesta seção são também detalhadas as configurações utilizadas na avaliação, bem como apresentamos o protocolo experimental utilizado para os experimentos com usuários.

Na seção A.8 discutimos detalhes práticos da implementação de um dispositivo rastreador de olhar capaz de operar em tempo real. Descrevemos as principais características do hardware utilizado e do software implementado.

Finalmente, a seção A.9 conclui este trabalho, fazendo um balanço do que foi desenvolvido, dos resultados alcançados e das contribuições. Na seção A.10 discutimos possíveis

melhorias e caminhos para trabalhos futuros.

Appendix A

New head movement compensation methods for cross-ratio based remote eye gaze tracking

This text corresponds to the english final version of the thesis defended by Flávio Luiz Coutinho in 14/10/2011 and approved by the defense comittee.

A.1 What is eye tracking?

In a simplified way, eye tracking consists of measuring eye movements. By tracking eye movements it is possible to either infer the line of sight or a specific target being gazed by a subject. This information can be useful to a series of applications including psychological and neurological studies, usability tests, and interactive applications [Duc03].

Several methods have been developed to track eye movements as described in [YS75, MM05, VCP+08, HJ10]. Since we are primarily interested in the use of eye trackers for interactive applications our focus is on devices that are non intrusive and remote. This way, devices that use special contact lenses [Rob63] or electrodes attached around the eyes [AKS93] are not interesting, since they require preparation before use and use for long periods of time can be uncomfortable. Camera based devices overcame these limitations, specially those that use remote configurations, i.e., where the user does not need to wear or be in contact with any kind of equipment.

In general, camera based devices captures and process images of a person's eye. During image processing, relevant eye features are detected and tracked and used to compute the point of regard. Typical eye features used are the iris and pupil borders, eye corners, and corneal reflections generated by light sources (active illumination) [VCP+08].

When we think about interactive applications, it is also desirable to have devices that allow free head movement, which improve usability and comfort, and low are cost, as a way to spread these kind of devices among the public.

A.2 Remote eye gaze tracking

Remote eye gaze tracking methods can be classified into two groups [HJ10]: interpolation based methods and model based. Interpolation based methods map image features to gaze points. Model based methods estimate the 3D gaze direction and intersection between scene geometry and gaze direction is computed as the point of regard (PoR). System requirements of interpolation based methods tend to be smaller than model based methods but head movement is restricted. Model based methods, on the other hand, offers greater freedom of movement though they require more complex system setup. Next in this section a traditional interpolation based remote eye gaze tracking that illustrates the working principle of gaze tracking is presented, followed by a discussion about model based methods.

A.2.1 Interpolation based methods

The *pupil corneal reflection* method (**PCR**) is an example of an interpolation based gaze tracking technique. The **PCR** method detects and tracks the pupil and a corneal reflection, generated by a light source. Infrared light sources are often used as they do not distract users, offer a more homogeneous lighting condition and improve the robustness to ambient light changes in indoor environments.

Figure A.1 illustrates the geometric setup considered by the **PCR** method. Considering the cornea surface as a sphere centered at C, the corneal reflection G and its image g do not move when the eye rotates around C. Thus, g can be used as a reference point. As the eye rotates, the pupil center position P moves in space, and G and P define an image vector $\vec{gp}$ which is mapped to screen coordinates through a mapping function. The mapping function is obtained by a calibration procedure in which the user is asked to gaze at specific screen targets. The work by Morimoto *et al.* [MKAF99] uses a second order polynomial as a mapping function since a linear mapping may not be adequate for large eye rotations [CVC08].

Since the observed $\vec{gp}$ vector is a function of the scene geometry (camera, eye and screen), different eye positions will define different vectors for a given point on the screen. This way, it is not expected that the mapping function, once optimized for the calibration position,

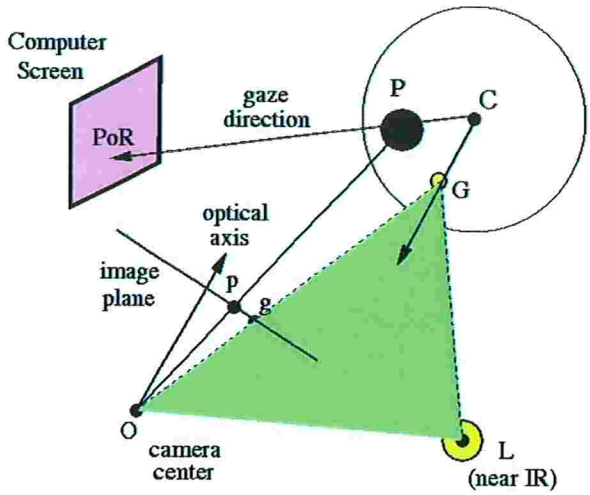

Figure A.1: *Geometric setup of the **PCR** method.*

will estimate gaze with the same accuracy for other different eye positions. This accuracy decay was evaluated by Mimica e Morimoto [MM05] and illustrates two limitations of the **PCR** method: low tolerance to head movements and the need of frequent recalibration.

A.2.2 Model based methods

Model based methods use geometric models of the eye to estimate the line of sight in 3D [SL03, GE06, HNL06, GE08, CTGJ08, NKIT08, NSI+10, ME10]. An eye model that is usually considered for model based methods is shown in Figure A.2. Important elements of this model for gaze tracking methods are: the eyeball, modeled as a sphere; the foveola, the central region of the fovea (the retinal region on the back of the eye that is responsible for the detailed vision) that comprehends about 1.3^o of visual angle [Duc03]; the pupil, a circular orifice defined by the iris, by which light enters into the eye; the cornea, a transparent membrane that covers the iris and can be approximated by a spherical surface; the optical axis of the eye, the line defined by the centers of the eyeball, cornea, and pupil; and the visual axis of the eye, the line connecting the foveola and the point of regard, and passing also through the cornea center (typically pointing in the nasal direction). Average values for this model are: cornea radius of 0.78 cm; distance from pupil to cornea center of 0.42 cm; horizontal and vertical angles between visual and optical axis of 5^o and 1.5^o (the combined angle between these two axis is usually referred to as the κ angle); and a combined index of refraction of 1.3375 for the cornea and aqueous humour [GE06].

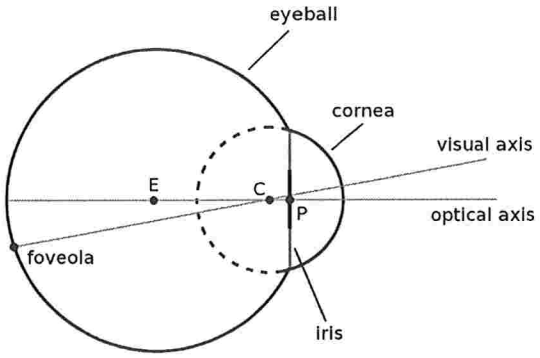

Figure A.2: *Geometric eye model and relevant elements for gaze tracking methods.*

All model based methods follow a common strategy: first the optical axis of the eye is reconstructed in 3D; second the visual axis (deviated from the optical axis by κ angle) is reconstructed; and finally PoR is estimated by intersecting the visual axis with scene geometry. Reconstruction of the optical axis of the eye is done by estimation of the cornea (C) and pupil (P) centers. Since the line of sight is defined by the visual axis and not the optical axis of the eye, the angular deviation between them must be known in order to reconstruct the visual axis from the optical axis.

Some of these methods use stereo cameras [SL03, GE08, CTGJ08, NKIT08], while a single camera is used by others [GE06, HNL06]. In both cases, the cameras need to be calibrated and scene geometry must be known so that the PoR can be computed. Thus, these systems need to be fully calibrated, a requirement that does not exist for the **PCR** technique. This way the freedom of movement is achieved by an increase in complexity of system setup.

Guestrin and Eizenman [GE06] showed that, for gaze estimation methods based on detection and tracking of the pupil and corneal reflections, eye model complexity varies with the number of cameras and light sources used by them. The minimum system configuration that allows for free head motion uses a single camera and two light sources. With such a setup, an eye model with 5 parameters must be used: cornea radius, distance from pupil to cornea center, combined index of refraction of the cornea and aqueous humour and vertical and horizontal rotation angles describing the deviation between the visual and optical axis of the eye. These personal parameters are estimated by a one time per subject calibration

procedure. When a setup that uses at least two cameras and at least two light sources are used, the optical axis of the eye can be reconstructed without the use of any personal parameters. Horizontal and vertical rotation angles of visual axis still need to be known in order to reconstruct the visual axis in 3D, but the number of calibration points required to estimate these parameters is reduced to 1.

Nagamatsu *et al.* [NSI+10] presented a model based method that completely eliminates personal calibration requirements by using a binocular setup (both eyes are tracked simultaneously). They assume that the visual axis of the left and right eyes are symmetric about the sagital plane, and ignore the vertical angle of the visual axes due to its typical low values. By these assumptions, the PoR is computed as the mid point of the points given by the intersections of both optical axis with the screen. Since reconstruction of the optical axis for each eye requires the use of two cameras, in total 4 cameras are used by this method.

A similar approach for a user-calibration-free gaze estimation system was proposed by Model and Eizenman [ME10]. They also use a binocular solution, but do not assume symmetry of the visual axis of each eye. Their method estimates horizontal and vertical rotation angles of the visual axis of both eyes (4 parameters in total) during eye tracking usage, but without relying on subjects to stare at specific points on screen. Assuming that at each time instant both visual axis stare at the same point, parameters are estimated by minimizing the distance between intersections of both visual axes with the screen.

A.3 Cross-ratio based eye tracking

In the previous section two types of remote eye gaze tracking methods were presented: interpolation based methods and model based methods. Interpolation based methods usually have simpler system setups, but the mapping function is suitable just near the calibration position. Model based methods allow free head motion, but the systems require full calibration. In this section we present a method for eye gaze tracking that combines the advantages of both interpolation and model based methods: it does not require system calibration and allows free head motion.

A method for remote eye gaze tracking based on the cross-ratio invariant property of projective geometry was introduced by Yoo *et al.* [YKLC02]. The method uses 4 light sources arranged in a rectangular shape, attached over a surface of interest. Typically this surface is the computer screen and each light source is placed at a screen corner. When a person faces the screen with these lights attached, four corneal reflections are generated on the cornea surface. These reflections, together with the observed pupil center, are then used to compute the PoR. The PoR is computed by the use of the cross-ratio, an invariant property of projective geometry. Figure A.3 illustrates the geometric setup considered for this method, where the following elements can be pointed:

- L_i: light sources (screen corners).
- G_i: corneal reflections of L_i.
- g_i: projection of G_i in the image.
- J: point of regard.
- P: pupil center.
- C: cornea curvature center.
- p: image of P.
- O: camera projection center.

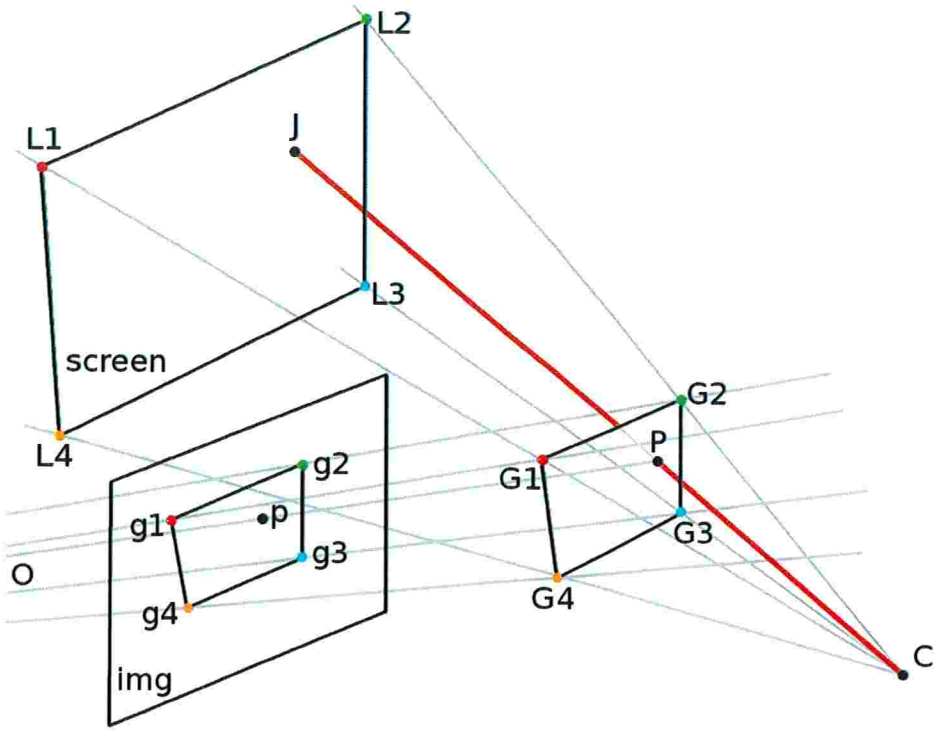

Figure A.3: *Geometric setup used by the cross-ratio method for remote eye gaze tracking.*

In its basic form, the cross-ratio based method for remote eye gaze tracking assumes g_i as projection of G_i, G_i as projection of L_i, and that each one of these groups (g_i, G_i and L_i) is coplanar, defining the planes Π_g, Π_G and Π_L (note that Π_g is coincident with the image plane). Besides that, p (in Π_g) is the projection of P (in Π_G) and P is the projection of J (in Π_L).

Points in Π_g are the result of the composition of two projective transformations of Π_L, and therefore the composition is also a projective transformation. This way, being invariant for this kind of transformation [HZ00], the cross-ratio can be used to compute J. The cross-ratio (cr) is defined for four collinear points (Q_1, Q_2, Q_3, Q_4) by:

$$\mathrm{cr}(Q_1, Q_2; Q_3, Q_4) = \frac{\delta_{13}\delta_{24}}{\delta_{14}\delta_{23}} \tag{A.1}$$

where δ_{ij} is the euclidean distance between points Q_i and Q_j.

Figure A.4 shows how the cross-ratio invariant can be applied to compute J. From the points g_i, it is possible to compute m (equivalent to the projection of M, which is the middle point of the rectangle formed by L_i), as well as the ideal points i_1 and i_2. An important property of ideal points is that the images of any pair of parallel lines crosses at

their correspondent ideal point. This way, as $\overline{L_1L_2}$ and $\overline{L_3L_4}$ are parallel at the computer screen, $\overline{g_1g_2}$ and $\overline{g_3g_4}$ can be used compute i_1. By geometric construction, lines $\overline{i_1p}$ and $\overline{i_1m}$ can be used to determine p_y and m_y as shown in Figure A.4. Similarly, p_x and m_x can be determined, being possible to define two sets, each one with 4 collinear points: $\{g_1, p_x, m_x, g_2\}$ and $\{g_1, p_y, m_y, g_4\}$. For these two sets the following ratios can be computed:

$$r_1 = cr(g_1, p_x, m_x, g_2) \tag{A.2}$$

$$r_2 = cr(g_1, p_y, m_y, g_4) \tag{A.3}$$

Due to the cross-ratio invariance to projective transformations, it is also known that:

$$r_1 = cr(L_1, J_x, M_x, L_2) \tag{A.4}$$

$$r_2 = cr(L_1, J_y, M_y, L_4) \tag{A.5}$$

Thus, given ratios r_1 and r_2, J_x and J_y (only unknown values in equations A.4 and A.5) can be computed, and consequently the point of regard J.

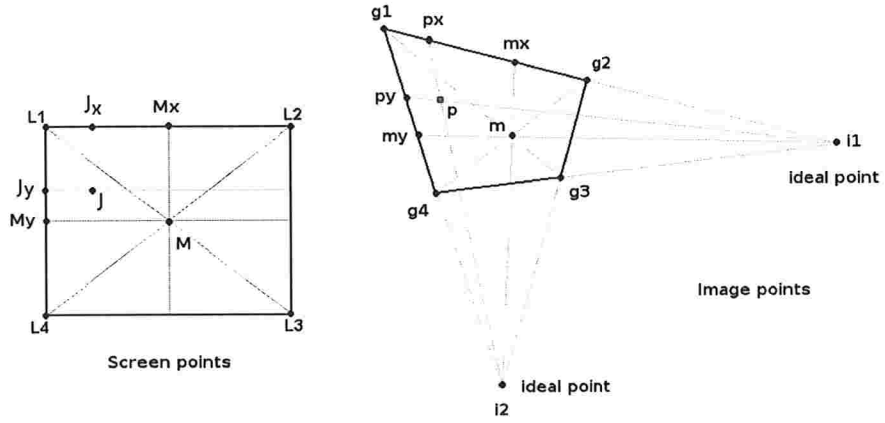

Figure A.4: *Estimation of the point of regard J using the cross-ratio invariant. First i_1 and i_2 are computed using points g_i. Next, (p_x, p_y) and (m_x, m_y) are computed and used to estimate (J_x, J_y).*

Since the cross-ratio method is based on projective transformations between planes, these transformations can also be described by means of homographies. In this case, p can

be expressed as

$$p = \mathbf{H_2}(\mathbf{H_1}(J)) \tag{A.6}$$

where $\mathbf{H_1}$ is the homography that transforms points from Π_L to Π_G and $\mathbf{H_2}$ the one that transforms points from Π_G to Π_g. Homographies $\mathbf{H_1}$ and $\mathbf{H_2}$ can be combined into a single transformation $\mathbf{H}$ that directly transforms points from Π_L to points in Π_g. Matrix $\mathbf{H}$ can be estimated from the correspondence between points g_i and L_i, and J can be computed as

$$J = \mathbf{H}^{-1}(p) \tag{A.7}$$

To facilitate the presentation and discussion of other gaze tracking methods based on the cross-ratio concept we will define the **CRf** function. The **CRf** function receives as input the points g_i and p, and also the dimensions of the rectangle formed by L_i. It returns the PoR inside the rectangle formed by L_i that corresponds to the point p in the quadrilateral formed by g_i. Since the dimensions of the rectangle formed by L_i is usually constant considering a typical scenario where we want to use the cross-ratio method to estimate gaze, we can drop the dimensions of the rectangle L_i from the input arguments for the computation of the **CRf** function. Thus, we will define the following notation for this function:

$$PoR = \mathbf{CRf}(g_i,\ p) \tag{A.8}$$

For the basic form of the cross-ratio (**CR**) method described until now, the gaze estimation procedure can be represented in a compact way by the **CRf** function.

Observe that, in theory, this method of remote eye gaze tracking does not impose any restriction on the eye position and no previous parameter value needs to be used. It is, therefore, an elegant and simple solution that tolerates head movements and is calibration-free. Unfortunately, large gaze estimation errors are observed when this basic form of the **CR** method is used. Coutinho and Morimoto [CM06], and Guestrin *et al.* [GEKE08] made a detailed investigation to explain the large observed estimation error, identifying two major sources of errors which are, in fact, two simplifying assumptions that are not valid in practice.

These assumptions are:

1. P and G_i are coplanar.

2. $\overrightarrow{CP}$ is considered as the line of sight.

The first assumption is inaccurate because pupil location relative to Π_G is a function of both the distance between P and C and the current eye rotation. Π_G, on the other hand, is defined by L_i, C, O and the curvature radius of the cornea. Therefore, there is no guarantee that P and Π_G will be coplanar. Since the **CR** method is based on transformations between planes, the estimation error will increase as distance from P to Π_G increases. It is also important to note that G_i are, in fact, not coplanar, although an approximation to a plane (Π_G) is reasonable [HAV10].

The second assumption affects gaze estimation results due to the fact that the visual axis of the eye (the true line of sight) is deviated from the optical axis. When J is computed, what is actually being computed is the point where the optical axis intercepts the screen plane. The point intercepted by the visual axis is displaced relative to J, and the observed displacement is a function of both eye distance and rotation relative to the screen.

When a more realistic geometric setup, as shown in Figure A.5, is considered, the **CR** method can not be directly applied to estimate the PoR given image points g_i and p. Consider the following elements of this new setup:

- L_i: light sources (screen corners).
- G_i: corneal reflections of L_i.
- g_i: projection of G_i in the image.
- C: center of curvature of the cornea.
- P: pupil center (coincident with iris center).
- J: intersection between optical axis and Π_L.
- P': intersection of optical axis with Π_G.

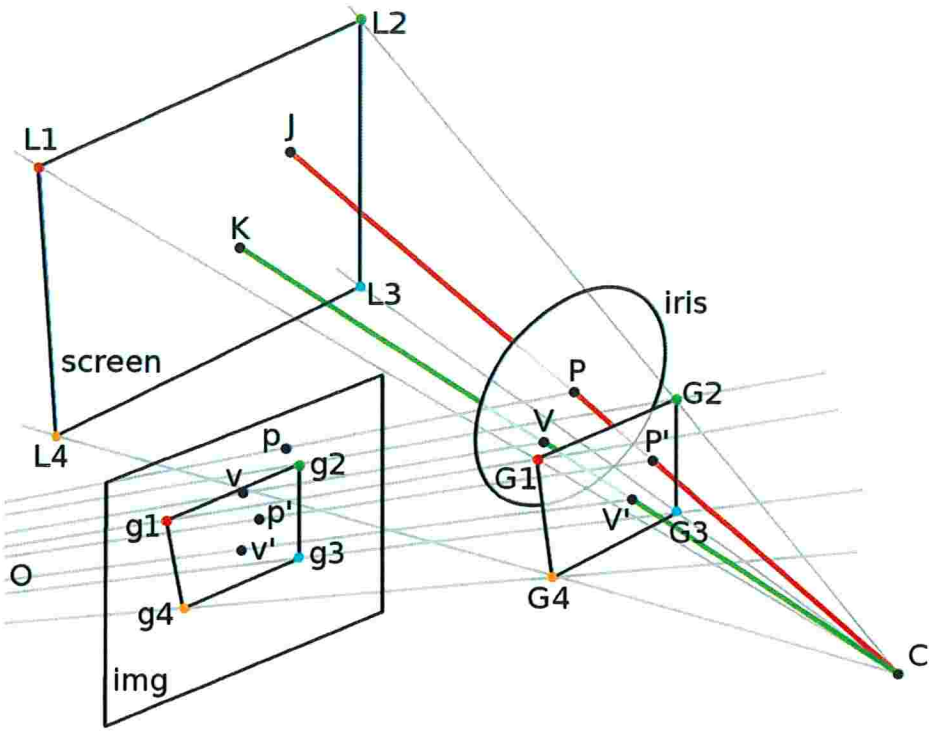

Figure A.5: *More detailed geometric setup that should be considered for cross-ratio based eye gaze tracking.*

- V: intersection of visual axis with iris disc.
- K: intersection of visual axis with com Π_L (PoR).
- V': intersection of visual axis with Π_G.
- p, p', v, v': images of P, P', V e V'.
- O: camera center of projection.

Observing Figure A.5 it is possible to notice what happens when p and g_i are directly used to compute the PoR by application of the cross-ratio. First, p is the projection of P, a point that does not belong to Π_G. Consequently it is incorrect to assume that p and g_i are images of coplanar points. Besides that, the optical axis of the eye intercepts the screen at J that does not correspond to what is being observed (point K).

New methods were developed to deal with these sources of error for the **CR** method. These methods try, to a greater or lesser degree, to employ models that approximate the more detailed scenario shown in Figure A.5. The following sections will present some of these

methods, discussing how each deals with the limitations of the **CR** method, and problems that are still left open.

A.3.1 Cross-ratio with multiple alpha correction

Yoo and Chung [YC05] improved the **CR** method by correcting the gaze estimation error caused by the non-coplanarity of G_i and P. In their solution PoR is computed in the following way:

$$PoR = \mathbf{CRf}(T_s(g_i,\ \alpha_i),\ p) \tag{A.9}$$

where T_s is a transformation defined by:

$$T_s(x,\ \alpha) = \alpha\,(x - g_0) + g_0 \tag{A.10}$$

In other words, T_s scales any image point x by α, relative to point g_0. This point is the image of the corneal reflection G_0, generated by a fifth light source that is attached near the camera's optical axis (note that when we refer to points g_i or G_i we are just considering the corneal reflections generated by the light sources attached to the screen corners). An important property of the G_0 corneal reflection, is that it belongs to line $\overline{OC}$, and as such, g_0 is the projection of C in the image plane.

The transformation of g_i points in the image plane by T_s is equivalent to perform the scale of G_i in space (relative to C), so that G_i and P become coplanar, and then projecting these transformed points into the image plane.

Each point g_i has its own scale factor α_i (because of this we will denote this method as the *cross-ratio with multiple alpha correction* method (**CR-M**α). These values are obtained by a calibration procedure where a person has to look at each L_i point. Each α_i is computed as:

$$\alpha_i = \frac{\|p_i - g_0\|}{\|g_i - g_0\|} \tag{A.11}$$

where p_i corresponds to the projected pupil center p, when the person is gazing at L_i.

The idea behind this procedure lies in the fact that it is expected that p_i perfectly matches $T_s(g_i,\ \alpha_i)$ when the eye is gazing L_i. A problem with this approach is that due to the difference between the optical and visual axis of the eye (not taken into account by the method) there is no guarantee that p_i will be in the line $\overline{g_i g_0}$. This way, the calibrated α_i parameters may not be accurate enough to compensate the non-coplanarity of P and G_i. Add to that the lack of a compensation for the κ angle, which will lead to estimated gaze points being displaced from actual observed points.

A.3.2 Cross-ratio with displacement vector correction

The *cross-ratio with displacement vector correction* (**CR-D**) method developed by Coutinho and Morimoto [CM06] is an extension **CR-M**α method [YC05], in which the error introduced due to the angle between visual and optical axis of the eye are also compensated. For this method the point of regard is computed by the following equation:

$$PoR = \mathbf{CRf}(T_s(g_i,\ \alpha),\ p) + \vec{d} \tag{A.12}$$

or equivalently by:

$$PoR = \mathbf{CRf}(g_i,\ T_s(p,\ \alpha)) + \vec{d} \tag{A.13}$$

Although in [CM06] the PoR is estimated using (A.12), we prefer to use the equivalent version of (A.13) because it is more suited to the geometric setup described in Figure A.5. In this version, instead of scaling all g_i points, just p is scaled. Since a single α value is used, it does not matter which set of points is scaled. Just keep in mind that the α used in (A.12) will be the inverse of the α used in (A.13).

The transformation of p using T_s can be thought as a way to approximately compute p', which in turn is the projection of P', the point where the optical axis intercept the Π_G plane. Since P' and Π_G are coplanar, the first source of error of the basic **CR** method is compensated.

It is not enough, though, to correct the PoR estimation. As can be seen in Figure A.5, the result of applying the function **CRf** to points g_i and p' is the J point, displaced from the

actual PoR point (K). To correctly compute K the displacement vector $\vec{d}$ must be added to J. The addition of $\vec{d}$ compensates the second source of error of the **CR** method, the displacement between J and K due to the angle that exists between the optical and visual axis of the eye.

Parameters α and $\vec{d}$ are obtained by a calibration procedure where a person gazes at a set of on screen target points. Let X be the set of n calibration points and $Y^{\alpha c}$ the set of estimated PoRs for a given αc (α candidate) without the addition of any displacement vector. Let $\Delta^{\alpha c} = \{x_i - y_i^{\alpha c} \mid x_i \in X,\, y_i^{\alpha c} \in Y^{\alpha c}\}$ be the set of displacement vectors given by the difference between calibration and estimated points. Based on the observation that for the optimum α value vectors in Δ^{α} should be approximately constant, the optimum α will be the αc value that minimizes the following summation:

$$\sum_{i=1}^{N} \parallel (x_i - y_i^{\alpha c}) - mean(\Delta^{\alpha c}) \parallel \tag{A.14}$$

After the α parameter is computed, $\vec{d}$ is taken as the mean vector of the Δ^{α} set.

A.3.3 Homography based methods

Another approach to compensate sources of errors for the basic **CR** method is to use a *homography* (**HOM**) transformation to map estimated gaze points (affected by both sources of errors) into the expected gaze points. This idea is presented by Kang *et al.* [KGME07] and Hansen *et al.* [HAV10]. In both cases the homographies used to correct estimated gaze points are obtained by a calibration procedure, where a person has to gaze at some calibration target points.

In [KGME07], the point of regard is computed by:

$$PoR = \mathbf{H_{LL}}(\mathbf{CRf}(g_i,\, p)) \tag{A.15}$$

where $\mathbf{H_{LL}}$ is a homography that transforms estimated (incorrect) points in Π_L to expected (corrected) points in Π_L. Notice that no prior processing of the points passed as input to the **CRf** function is performed.

An advantage of the homography mapping is that there is no need for the extra light

source responsible for generating corneal reflection G_0. The homography mapping can also be thought of as a generalization of the transformations realized by the **CR-Mα** (scale) and **CR-D** (scale and translation) methods, being able to correct perspective distortions.

In the homography method presented in [HAV10] the PoR is computed by:

$$PoR = \mathbf{H_{NL}}(\mathbf{CR_N}(g_i,\, p)) \tag{A.16}$$

The function $\mathbf{CR_N}$ is a variation of the **CRf** function in which the returned point is computed relative to a unitary square (normalized space), instead of being relative to the rectangle formed by L_i points. The homography $\mathbf{H_{NL}}$ then transforms estimated gaze points in the normalized space to expected gaze points in the screen space (Π_L).

The use of a normalized space adds another advantage to the homography method: the dimension of the rectangle formed by L_i does not need to be known. When the normalized space is not used and dimensions of L_i needs to be known, conversions between metric unit (physical size of the rectangle) and pixel unit must take place, during which eventual offsets between L_i rectangle and useful screen area must also be taken into account. This way, the use of the normalized space facilitates implementation, by dissociation of the Π_L plane from the plane over which we want to track a person's gaze.

In addition to the use of a normalized space, the work in [HAV10] also presents an approach to model and correct the gaze estimation error due to the planarity assumption of the pupil and corneal reflections plane.

A.4 Head movement compensation for cross-ratio based methods

The geometric modeling for the basic cross-ratio method [YKLC02] does not impose any restriction or make any assumption about the eye position and therefore allows free head motion. When more realistic geometric modeling is considered, however, two main sources of errors are introduced as pointed by Guestrin *et al.* [GEKE08] causing estimated gaze points to be shifted away from the expected gaze points. The first source of error is the non-coplanarity of the pupil center and the plane where corneal reflections are formed. The second is the angular deviation of the visual axis of the eye from the optical axis, measured by κ. The error introduced by this sources are not fixed, but varies for different eye locations and orientations (mainly due to κ, while the non-coplanarity tends to contribute in a more uniform way for different eye locations).

Correction of such sources of errors can be thought of as a transformation taking incorrect estimated gaze points into the expected points. The transformation relies on some parameters that are usually obtained by a calibration procedure and, since the error varies with eye location, the parameters will be optimal just for the calibration position. When the eye moves from the calibration position, the transformation used to correct the estimated gaze points will not be as effective, thus some estimation error will still be present.

Methods for compensating such sources of errors were introduced in [YC05, CM06, KGME07, HAV10]. Calibrated parameters are used to pre-process image features before gaze estimation [YC05], post-process incorrect gaze estimation results [KGME07, HAV10], or even combine pre and post-processing [CM06], so that the final gaze estimation is closer to what is expected.

The **CR-M**α method [YC05] just corrects the first source of error (non-coplanarity of pupil center and reflection plane). By neglecting the κ compensation, in the ideal case, gaze estimation error for any eye location will be equal to κ. As this angle is variable among different people, precision of gaze estimation for people with smaller κ angles will be better than for those with larger κ. Also interesting to note is that despite the large estimation error considering an average κ value of 5^o, this error is stable for all eye locations (in contrast with

the traditional pupil-corneal reflection technique that achieves a low gaze estimation error at calibration position, but shows a rapid increase in it even for moderate head movements, rapidly achieving gaze estimation error in the order of 5^o).

For the **CR-D** [CM06] and **HOM** [KGME07, HAV10] methods, both sources of errors are compensated. The problem with these approaches is that the parameters used to compensate κ are fixed to the calibration position (the explicit scale and displacement for **CR-D** method and embedded parameters of the homography matrix for the **HOM** method). Considering again an ideal case, gaze estimation error for the calibration position will be zero, but as the eye moves away from this location, the compensation will not be optimal, either overcompensating it or under compensating it. In both cases, however, the corrected gaze point will not exactly match the expected point. The gaze estimation error for this methods are now upper bounded by κ but in practice the error will be upper bounded by a smaller fraction of κ, a big improvement despite the not optimal compensation of head movements. Again, since κ varies among people, gaze estimation error for people with smaller κ may still be within acceptable levels under certain amounts of head movement.

This work was motivated by the observations that head movement tolerance could still be improved for cross-ratio based eye trackers. As a result, we present two new methods: the *cross-ratio with dynamic displacement vector correction* (**CR-DD**) method and the *planarization of CR features* (**PL-CR**) method.

Before introducing the new methods, a more detailed analysis of the influence of κ on gaze estimation results is presented for better understanding of how head movements affects the cross-ratio based methods.

A.4.1 The influence of κ on gaze estimation

To illustrate how κ affects gaze estimation, consider two simple scenarios shown in Figure A.6. C_i correspond to the position of the cornea center, J_i the intersection of the optical axis with the screen, and K_i the intersection of the visual axis with the screen. Assume that the values C_0, J_0, and K_0 are computed at a fixed calibration position. The first scenario, on the left of Figure A.6, shows a depth translation of the eye, and the second scenario, on the right of Figure A.6, shows a rotation. For simplicity, consider that the optical axis is

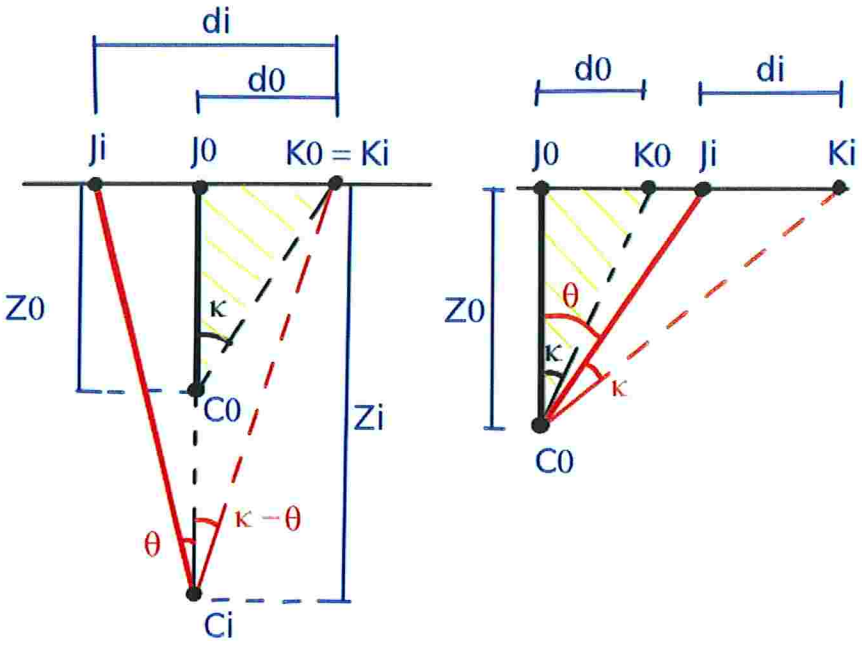

Figure A.6: *Variation of the on-screen offset d_i due to eye translations and rotations.*

perpendicular to the screen at the calibration position in both scenarios.

At the calibration position, κ can be computed as:

$$\kappa = \text{atan}(\frac{d_0}{z_0}) \tag{A.17}$$

where d_0 is the displacement between the intersections between the optical and visual axis, and z_0 is the distance of C_0 to the screen.

As the eye gets farther from the screen, to keep the gaze on the same screen position, the eye has to rotate by an angle θ. Assuming that κ is a constant eye parameter, the offset between the intersections of the optical and visual axis is now d_i. Since most gaze methods are only able to compute J_i, if a constant offset, such as d_0, is used to compensate for κ (as suggested in [CM06]),than the new gaze position would be computed as:

$$K_i' = J_i + d_0 \tag{A.18}$$

which is different than the true K_i position shown in Figure A.6. Therefore the error con-

tribution due to this constant offset would be:

$$\begin{aligned} \epsilon = \quad & \|K_i - K_i'\| = \|d_i - d_0\| = \\ & \|J_i - J_0\| = z_i tan(\theta) \end{aligned} \tag{A.19}$$

As z_i goes to infinity, the visual axis becomes perpendicular to the screen, and θ becomes equivalent to κ. This results shows that the methods that use a constant offset have an upper bound on the estimated gaze error due to translation equal to κ.

The second scenario shows a rotation around C_0. Assuming once again that κ is a constant eye parameter, a rotation of the optical axis by θ would move J_0 to J_i. From the geometry shown in Figure A.6, K_i and K_i' can be computed as:

$$\begin{aligned} K_i &= Z_0 tan(\theta + \kappa) + J_0 \\ K_i' &= J_i + d_0 = Z_0 tan(\theta) + J_0 + d_0 \end{aligned} \tag{A.20}$$

and therefore, assuming J_0 as a reference point, we can compute the estimation error due to a rotation by θ as:

$$\begin{aligned} \epsilon_\theta = \quad & K_i - K_i' = \\ & = Z_0(tan(\theta + \kappa) - tan(\theta) - tan(\kappa)) \\ & = Z_0 \frac{tan(\theta)tan(\kappa)(tan(\theta)+tan(\kappa))}{1-tan(\theta)tan(\kappa)} \end{aligned}$$

.

Observe that for large values of θ, the offset contribution due to κ becomes larger and it is not bounded since the visual axis can be parallel to the screen. Figure A.7 illustrates how the rotation error behaves for $Z_0 = 60cm$, assuming $\kappa = 5^o$ and $\kappa = 2^o$. Assuming a 19" monitor and the eye position directly in front of the center of the screen, the eye would need to rotate about 18^o to cover the whole screen. If the eye is positioned towards the edge of the screen, it would have to rotate about 36^o to look at the other end. Observe from Figure A.7 that the error for $\kappa = 5^o$ for a 20^o rotation is about 1 cm (approximately 1^o of the visual angle) and about 3cm for a 35^o rotation. For $\kappa = 2^o$, the influence on the error magnitude is much smaller.

These results show that translations of the eye parallel to the screen, that would require

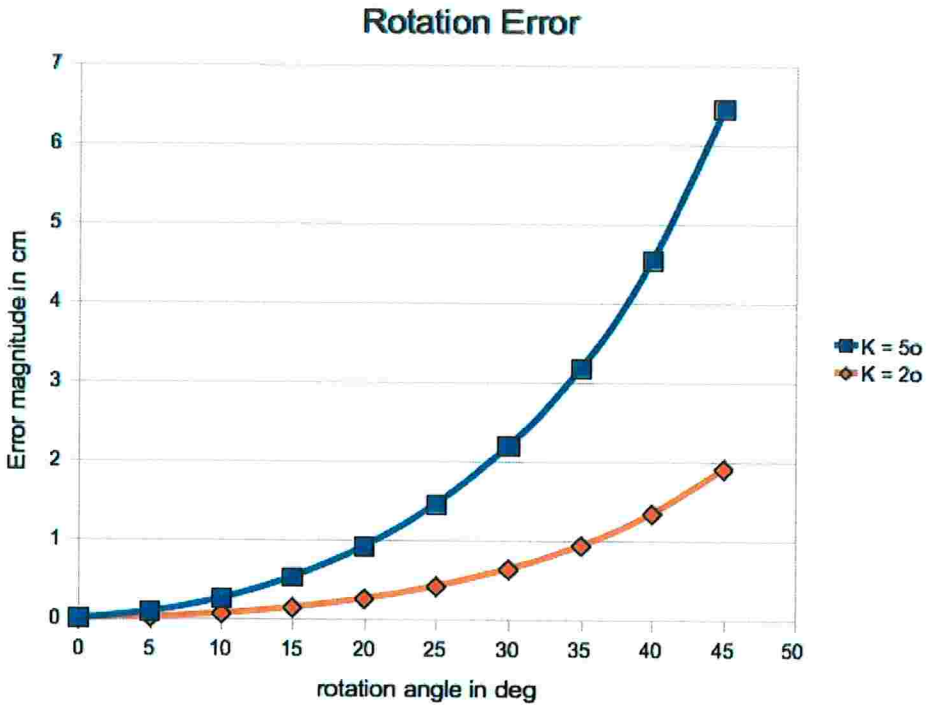

Figure A.7: *Variation of the on-screen error of a constant offset method due to eye rotation, assuming the eye is 60cm far from the screen.*

a rotation of the eye relative to the calibration position, may cause large estimation errors towards the edges of the screen, when a constant offset method is applied.

Also observe that for eyes with small κ the influence of the correction mechanisms on the gaze estimate will be smaller.

A.5 CR-DD: cross-ratio with dynamic displacement vector correction

The **CR-D** method [CM06] treats both error sources for the cross-ratio methods pointed by Guestrin *et al.* in [GEKE08]. However, as verified in [CM06], accuracy in gaze estimation still show some decay under head movements, mainly depth movements, i.e., movements when head is moved in the direction perpendicular to the screen.

The goal of the **CR-DD** method is to extend the **CR-D** method to improve gaze tracking accuracy under head movements, in particular depth movements of the head, the type of head movement that most affects the **CR-D** method. Previous analysis of the influence of κ on gaze estimation results showed why depth movements increase gaze estimation error.

If it is possible to measure the eye distance to the screen, it is possible to adjust $\vec{d}$ so that its length is adequate to the eye distance in a given moment, thus minimizing error. This solution is not ideal, since the length and orientation of $\vec{d}$ are functions of both eye distance and rotation, but it is possible to compensate a portion of the error introduced due to eye translations in z (remember that translation in this direction is the one that contributes the most for error increase of the **CR-D** method when eye moves away from calibration position).

Consider $\vec{d_0}$ the reference displacement vector obtained by the calibration procedure of the **CR-D** method, which was executed at a reference distance z_0. As $\|\vec{d_i}\|$ is directly proportional to current distance z_i (as shown in equation A.17), a more suitable displacement vector $\vec{d_i}$ for an arbitrary distance z_i can be computed by:

$$\vec{d_i} = \left[\frac{z_i}{z_0}\right] \vec{d_0} \tag{A.21}$$

A.5.1 Estimating distance variation

In order to compute the displacement vector as indicated by equation A.21, both the reference distance z_0 and the current distance z_i need to be known. Alternatively, by observation of the size of the quadrilateral formed by points g_i (corneal reflections in the image plane), it is possible to estimate the ratio z_i/z_0 without needing to know absolute values of

z_i and z_0.

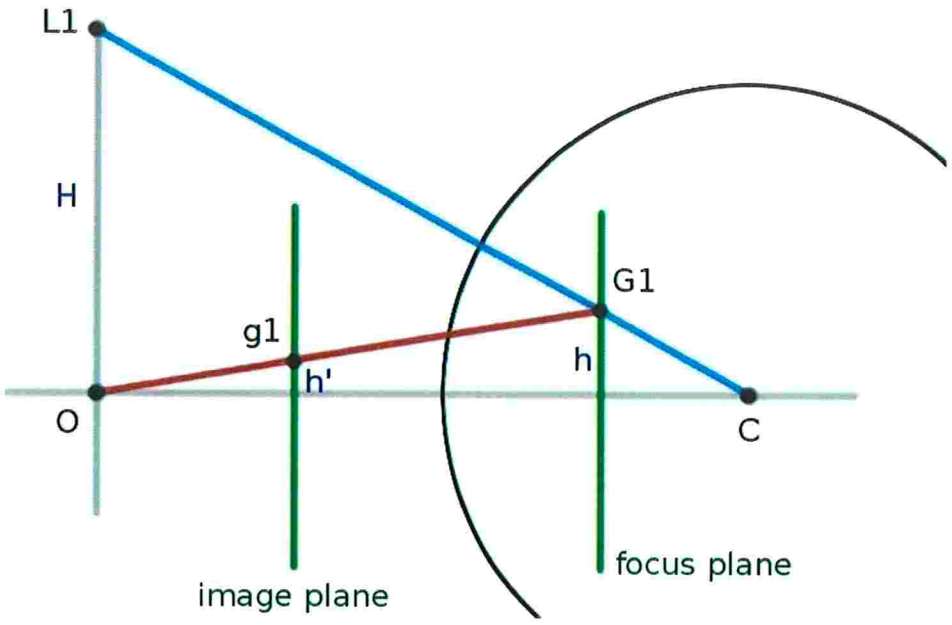

Figure A.8: *Image formation of a corneal reflection for a light source.*

Figure A.8 illustrates the image formation of a corneal reflection for a light source. Be L_1 the light source, O the camera center of projection, C the center of curvature of the cornea and z the distance from O to C. Consider also that the camera has focal length f_{cam}, that the cornea surface is a convex spherical mirror of focus f_{cor}, G_1 is formed at the focus plane and projected onto the image plane g_1. Heights h and h' (from G_1 and g_1, respectively, relative to the camera axis) are given by:

$$h = \frac{f_{cor} H}{z} \tag{A.22}$$

$$h' = \frac{f_{cam} h}{z - f_{cor}} \tag{A.23}$$

Substituting (A.22) in (A.23), we have that:

$$h' = \frac{f_{cam} f_{cor} H}{(z - f_{cor}) z} \rightarrow h' \sim \frac{f_{cam} f_{cor} H}{z^2} \tag{A.24}$$

Using the approximation for h' (justified by the fact that distance f_{cor} is much smaller than z), and considering two different distances z_i and z_0 with respective heights h'_i and h'_0,

it is possible to compute the ratio z_i/z_0 in the following way:

$$\frac{z_i}{z_0} = \sqrt{\frac{h'_0}{h'_i}} \tag{A.25}$$

Therefore, by measuring the size variation of the quadrilateral formed by points g_i, it is possible to estimate the relative translation of the eye from an initial reference position, and the displacement vector $\vec{d}$ can be adjusted accordingly.

The **CR-DD** method uses the same calibration procedure of the **CR-D** method with a few additions. At calibration distance z_0, besides computation of the α scale factor and the displacement vector $\vec{d_0}$, we also compute the reference size $size_0$ of the quadrilateral formed by g_i. The size of quadrilateral was taken as the sum of its diagonal lengths. Since during calibration a person looks to different points across the screen, $size_0$ is computed as the average of the quadrilateral sizes measured for all calibration points.

After calibration of the parameters α, $\vec{d_0}$ e $size_0$, gaze estimation is performed in the following way:

$$PoR = \mathbf{CRf}(g_i,\ T_s(p,\ \alpha)) + \frac{z_i}{z_o}\vec{d_0} \tag{A.26}$$

which is equivalent to:

$$PoR = \mathbf{CRf}(g_i,\ T_s(p,\ \alpha)) + \sqrt{\frac{size_0}{size_i}}\ \vec{d_0} \tag{A.27}$$

A.6 PL-CR: planarization of CR features method

Recall Figure A.5 that illustrates a more realistic geometric setup for the cross-ratio based methods for remote eye gaze tracking. For this scenario, the basic cross-ratio principle presented in section A.3 cannot be directly applied to estimate the point of regard K. Observing Figure A.5 it is possible to see that p is projection of P, a point that does not belong to Π_G. As a consequence it is incorrect to assume that p and g_i are images of coplanar points. Also, the intersection of the optical axis with the screen (J) does not correspond to the point that is actually being gazed (K).

It is straightforward to see that if point v' can be computed, and v' and g_i are used to compute K using the basic cross-ratio principle, then all sources of error regarding the geometric setup will be eliminated. Remember that v' is the image of V', the point where the visual axis intercepts the plane Π_G. Therefore the use of V' satisfies the two simplifying assumptions assumed by the basic cross-ratio method: V' and G_i are coplanar and V' is a point that belongs to the line of sight. The challenge is to find a way to estimate v' given image features p and g_i. Through the remainder of this section we present a solution to this problem which we called the *planarization of CR features* (**PL-CR**) method, as it brings the relevant features used by the **CR** method to a plane. In the **PL-CR** method, the PoR is computed in the following way:

$$PoR = \mathbf{CRf}(g_i,\ v') \tag{A.28}$$

Since v' is the projection of V', which in turn is defined by the intersection of $\overrightarrow{CV}$ with Π_G, there are two sub problems to solve: the estimation of $\overrightarrow{CV}$ and Π_G. The **PL-CR** method assumes G_i to be coplanar, however we will be actually computing an approximation of Π_G which minimizes the distances from G_i to the computed plane.

To keep hardware requirements of the **PL-CR** method simple we will assume a weak perspective camera model for estimation of $\overrightarrow{CV}$ and Π_G. In the weak perspective camera model, image formation can be described by an orthographic projection, followed by a scale [ET98]. The use of such camera model is justified by the fact that the size of the eye (our object of interest) is much smaller than the typical distance from the eye to the camera. For

the solutions to these sub problems, the scale component of the weak perspective model is not relevant since estimation of $\overrightarrow{CV}$ and Π_G do not take place in real world metrics. This way, a simpler orthographic camera model can be assumed.

Before presenting the solutions to each sub problem, we will first introduce an eye model and coordinate systems considered by the **PL-CR** method.

A.6.1 Eye model

In order to reconstruct the visual axis in 3D space and compute its intersection with Π_G, we will consider the eye model that is shown in Figure A.9. For this model, consider the following orthonormal coordinate system: origin at pupil/iris center P, plane xy coincident with the iris plane, with y axis pointing in the upward direction, x in the horizontal direction and z perpendicular to the iris (corresponding to the optical axis of the eye).

Relevant points for this model are the cornea center C and the point V where the visual axis intercepts the iris. C belongs to the z axis and its coordinates are given by $(0, 0, -c_z)$. V belongs to the xy plane and has coordinates $(v_x, v_y, 0)$. This model has, therefore, 3 parameters (v_x, v_y and c_z) that will be estimated by a calibration procedure. Similar to other gaze tracking methods, the calibration procedure consists of finding values for v_x, v_y and c_z that minimize the gaze estimation error for a set of calibration points. Since the model parameters are independent on eye location, the calibration procedure needs to be done just once per person.

This model is a normalized model where the cornea radius has a value of 1.0. This way, the iris radius is given by $\sqrt{1 - cz^2}$. The use of a normalized eye eliminates the need to know absolute values of the eye structures. What is important, in this case, are the ratios between model elements.

A.6.2 Coordinate systems

Besides the normalized eye model, it is also important to define 3 orthonormal coordinate systems, shown in Figure A.10: the image coordinate system $\mathbf{I}$ (represented by the $\mathbf{F_I}$ matrix), the translated image coordinate system $\mathbf{I}'$ (represented by the $\mathbf{F'_I}$ matrix) and the

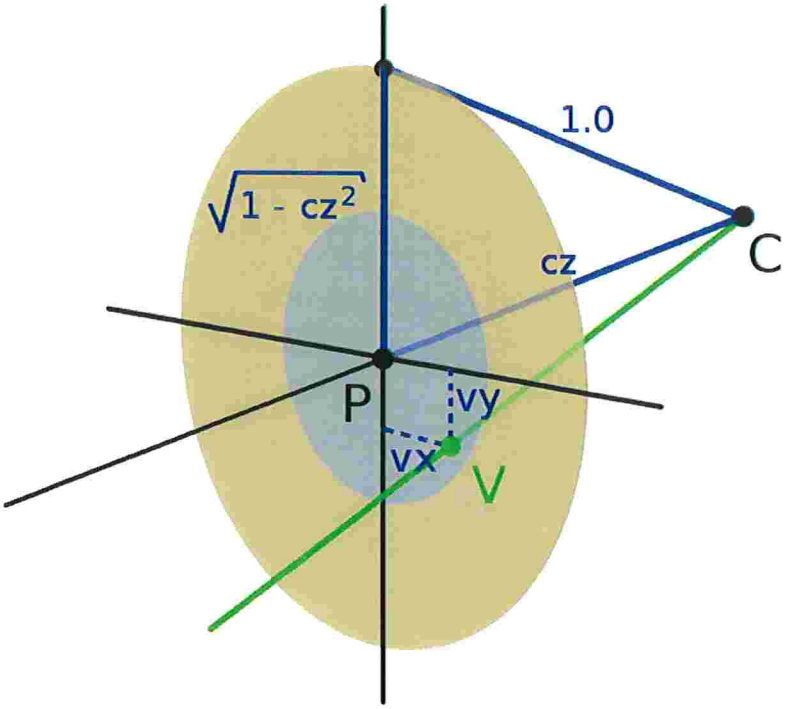

Figure A.9: *Normalized eye model*

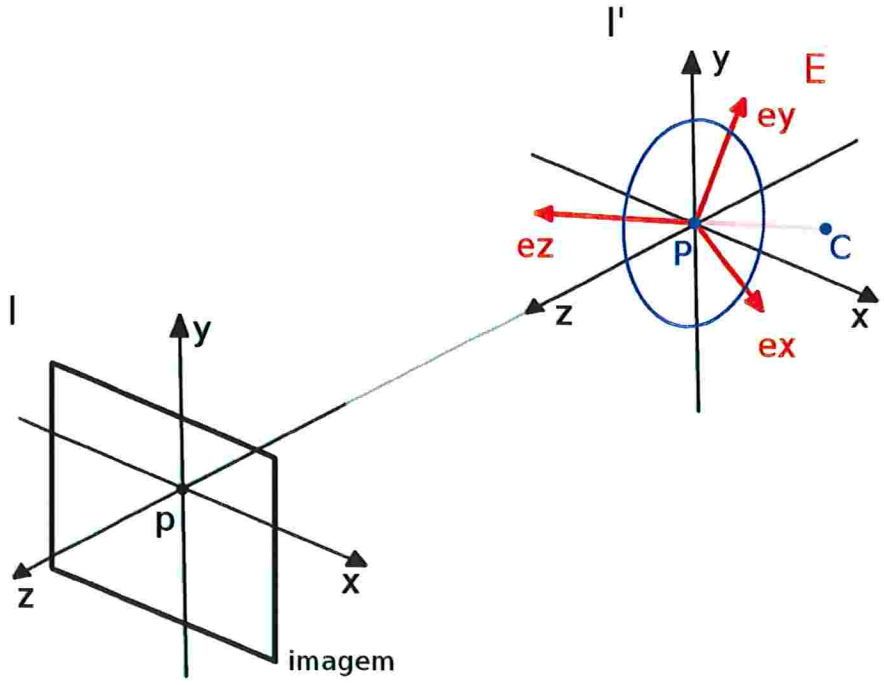

Figure A.10: *Relevant coordinate systems for the* ***PL-CR*** *method.*

eye coordinate system **E** (represented by the $\mathbf{F_E}$ matrix).

Coordinate system **I** has its xy plane coincident with the image plane, z axis perpendicular to xy, origin in p, and units given in pixels. Coordinate system $\mathbf{I}'$ has x, y and z axis equal to those from **I**, with origin in P.

Since an orthographic camera model is being used, the projection of a given point in the image plane is equivalent to the projection of the corresponding point in the plane xy of $\mathbf{I}'$. The distance between the origins of $\mathbf{I}$ and $\mathbf{I}'$ are unknown and can have an arbitrary value. We will assume $\mathbf{I}'$ to be our reference coordinate system. Estimation of the visual axis and the plane Π_G will take place relative to this reference system. This way, any point that does not have an explicit indication of a coordinate system are assumed to be relative to $\mathbf{I}'$.

Coordinate system $\mathbf{E}$ is also centered in P with its orthonormal axes are defined by:

$$\vec{e_z} = \frac{\vec{n}}{\|\vec{n}\|} \tag{A.29}$$

$$\vec{e_x} = \frac{\vec{up} \times \vec{e_z}}{\|\vec{up} \times \vec{e_z}\|} \tag{A.30}$$

$$\vec{e_y} = \vec{ez} \times \vec{e_x} \tag{A.31}$$

where $\vec{n}$ is the normal to the iris (it represents the optical axis of the eye) and the $\vec{up}$ vector is a reference to the world vertical direction. Without this reference, there would be infinite possibilities for the $\vec{e_x}$ e $\vec{e_y}$ vectors of coordinate system $\mathbf{E}$, and consequently infinite possibilities for the V_E point when transformed to the reference coordinate system $\mathbf{I}'$. Estimation of the $\vec{up}$ vector and $\vec{n}$ will be detailed in the following sections.

A.6.3 Visual axis estimation

Contrary to what happens for the optical axis, for which there is the pupil center, there is no visible eye structure associated with the visual axis. One could argue the pupil center is not a visible structure as well, but from the pupil border it is possible to have a good estimate of its center.

It is due to this lack of something "visible" that the eye model presented is important. Having knowledge of the normalized model, we can compute C and V from the observed iris pose from an eye image. Estimation of the visual axis consists of finding coordinates of C and V in the reference coordinate system $\mathbf{I}'$. C and V can be computed by the following

formulas:

$$C = s\,\mathbf{F_E}\,C_E \tag{A.32}$$

$$V = s\,\mathbf{F_E}\,V_E \tag{A.33}$$

where $C_E = (0, 0, -c_z)$, $V_E = (v_x, v_y, 0)$ and s is a scale factor given by:

$$s = \frac{r_t}{\sqrt{1 - cz^2}} \tag{A.34}$$

that has the role of scaling the normalized eye model so that its dimensions match the dimensions of the eye in the image at a given instant t, with r_t being the iris radius (in pixels) at this time. Despite the fact that the iris can have an elliptical shape in the image, its radius is given by the length of its major semi-axis.

The $\vec{up}$ vector used to define the $\mathbf{E}$ coordinate system is a reference to the real world vertical direction. The $(0, 1, 0)$ vector in the $\mathbf{I}'$ coordinate system may not correspond to the real world vertical direction if the camera is pointed upwards, downwards or is rotated around its optical axis. Assuming that the screen plane is parallel to the world vertical direction, the $\vec{up}$ vector can be inferred by the positions of the L_i light sources by:

$$\vec{up} = \frac{(L_1 - L_4) + (L_2 - L_4)}{\|(L_1 - L_4) + (L_2 - L_4)\|} \tag{A.35}$$

with L_i values expressed in the $\mathbf{I}'$ coordinate system. As will be shown in section A.6.6, L_i are computed before Π_G estimation and can be used to compute the $\vec{up}$ vector. However, as will also be shown in section A.6.6, in order to estimate L_i, C must be known, and C computation depends on the $\vec{up}$ vector to define the $\mathbf{F_E}$ matrix. How to compute C, then? Since C lies in the $\vec{e_z}$ direction, any arbitrary $\vec{up}$ vector can be used to compute it. For estimation of C we assume $\vec{up} = (0, 1, 0)$ and after L_i estimation $\vec{up}$ is recalculated using equation A.35, for computation of V.

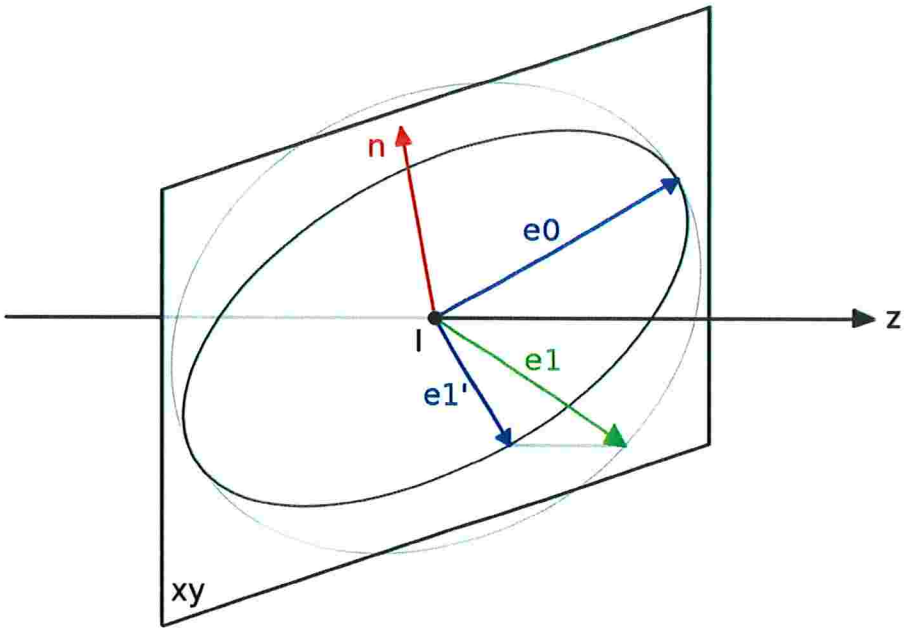

Figure A.11: *Circle (gray ellipse) centered in P, projected as an ellipse (black ellipse) in the xy plane by an orthographic projection. $\vec{e_0}$ and $\vec{e_1}$ are two radii of the circle, with $\vec{e_0}$ belonging to the xy plane, while $\vec{e_1}$ is projected to $\vec{e_1}'$ in xy. $\vec{n}$ is the normal vector to the circle.*

A.6.4 Iris normal estimation

Estimation of the iris normal $\vec{n}$ is a prerequisite to define the transformation matrix $\mathbf{F_E}$, which in turn is used to estimate the visual axis. Since the shape of the iris can be considered approximately circular, a solution that computes the normal vector of a circle in 3D space, assuming the orthographic camera model, is presented. Another solution to this problem can be find in the work by Wang and Sung [WS02].

Figure A.11 illustrates a circle centered in P (gray ellipse) and its orthogonal projection (black ellipse) on the xy plane of coordinate system $\mathbf{I}'$. Consider the vectors $\vec{e_0}$ and $\vec{e_1}$ as two radii of the circle. Vector $\vec{e_0}$ belongs to the line resulting from the intersection of the circle with the xy plane and is, therefore, the primary axis of the ellipse projected in xy. Vector $\vec{e_1}$ is outside the xy plane, but is projected as the secondary ellipse axis $\vec{e_1}'$. The normal vector to the circle can be computed by:

$$\vec{n} = \vec{e_0} \times \vec{e_1} \tag{A.36}$$

Since $\vec{e_0}$ belongs to xy, its coordinates can be taken directly from the image of the ellipse

(remember that P is the image center):

$$\vec{e_0} = (e_{0x}, e_{0y}, 0) \tag{A.37}$$

Once the primary and secondary ellipse axes in xy are perpendicular, $\vec{e_1}'$ and $\vec{e_1}$ can be described by:

$$\vec{e_1}' = (a\, e_{0_y}, -a\, e_{0_x}, 0) \tag{A.38}$$

$$\vec{e_1} = (a\, e_{0_y}, -a\, e_{0_x}, e_{1_z}) \tag{A.39}$$

where

$$a = \frac{|\vec{e_1}'|}{|\vec{e_0}|} \tag{A.40}$$

The a ratio and coordinates of vector $\vec{e_0}$ are extracted directly from the ellipse image. This way, the only missing value preventing normal estimation by (A.36) is the coordinate e_{1_z} of $\vec{e_1}$. Knowing that $|\vec{e_0}| = |\vec{e_1}|$, because both are radii of the circle, we have:

$$< \vec{e_0}, \vec{e_0} >=< \vec{e_1}, \vec{e_1} > \tag{A.41}$$

$$e_{0_x}^2 + e_{0_y}^2 = e_{0_y}^2 a^2 + e_{0_x}^2 a^2 + e_{1_z}^2 \tag{A.42}$$

$$e_{1_z}^2 = e_{0_x}^2(1 - a^2) + e_{0_y}^2(1 - a^2) \tag{A.43}$$

$$e_{1_z} = \pm\sqrt{e_{0_x}^2(1 - a^2) + e_{0_y}^2(1 - a^2)} \tag{A.44}$$

Solving (A.44) for e_{1_z}, there are two possibilities for $\vec{e_1}$ and consequently two candidates to $\vec{n}$. In the particular case of iris pose estimation a simple verification can be done to choose the correct normal vector.

A.6.5 Alternative method for iris normal estimation

In this section we present an alternative method for estimating the iris normal without having to rely on the image of the iris edges. The motivation of presenting this alternative solution is that iris edge detection is more difficult than pupil detection, mostly due to occlusion by the upper and lower eyelids.

Instead of the iris edges, this alternative method is based of the pupil center P, the c_z parameter of the eye model, and the corneal reflection G_0 (generated by a light source placed at the optical axis of the camera).

Consider the points G_0 and C and their projections on the image g_0 and c. Assuming the cornea as a spherical surface O, G_0 and C are collinear and consequently c coincides with g_0 in the image.

Since the iris normal is given directly by $\vec{n} = P - C$, the projection of $\vec{n}$ in the image will be $\vec{n_i} = p - g_0$. Remembering that a orthographic camera model is assumed, then $n_x = n_{i_x}$ and $n_y = n_{i_y}$. These values can be directly extracted from the image points p e g_0. The missing value now is the n_z of $\vec{n}$ whose module is given by:

$$|\vec{n}| = \sqrt{n_x^2 + n_y^2 + n_z^2} \tag{A.45}$$

Using the scale factor s previously introduced, it is also known that:

$$|\vec{n}| = s\ cz \tag{A.46}$$

Combining (A.45) and (A.46) we have:

$$n_z = \sqrt{s^2\ cz^2 - n_x^2 - n_y^2} \tag{A.47}$$

and consequently all components of $\vec{n}$.

A.6.6 Plane Π_G estimation

We need to find a plane in space where G_i can be taken as projections of L_i, with C being the projection center. This implies that G_i belongs to the line segment $\overline{L_iC}$. Also, if

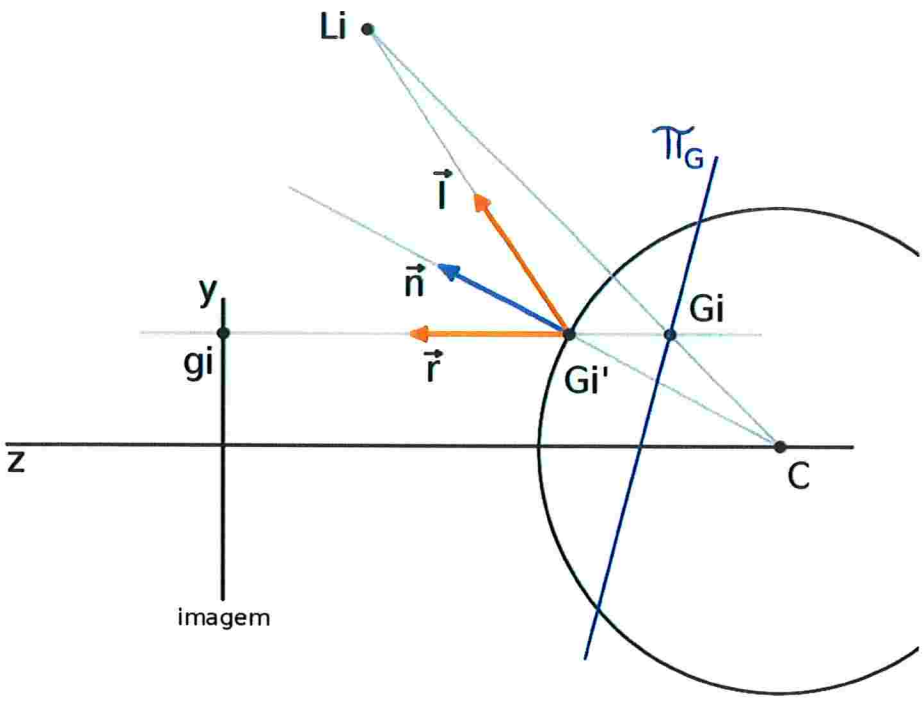

Figure A.12: *Corneal reflection formation, assuming orthographic projection. Also depicted in the figure point G_i and plane Π_G.*

G_i' is the point on the spherical cornea surface where the specular reflection due to L_i occurs and G_i' is projected to the image point g_i, then g_i, G_i' e G_i are collinear points. This way, two lines that contains G_i are defined and G_i can be computed by their intersection (see Figure A.12). It is worth noting that the intersection of two lines in 3D space can have a complicating factor because they hardly intersect at an exact point. To avoid this problem, the middle point between the smallest line segment connecting the two lines is computed as the intersection.

Computation of three among the four G_i points is sufficient to estimate the Π_G plane (the plane were all light sources L_i are projected, with C being the projection center). However, since the surface that contains all G_i points is not exactly planar, Π_G is computed as a plane such that the summation of distances between G_i and Π_G are minimized.

To compute each G_i point, two lines passing through it must be defined. The line defined by g_i and G_i' is simple to be described considering the orthographic camera assumption. By this hypothesis, coordinates x and y of g_i, G_i' and G_i are the same, and the vector

representing the reflected light ray is given by $\vec{r} = (0, 0, 1)$. The first line is then defined by:

$$R_i : g_i - a_i \vec{r} \tag{A.48}$$

with x and y coordinates of g_i being extracted directly from the image. For the z coordinate any arbitrary positive value bigger than the cornea radius can be defined to ensure that g_i is in front of the eye. In the line equation, a_i is a coefficient that represents the distance between a point in the line and g_i.

The second line is given by C e L_i, but just C is known. In order to define this second line, L_i must be computed as well. L_i can be expressed by the following equation:

$$L_i = G'_i + b_i \vec{l_i} \tag{A.49}$$

where: $\vec{l_i}$ is the vector in the direction of the light ray reaching G'_i (see A.12); G'_i can be computed by the intersection of R_i with the cornea surface (a sphere of radius s, centered in C); and finally, $\vec{l_i}$ can be obtained by reflecting $\vec{r}$ at G'_i.

If each equation for L_i is taken individually, it is not possible to compute L_i because b_i remains unknown at each equation. However, from knowledge of the distances between the L_i points (i.e., knowledge of the dimensions of the rectangle formed by L_i), an overdetermined system of six equations with four unknowns (b_1, b_2, b_3, and b_4) can be defined and solved, for example by least squares minimization. This system of equation is described in (A.50).

$$\begin{aligned}
&< L_1 - L_2 \, , \, L_1 - L_2 >= w^2 \\
&< L_2 - L_3 \, , \, L_2 - L_3 >= h^2 \\
&< L_3 - L_1 \, , \, L_3 - L_1 >= d^2 \\
&< L_3 - L_4 \, , \, L_3 - L_4 >= w^2 \\
&< L_4 - L_1 \, , \, L_4 - L_1 >= h^2 \\
&< L_2 - L_4 \, , \, L_2 - L_4 >= d^2
\end{aligned} \tag{A.50}$$

The values of w, h and d in the system correspond to, respectively, the width, height

and diagonal of the rectangle formed by L_i. Note that in our reference coordinate system units are given in pixels and therefore the values of w, h and d must be expressed in pixels as well. Conversion of such values from metric space to pixel space can be accomplished by:

$$value_p = s \frac{value_m}{r_m} \tag{A.51}$$

where $value_p$ is measured in pixels, $value_m$ in an arbitrary metric unit, r_m is the cornea radius in the same arbitrary metric unit, and s the scale factor that is equivalent to the cornea radius in pixels. For the value of r_m we used the average value of 0.78 cm.

Once L_i is computed, we are able to define the line $\overline{L_iC}$, and compute its intersection with R_i, obtaining G_i. With G_i, the plane Π_G can finally be estimated.

A.6.7 v' estimation

Once Π_G, V and C are computed, estimation of v' is straightforward. First V' is computed as the intersection of $\overline{CV}$ with Π_G. Next we project V' to the image plane. Since an orthographic camera model is used $v' = (V'_x, V'_y, 0)$.

A.7 Evaluation of the proposed methods

To evaluate the performance of the **CR-DD** and **PL-RC** methods and compare them to other cross-ratio based methods (**CR-D** and **HOM**), simulations and user experiments were conducted. To facilitate analysis and discussion of the results, in the remainder of this paper we will define two groups of methods being tested. The first group contains the methods that apply some kind of head movement compensation (HMC methods), and includes the methods proposed in this work: **CR-DD** and **PL-CR**. The second group includes the methods that do not explicitly perform head compensation (non-HMC methods) and includes the **CR-D** and **HOM** methods (the **HOM** method tested is the one described in [HAV10] without the Gaussian process error modeling).

For both simulations and experiments, evaluation consisted of measuring the gaze estimation error (in degrees) at different head positions. At each position the subjects (or the simulated eye) gazed at some screen targets and the gaze estimation error for each observed target was computed using the following formula:

$$\vec{t} = \frac{T - S}{\|T - S\|} \tag{A.52}$$

$$\vec{k} = \frac{K - S}{\|K - S\|} \tag{A.53}$$

$$error = \frac{180\ cos^{-1}(<\vec{t}, \vec{k}>)}{\pi} \tag{A.54}$$

where T is the observed target, K the estimated gaze point and S the subject's position.

The average gaze estimation error for a head position is given by:

$$\frac{1}{N}\sum_{i=1}^{N} error_i \tag{A.55}$$

where N is the number of screen targets used as test points and $error_i$ the gaze estimation error when the eye gazes at the target point number i.

A.7.1 Simulation setup

For the simulations, synthetic images generated by ray tracing were used. The LeGrand eye model was adopted for image generation and its measures were extracted from the table compiled in [GE06]. In this model the cornea and the aqueous humour are combined into a single medium (with index of refraction of 1.3375) so that refraction occurs only at the external cornea surface. The cornea has a radius of 0.78cm and the pupil center is located 0.42 cm from the cornea center. In the simulations, when the eye is directed to a given target, it is the visual axis of the eye that effectively intercepts the target. Two configurations for the visual axis were used. The first has horizontal and vertical angle values of 5^o and 1.5^o respectively. In the second configuration a horizontal angle of 2^o vertical angle of 0.6^o was used.

The simulated screen is a rectangle of 34 by 27 cm, with light source L_i positioned at each screen corner. The camera was located 2 cm below the middle point of the bottom screen border. The central point in the screen is the origin of the coordinate system used in simulations with x and y axis corresponding to horizontal and vertical direction and z axis perpendicular to the screen plane. A set of 49 test targets were used, each one corresponding to the center of the cells obtained by dividing the screen into a regular 7 $\times$ 7 grid. During calibration of each gaze estimation method, a subset of 9 points from the 49 targets was used (corresponding to the central point, corners, and middle points of the edges of the rectangle defined by 49 test points).

A perspective camera model with a vertical field of view of 5^o was used to generate the images used in the simulations. Due to its limited field of view, for each different eye position the camera was rotated to point to the center of curvature of the cornea. This camera setup mimics the conditions of the experiment setup.

Two layouts of head positions were used. The first layout is shown in Figure A.13. Head positions for this layout are given by the following coordinates: $P_0 = (0, 0, 50)$, $P_1 = (0, 0, 62.5)$, $P_2 = (0, 0, 75)$, $P_3 = (0, 0, 87.5)$, $P_4 = (-25, 0, 62.5)$, $P_5 = (-12.5, 0, 62.5)$, $P_6 = (12.5, 0, 62.5)$ and $P_7 = (25, 0, 62.5)$. Position P_1 was used to calibrate the gaze tracking methods. The second layout can be viewed in Figure A.14, with the following position coordinates: $P_0 = (0, 0, 57.5)$, $P_1 = (0, 0, 70)$, $P_2 = (0, 0, 82.5)$, $P_3 = (-12.5, 0, 57.5)$,

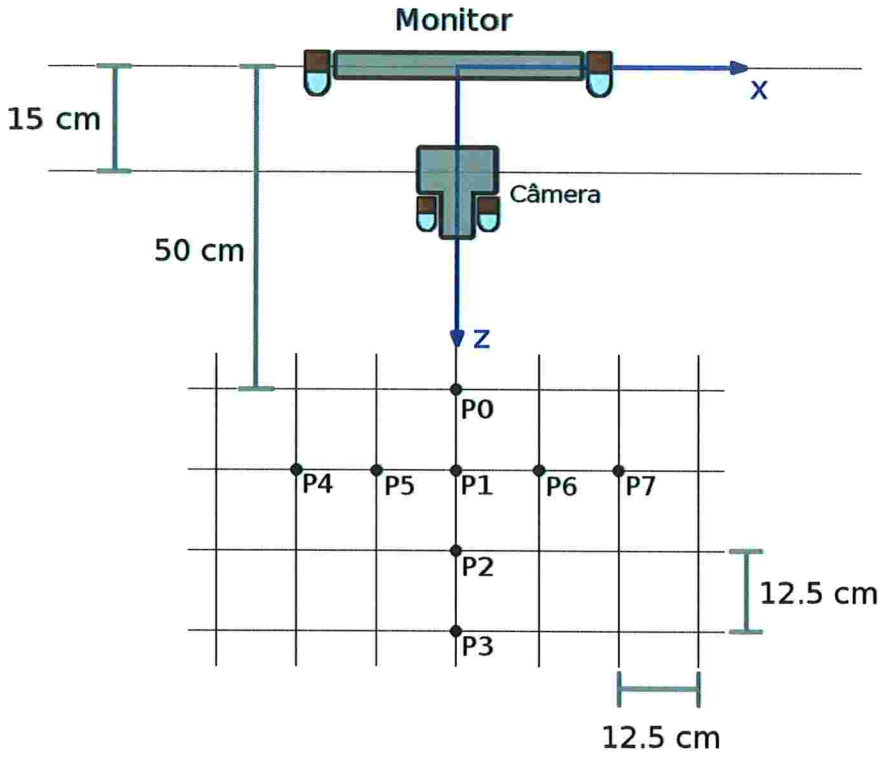

Figure A.13: *First layout of head positions used in simulations and experiment.* P_1 *was used as calibration position.*

$P_4 = (-12.5, 0, 70)$, $P_5 = (-12.5, 0, 82.5)$, $P_6 = (-25, 0, 57.5)$, $P_7 = (-25, 0, 70)$, $P_8 = (-25, 0, 82.5)$. Position P_0 was used for calibration in this case.

For each layout simulations were computed for both configurations of visual axis, resulting in a total of 4 different conditions. The layouts used in the simulation reflect those that were also used during the experiments. A first experiment was conducted using the first layout and we later redefined the set of head conditions for a second experiment (which solved some other issues encountered in the first experiment besides the head positions used). Differences between the two experiments will be discussed in more detail in section A.7.5. Note that the camera positions shown in the both layout figures are the positions used for the experiments.

The use of the second layout allowed observations of results when combining lateral and depth translations (x and z axis). Since we expected for the positive and negative translation on the x axis to have the same effect, we just considered lateral translations to the left of the screen. This avoided a big increase in the number of head positions as well. The motivation to use the second layout of positions will be discussed in more detail in sections A.7.5, A.7.6, and A.7.7.

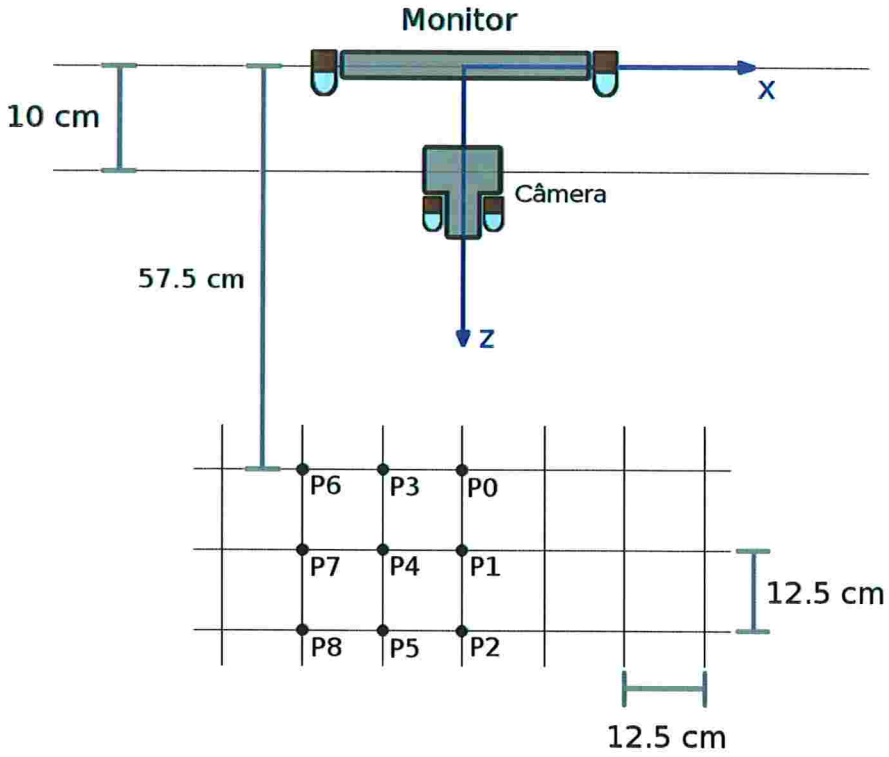

Figure A.14: *Second layout of head positions used in simulations and experiment.* P_0 *was used as calibration position.*

A.7.2 Simulation results for layout 1

Simulations for the first layout of head positions ("pure" translations in x and z) are presented in figures A.15 and A.16. Figure A.15 shows the results for the condition where the horizontal angle between the optical and visual axis is 5^o. Figure A.16 shows the result for an angle value of 2^o.

Each graph presents the average gaze estimation error for all methods at each position. The graph's vertical axis corresponds to the visual angle error in degrees. The horizontal axis corresponds to each head position. Notice that we repeat position P_1 in the graph so that we have two continuous ordered set of positions. This facilitates observation of translation effects in each axis individually. One set represents translations in the z axis and comprises positions P_0, P_1, P_2 and P_3. The other represents translations in the x axis and are formed by P_4, P_5, P_1, P_6 and P_7.

As expected, HMC methods present a better performance (smaller average gaze estimation error) than non-HMC methods as the eye moves away from the calibration position for both simulation conditions (angles of 5^o and 2^o between optical and visual axis). The major

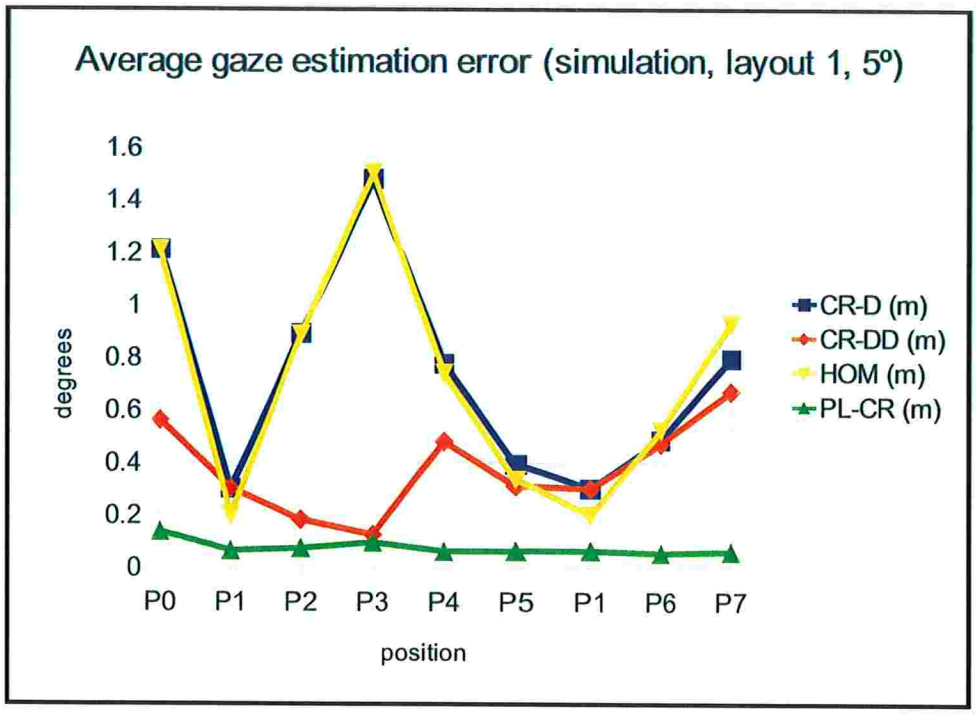

Figure A.15: *Average gaze estimation error for simulations, for $\kappa = 5^o$ and using head positions defined in layout 1.*

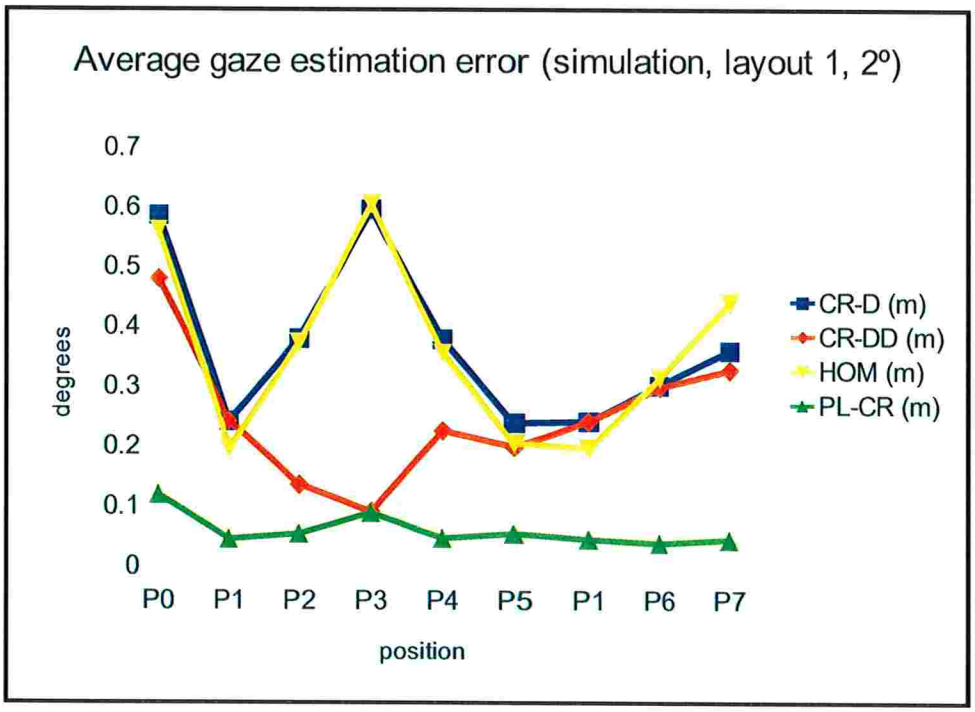

Figure A.16: *Average gaze estimation error for simulations, for $\kappa = 2^o$ and using head positions defined in layout 1.*

observed difference between the two conditions is that for the smaller angle, the error is also smaller. For the angle of 5^o, the maximum error observed for all methods and positions is 1.5^o of visual angle, while for the angle of 2^o, a maximum error of 0.6^o is observed. This

indicates that for subjects with smaller angles improvement for the HMC methods will be less noticeable than for subjects with larger angle values.

Between non-HMC methods it is possible to note they are more affected by translations in z than translations in x. This is related to how the angle affects gaze estimation results as discussed in section A.4.1, where eye distance from the screen plays a major role on the error due to κ.

The HMC methods show smaller gaze estimation error when compared to the non-HMC methods, but difference in performance between them can also be observed. The **PL-CR** method achieves better gaze estimation accuracy (maximum error of 0.13^o considering both conditions) than the **CR-DD** method (maximum error of 0.67^o for $\kappa = 5^o$ and 0.47^o for $\kappa = 2^o$). Results for the **PL-CR** method are also more stable across all positions when compared to the **CR-DD** method.

This difference is due to the different approaches of each method. Although the **CR-DD** method compensates head movement, the compensation applied is incomplete, as eye rotation is not taken into account. This effect can be observed by the smaller improvement of the **CR-DD** method (when compared to the non-HMC methods) for translations in x when compared to translations in z. The fact that the non-HMC methods are less affected by translations in x also contributes to this effect. Also note that even for translations in z the improvement is more significant at position P_3, where distance variation from the screen is maximum and the amount of eye rotation needed to scan through the entire screen is minimum. At this position the estimation error for the **CR-DD** method and **PL-CR** method are identical.

Our **PL-RC** method of estimating the point V' (intersection of visual axis with glint plane Π_G) compensates all aspects of eye movements (position and rotation). This explains smaller variation of gaze estimation error through all head positions.

A.7.3 Simulation results for layout 2

Figures A.17 and A.18 show simulation results for the second layout of head positions (average estimation error for all methods at each position). In this layout combined translations in x and z directions are considered. Again, two κ values were simulated: 5^o (Figure

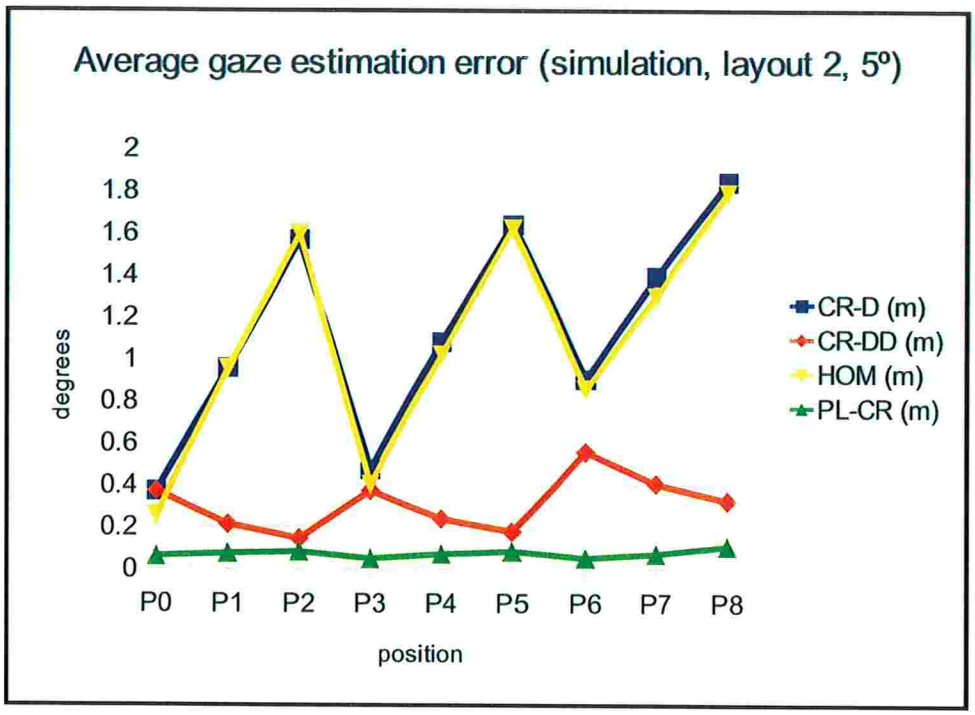

Figure A.17: *Average gaze estimation error for simulation, with a $\kappa = 5^o$ and using head positions defined in layout 2.*

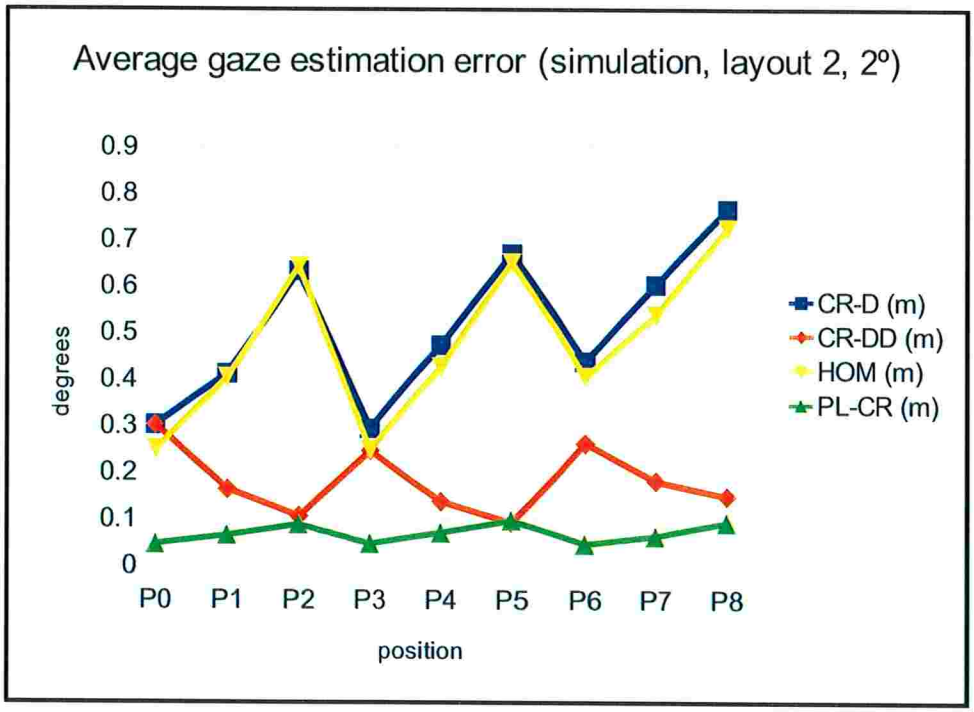

Figure A.18: *Average gaze estimation error for simulation, with a $\kappa = 2^o$ and using head positions defined in layout 2.*

A.17) and 2^o (Figure A.18).

Despite the different set of head positions used, the results are similar to those observed for the simulations using the first layout. The novelty for this simulation scenario is the

combination of x and z translations. Observing the results of the non-HMC methods it is possible to see that z translation is the major contributor to accuracy decay under head movement, but a contribution from the x translation can also be observed. As with the previous simulation, HMC methods performed better than non-HMC methods. The same observations regarding to κ and strengths and weakness of all methods also apply here.

A.7.4 Discussion of simulation results

Summing up, for all simulation conditions, the HMC methods perform better than non-HMC methods.

For the non-HMC methods, it is important to note that the accuracy decay is proportional to the magnitude of κ. Comparing the **CR-D** method with the **HOM** method it can be seen that the **HOM** performs better, specially at the calibration position. This is explained by the fact that the homography correction is more flexible than the scale and translation compensation used by the **CR-D** method.

For the HMC methods, the **PL-CR** showed better and more uniform performance when compared to the **CR-DD** method. This can be explained by the approach taken by the **CR-DD** method that focus just on compensating distance variation from the screen. Because of this the **CR-DD** performance is worse for head movements parallel to the screen. This shows the importance of also taking eye rotations into account. Nevertheless, it is a simple solution capable of improving gaze estimation. For the **PL-CR** method, on the other hand, lots of extra computation need to be done for estimation of point v' (which might be potential sources of error).

A.7.5 Experiment design

To be able to compare all methods using the same user data, the collected data was processed offline. The collected data consisted of images of the eye while the subjects gazed at the test points. As previously stated, two experiments were conducted. The first used layout 1 of head positions, while the second used layout 2. In addition to the use of the different layouts of head positions, there was also some variation between the setups of each

experiment, to improve reliability of collected data.

In general, experimental conditions are similar for those described in simulation. Some important differences need to be pointed out though.

The first is related to the screen (by screen we mean the visible area of the monitor). The simulated screen and the real screen used in the experiment had the same dimensions (34 by 27 cm). The difference was that in the simulations light sources L_i were coplanar and exactly matched the screen corners. In the experiment the light sources (infrared LEDs) did not exactly matched the screen corners, due to the monitor's border. LED positioning also varied slightly between the first and second experiment setups. Despite the fact that IR LEDs did not match the screen corners, the monitor's borders were taken into account to determine the dimensions of the rectangle defined by L_i. An important difference that should be noted is that in the first experiment the LEDs were off the screen plane by approximately 1 cm (being 1 cm closer to the subject's positions) and this difference was ignored in gaze estimation. For the second experiment LEDs were repositioned to be better aligned with the screen plane.

The camera's position relative to the screen also changed when simulation and experimental setup are compared. In simulations the camera was located in the screen plane (below the bottom border). In the first experiment the camera was displaced 15 cm towards the user. For the second experiment this distance was reduced to 10 cm. This different camera positions are not expected change results of any estimation method, with the exception of the **CR-DD** method, because changes in camera position can affect the estimation of distance variation. The camera positions of each experimental setup can be seen in Figures A.13 and A.14 of the layouts of head positions. The camera used in the experiments had a manual focus and a narrow field of view. Therefore, for each different head position the camera needed to be directed to the subject's eye and its focus adjusted.

A software was used for the data acquisition. The software was responsible for displaying a circular target at each of the 49 test points on screen and storing the image of the subject's eye. Starting from the top left point among the 49 test points, the target was displayed in a left to right and top to bottom sequence. At each test position the target stayed for about 1.3 seconds (equivalent to 40 video frames). During this time 20 images of the subject's eye

were stored. Also, during this interval, the size of the circular target varied from an initial radius of 20 pixels to a final radius of 5 pixels to serve as visual stimulus. Since multiple samples were used for each test point, the gaze estimation error for a given target point was computed as the average gaze estimation error for all samples for that target.

During the experiments a chin rest fixed to a tripod was used to maintain subjects' head still during capture at each head position. At the end of data acquisition for a given head position the chin rest was moved to the next one and the procedure was repeated.

Besides subjects' participation, data acquisition involved the participation of a operator responsible for controlling the software. The data acquisition process for each subject involved the following steps:

1. the operator explains experiment's objective, how data acquisition is done and gives the subject the consent term.
2. the operator places the chin rest at position P_0.
3. the subject sits down and accommodates his/her head in the chin rest.
4. the operator directs the camera to subject's eye.
5. the operator adjusts camera focus.
6. the operator adjusts parameters from the capture software (thresholds used in image processing).
7. the operator starts capture process.
8. the subject follows the target that scans through the 49 test points.
9. the operator places the chin rest in the next head position and steps 3 to 8 are repeated until the capture in the last position ends.

Despite the different layouts of head positions used in the two experiments, the sequence of head positions during data acquisition always followed the numbering order: positions P_0 through P_7 for the first experiment and positions P_0 through P_8 in the second.

Experiment 1 collected images of the right eye only for all subjects. We later realized that this cannot be recommended since a person might have problems with the right eye or significant preference (dominance) for the left eye. Therefore, images of both eyes were collected in Experiment 2. The data acquisition process was modified to include the data collection first for the right eye, and then for the left.

Incomplete feature vectors (where one of the expected corneal reflections of the pupil are missing) and incorrect feature detection (points that do not correspond to glints or pupil being detected as if they were) was a problem faced during processing of captured data. A manual evaluation of detected features would be impractical due to the large amount of collected images (20 samples per point $\times$ 49 test points $\times$ 8/9 positions). For this reason automated approach were used to perform this evaluation.

For the first experiment, gaze estimation results were filtered by a distance threshold between the test and estimated points. A threshold of 400 pixels have been used which is equivalent to an angular error of 12^o at 50 cm and 6.8^o at 87.5 cm distances. This are values that exceed expected gaze estimation errors and thus have been used to filter results that had a high probability of being outliers. If for a given head position a test point had no valid gaze estimation, that point was ignored when computing the average gaze estimation error for the position. This approach has an issue in that it is possible that a given feature vector produces an estimation that is filtered for a given method, while the estimation for another method is not. For this reason the outlier filtering scheme was redefined. The new scheme was used to repeat analysis of the first experiment, as well as the second experiment.

In the new scheme, instead of using a threshold on gaze estimation results, the filtering was performed on feature vectors. This way, filtering results affect all methods equally. To do such filtering we analysed the whole set of feature vectors (all samples for all test points) for a given head position. The analysis was carried in two steps: first feature vectors that presented incorrect corneal reflections were discarded and after that features that presented bad pupil detection was discarded as well.

To discover which corneal reflections were incorrect the following approach was taken: using g_0 as a reference point, and considering all 49 $\times$ 20 samples for a given head position, a representative point for g_1, g_2, g_3 and g_4 were computed. It is not expected that every

g_i sample matches its representative point, but it is expected that the set of all g_i samples are clustered around its representative. A radius of $W/7$ pixels was defined, where W is the width of the approximate rectangle formed by g_i corneal reflections. Feature vectors in which not all g_i are within this radius (relative to its representative) were discarded.

For pupil filtering a similar approach is taken, but considering just the 20 samples for a given test point. It is also expected that detected pupils are clustered around a representative point and a maximum radius of $W/14$ (equivalent to a diameter of $W/7$) was used. This filtering is useful for the cases where the pupil was not correctly detected or in cases where the subject moved his/her eye ahead of the target point due to prediction.

The representative point R for a set of $Q_i = (q_{i_x}, q_{i_y})$ points was computed as:

$$R = (\, median(q_{i_x}), \; median(q_{i_y}) \,) \tag{A.56}$$

The median was used because incorrect detected features usually displays a large displacement from the expected location which would affect computation of a representative based on average values.

A.7.6 Results of experiment 1

A group of 9 subjects participated in experiment 1. In this first experiment gaze was estimated from images of the right eye for all subjects. Gaze estimation results for each of them are shown in figures A.19 through A.27. Results shown in this figures were obtained when the outlier filtering scheme based on a distance threshold between estimated and expected points was used. Results obtained using the same collected data but the outlier filtering scheme based on the filtering of feature vectors are shown in figures A.29 through A.37.

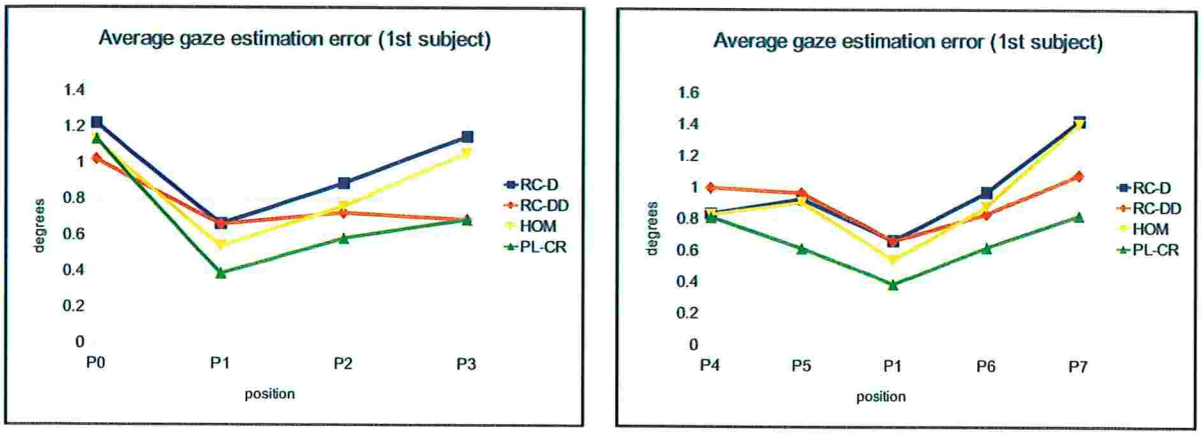

Figure A.19: *Average gaze estimation error for the 1st subject of first experiment. On the left results when head moves along the z axis. On the right results move head movements along the x axis.*

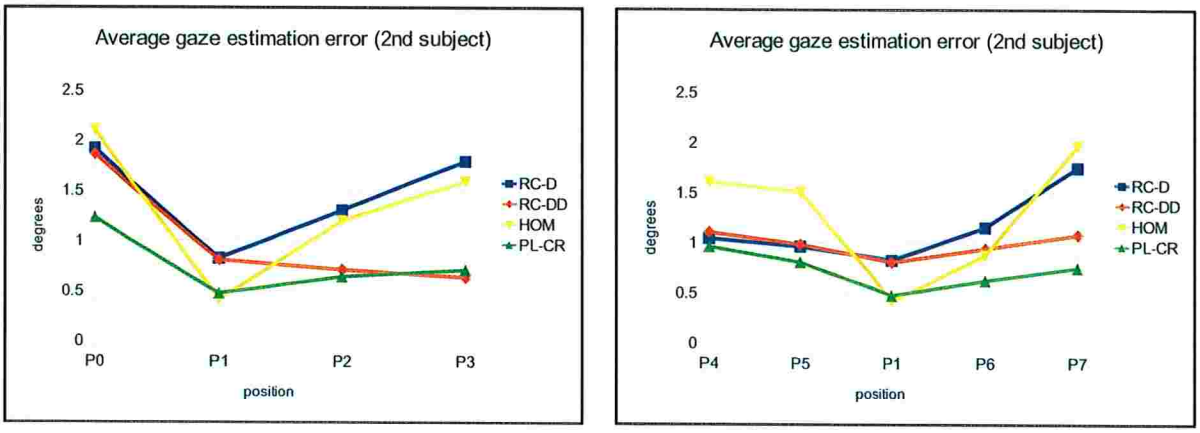

Figure A.20: *Average gaze estimation error for the 2nd subject of first experiment. On the left results when head moves along the z axis. On the right results move head movements along the x axis.*

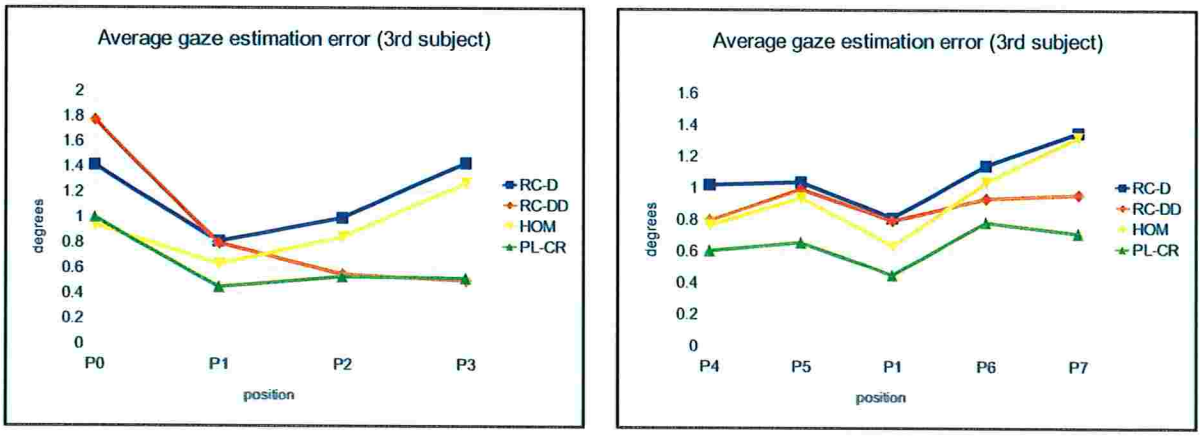

Figure A.21: *Average gaze estimation error for the 3rd subject of first experiment. On the left results when head moves along the z axis. On the right results move head movements along the x axis.*

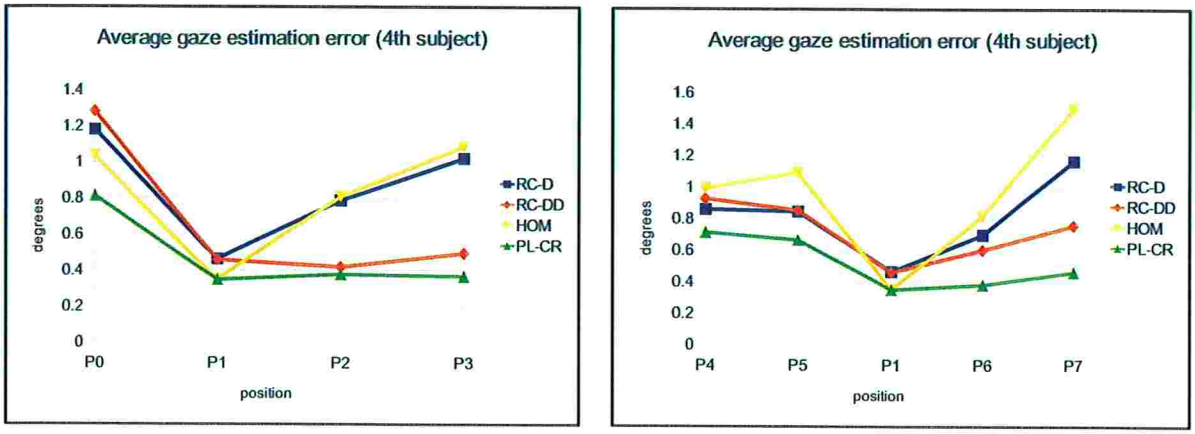

Figure A.22: *Average gaze estimation error for the 4th subject of first experiment. On the left results when head moves along the z axis. On the right results move head movements along the x axis.*

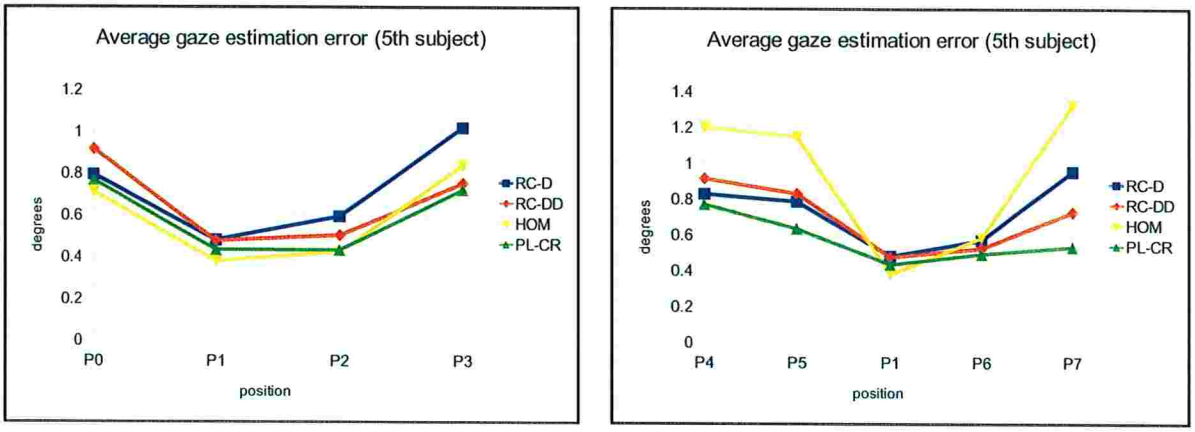

Figure A.23: *Average gaze estimation error for the 5th subject of first experiment. On the left results when head moves along the z axis. On the right results move head movements along the x axis.*

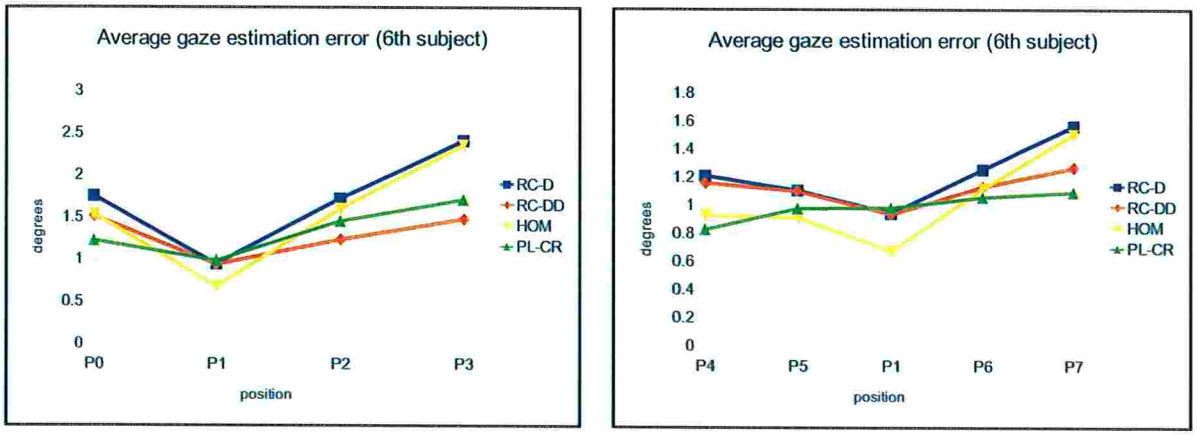

Figure A.24: *Average gaze estimation error for the 6th subject of first experiment. On the left results when head moves along the z axis. On the right results move head movements along the x axis.*

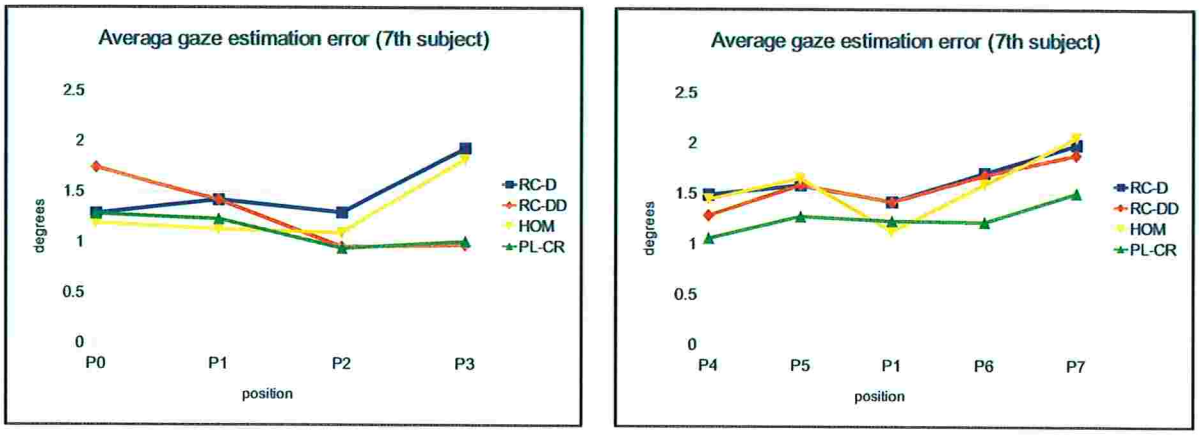

Figure A.25: *Average gaze estimation error for the 7th subject of first experiment. On the left results when head moves along the z axis. On the right results move head movements along the x axis.*

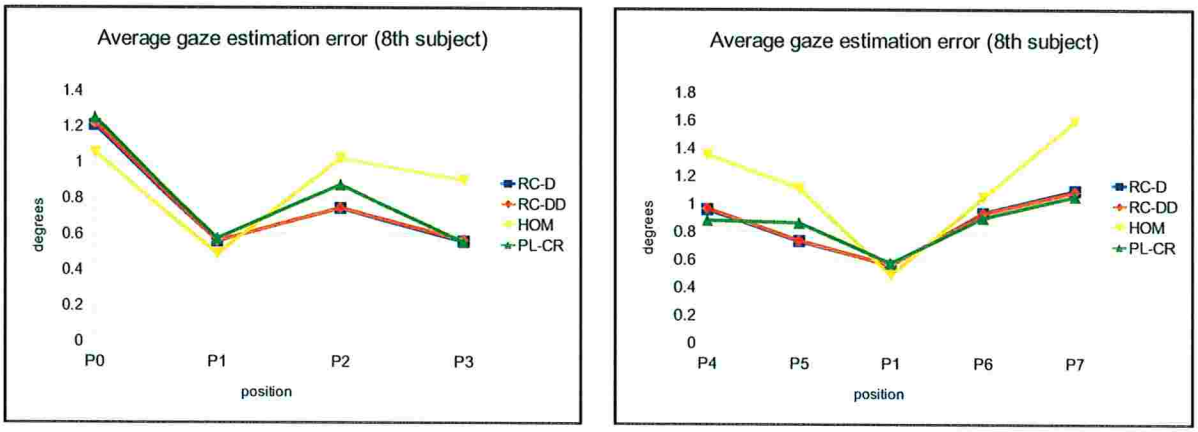

Figure A.26: *Average gaze estimation error for the 8th subject of first experiment. On the left results when head moves along the z axis. On the right results move head movements along the x axis.*

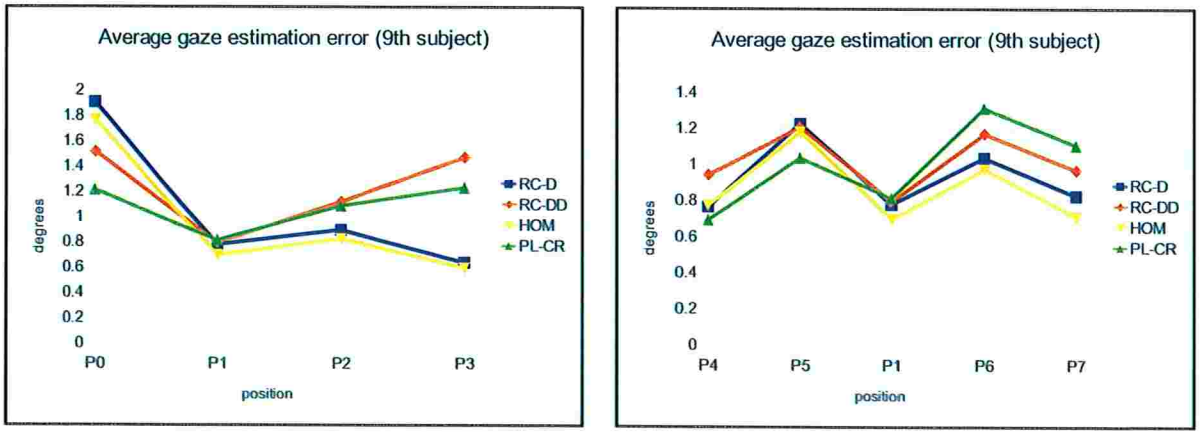

Figure A.27: *Average gaze estimation error for the 9th subject of first experiment. On the left results when head moves along the z axis. On the right results move head movements along the x axis.*

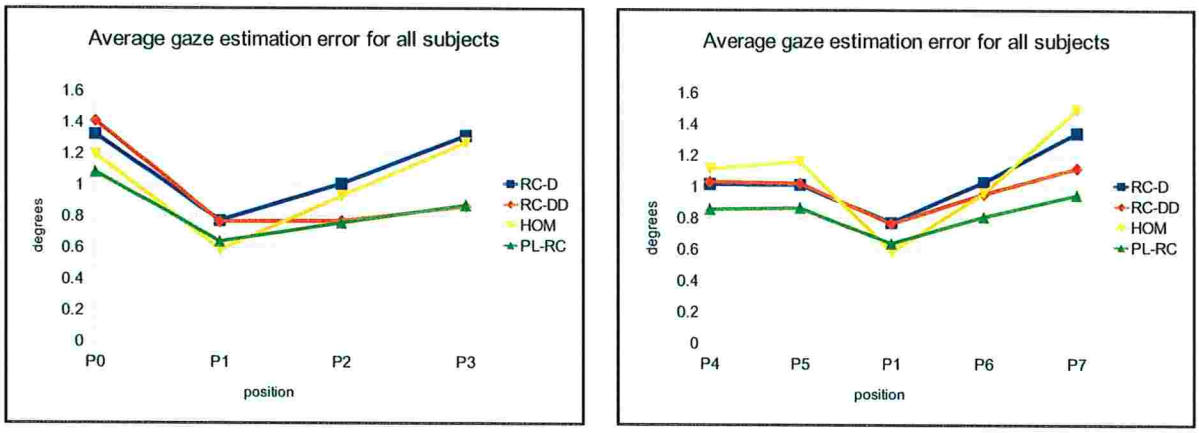

Figure A.28: *Average gaze estimation error for all subjects that participated in the first experiment. On the left results when head moves along the z axis. On the right results move head movements along the x axis.*

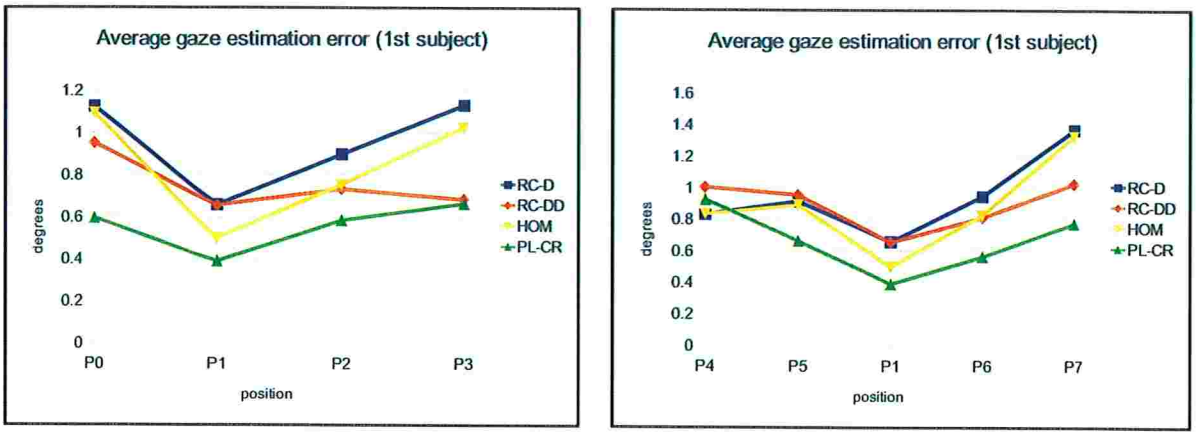

Figure A.29: *Average gaze estimation error for the 1st subject of first experiment, using second approach for outlier filtering. On the left results when head moves along the z axis. On the right results move head movements along the x axis.*

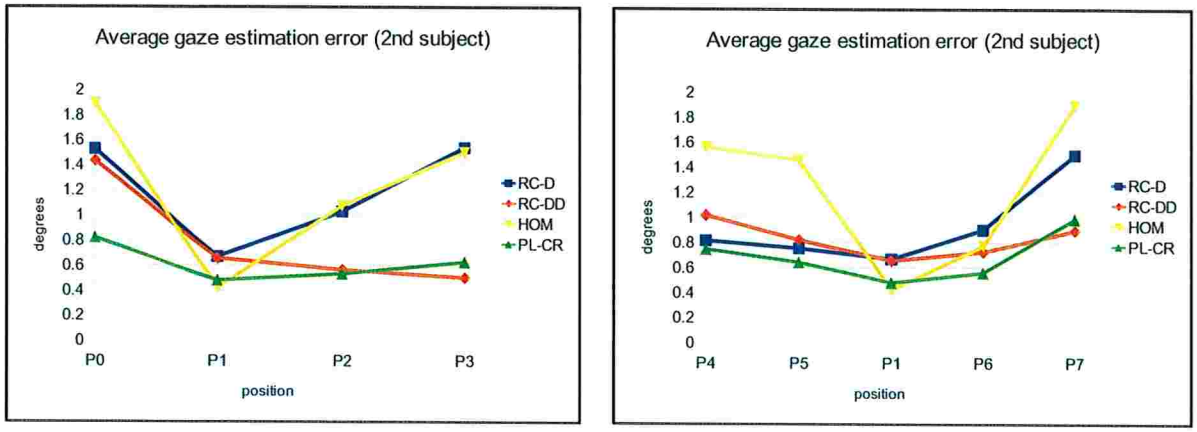

Figure A.30: *Average gaze estimation error for the 2nd subject of first experiment, using second approach for outlier filtering. On the left results when head moves along the z axis. On the right results move head movements along the x axis.*

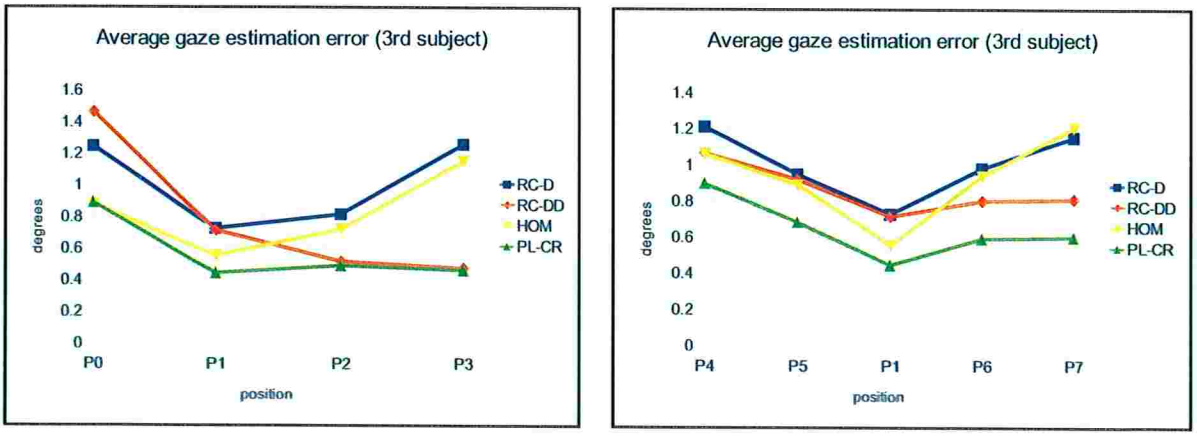

Figure A.31: *Average gaze estimation error for the 3rd subject of first experiment, using second approach for outlier filtering. On the left results when head moves along the z axis. On the right results move head movements along the x axis.*

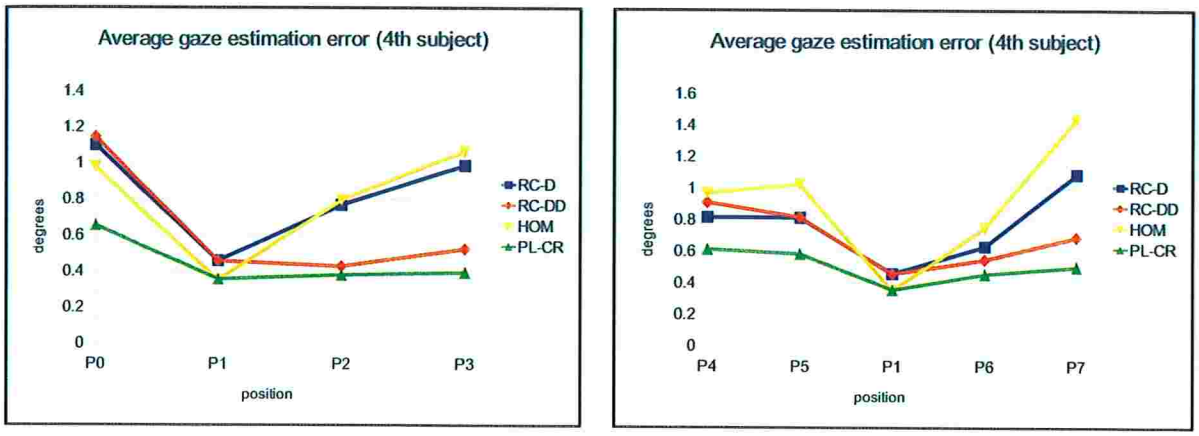

Figure A.32: *Average gaze estimation error for the 4th subject of first experiment, using second approach for outlier filtering. On the left results when head moves along the z axis. On the right results move head movements along the x axis.*

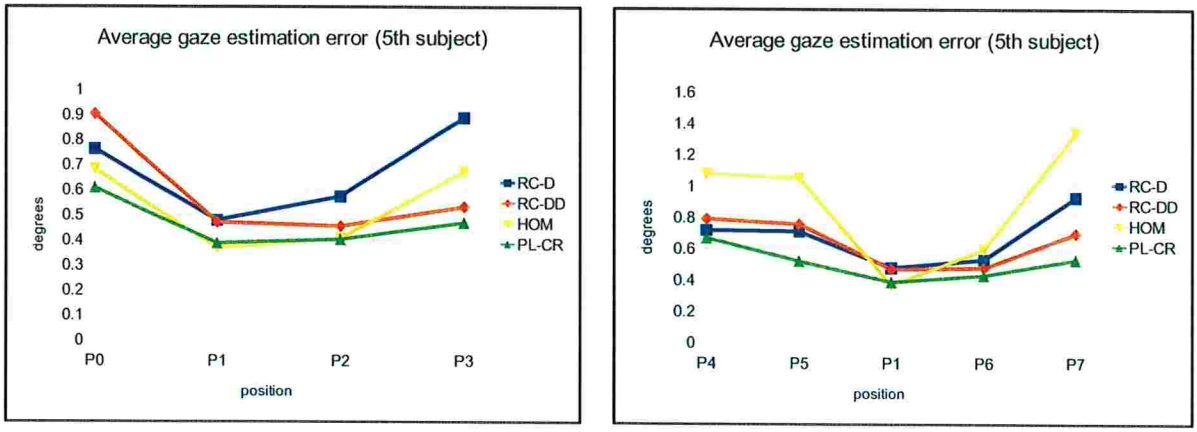

Figure A.33: *Average gaze estimation error for the 5th subject of first experiment, using second approach for outlier filtering. On the left results when head moves along the z axis. On the right results move head movements along the x axis.*

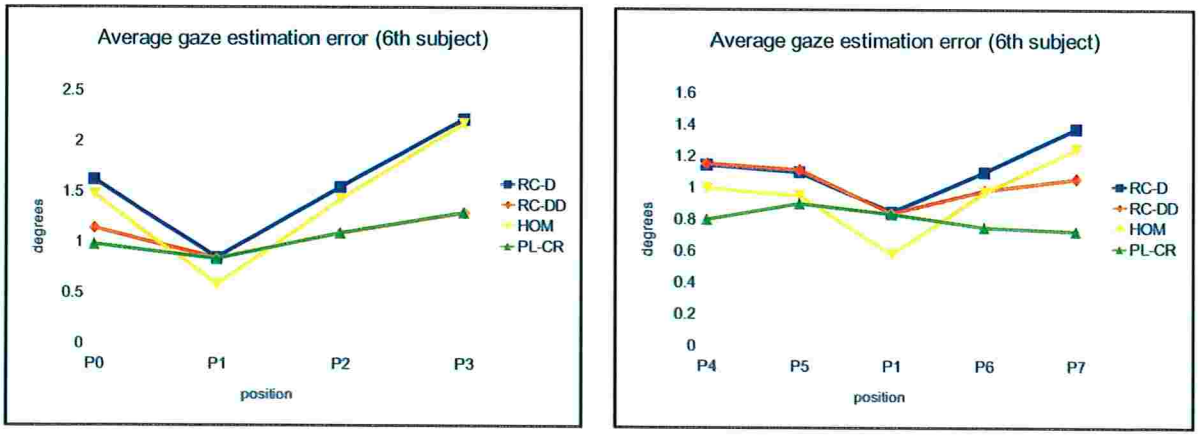

Figure A.34: *Average gaze estimation error for the 6th subject of first experiment, using second approach for outlier filtering. On the left results when head moves along the z axis. On the right results move head movements along the x axis.*

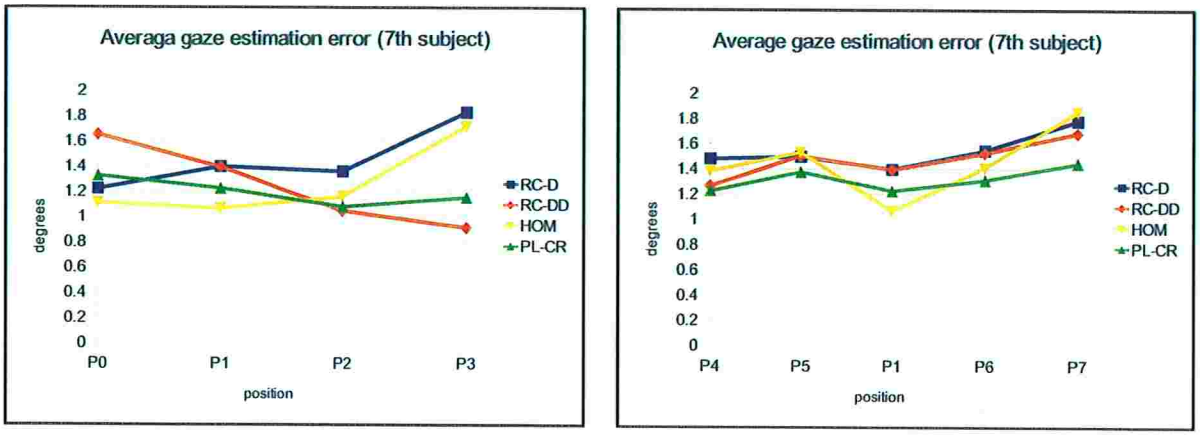

Figure A.35: *Average gaze estimation error for the 7th subject of first experiment, using second approach for outlier filtering. On the left results when head moves along the z axis. On the right results move head movements along the x axis.*

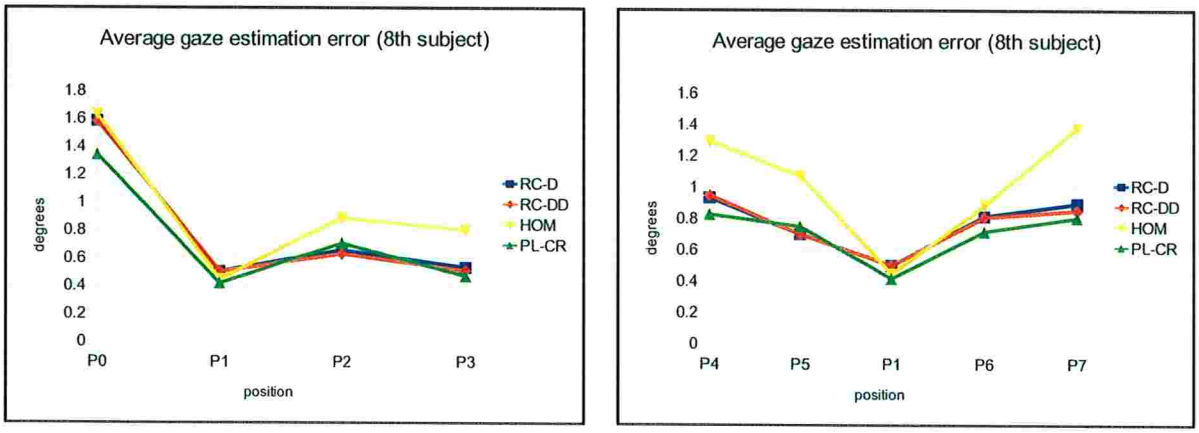

Figure A.36: *Average gaze estimation error for the 8th subject of first experiment, using second approach for outlier filtering. On the left results when head moves along the z axis. On the right results move head movements along the x axis.*

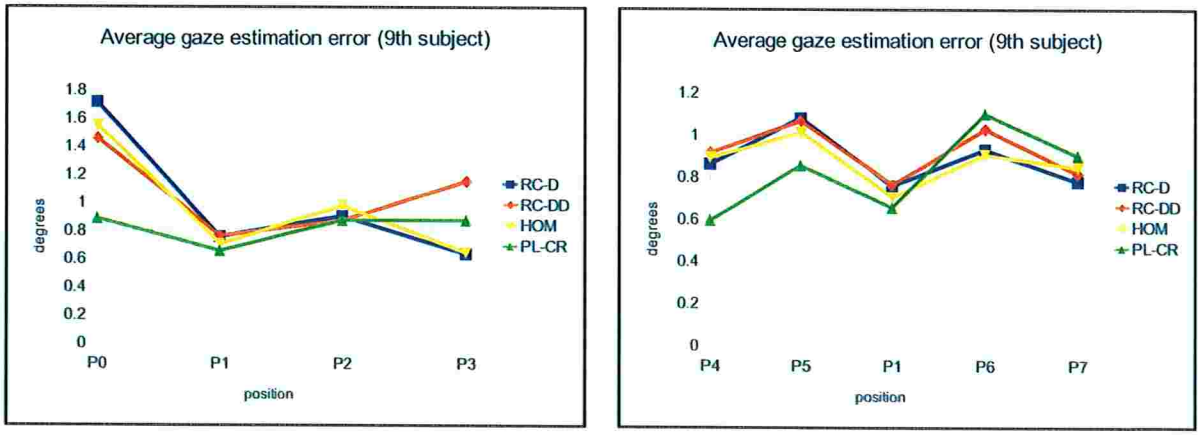

Figure A.37: *Average gaze estimation error for the 9th subject of first experiment, using second approach for outlier filtering. On the left results when head moves along the z axis. On the right results move head movements along the x axis.*

In section B.1, a detailed view of the gaze estimation results are shown. These detailed results were obtained when the outlier filtering scheme based on the filtering of feature vectors was used. Due to the large amount of graphs we preferred to present them in a separate section. Each figure from the section B.1 exhibits the results for one of the 9 subjects and one gaze tracking method (**CR-D**, **CR-DD**, **HOM**, or **PL-CR**). Also, each figure is composed by a set of heatmaps, where each heatmaps corresponds to a head position. The heatmaps are organized following the layout of head positions shown in Figure A.13. Each heatmap displays the distribution of the gaze estimation error across all test points of the screen. Cooler colors represent smaller estimation errors, while hotter colors represent larger errors. Note that the very dark blue color indicates screen points for which gaze estimation failed.

As in the simulations, we also expected the HMC methods exhibit better performance than the non-HMC methods, as the head moves away from the calibration position P_1. These expectations were met for subjects 1, 2, 3, 4, 5, 6, and 7 where HMC methods achieved smaller gaze estimation errors than non-HMC methods for most head positions. Besides that, error for HMC methods grew at a lower rate when compared to non-HMC methods under head movements. Comparing the results for each outlier filtering scheme, it is possible to see that results are very similar, with slightly improvements for all methods when the filtering of feature vectors was used. Just for position P_0 the difference is more noticeable, with observed improvement when the second scheme was used. The reason for this is that at P_0 corneal reflection formation was affected for many cases, mainly at the top row of test

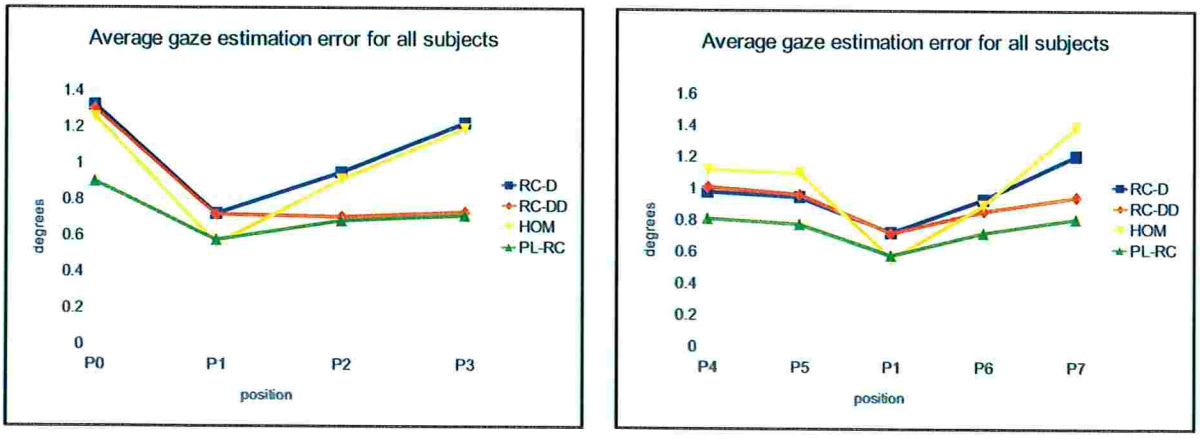

Figure A.38: *Average gaze estimation error for all subjects that participated in the first experiment, using second approach for outlier filtering. On the left results when head moves along the z axis. On the right results move head movements along the x axis.*

points, and the second filtering scheme was more effective detecting such cases.

Subject 8 is a case where the subject's κ angle is small. An estimation for κ for this subject can be derived by the calibrated $\vec{d}$ parameter used for the **CR-D** and **CR-DD** methods, which had a value of $(-5.85, -10.74)$ (units in screen pixels), which, at P_0 represents a angle of 0.31^o. As discussed in section A.4.1, in this case, head translations cause less decay of accuracy for the non-HMC methods, and the results for both HMC and non-HMC methods will be similar. This can be confirmed by the very close results for **CR-D** and **CR-DD** and **PL-CR** methods.

Subject 9 shows a case where results did not meet the expectations. Surprisingly, the methods that do not model any kind of head movement compensation (**CR-D** and **HOM**) achieved higher estimation accuracy across most head positions.

Despite the results for subject 9, in general the observed results met the expectations. Combined results for all users are shown in figures A.28 (first outlier filtering scheme) and A.38 (second (second outlier filtering scheme) where average estimation error for all users are displayed. For the results shown in figure A.28, average error ranges from 0.76^o to 1.41^o to the **CR-DD** method, 0.64^o to 1.08^o for the **PL-CR** method, 0.77^o to 1.34^o for the **CR-D** method and 0.58^o to 1.49^o for the **HOM** method. For the results in figure A.38, average error ranges from 0.7^o to 1.3^o to the **CR-DD** method, 0.58^o to 0.9^o for the **PL-CR** method, 0.72^o to 1.32^o for the **CR-D** method and 0.55^o to 1.39^o for the **HOM** method. When compared to the non-HMC methods, improvement in gaze estimation for the **CR-DD** method are more noticeable for translations in z than for translations in x. Also the **CR-DD** method performs better for larger distances. Improvement in gaze estimation for the **PL-CR** method are observed for head movements in both x and z directions. This is consistent with the simulation results.

A.7.7 Results of experiment 2

After realization of experiment 1 we noticed that for many test points at position P_0 (mostly the top row of test points, those for which the eye needed more rotation), corneal reflections G_i were formed too close to the iris/sclera boundary producing distorted reflections, and even not forming reflections at all. With the corneal reflection formation being compromised by the close distance from P_0 to the screen, we redesigned the conditions of the second experiment.

There were also some other issues with the first experiment that motivated us to redesign it and conduct a second one: IR LEDs were attached about 1 cm off the screen plane introducing a potential source of error; robustness of automatic feature detection could still be improved reducing the number of cases where incorrect features (pupil or corneal reflections) would be used as valid data; and finally, gaze was estimated just for the subjects' right eyes, but gaze estimation for both eyes could give a more complete view of the results. While redesigning the experiment the layout of head positions was also changed to include positions that combined translations in both x and z axis.

The motivation to redesign the layout of head positions, resulting in the layout presented in figure A.14, was to ensure proper corneal reflection formation at all positions. Maximization of the useful area of corneal reflection formation is an important challenge when designing gaze tracking devices based on multiple glints, because even that a method allows for large head movements in theory, if the area of corneal reflection formation is small then freedom of head movement is constrained. The screen size plays a major role in definition definition of head positions. The larger the screen, the larger the corneal reflection pattern, which in principle would be good, but the chance of corneal reflections dropping out of the cornea also increases. The angle of the cone of light provided by the LEDs were also important in determining the new positions. The use of infrared LEDs with an illumination cone of 30^o also restricted proper illumination of the eye at closer distances. It was also important to consider maximum distance as well. As the eye moves away from the screen, the corneal reflection pattern becomes smaller, which restricts accuracy of gaze estimation. Feature detection noise will also proportionally be larger for larger eye distances.

A group of 7 subjects participated in the second experiment. Gaze estimation results for

each subject are shown in the Figures A.39 through A.45. Detailed views of the gaze estimation results for this second experiment are also shown in section B.2. Each figure in section B.2 is composed by a set of heatmaps. Each heatmap corresponds to a head position and displays the distribution of the gaze estimation error across the screen. Also, the heatmaps are organized following the second layout of head positions as shown in Figure A.14. For subjects 1, 2, 4, 5, 6, and 7 the results reflect what was expected. Some particularities can be pointed for subjects 6 and 7 though.

Results for subject 6's right eye are quite reasonable with the exception of test position P_6. For the left eye of subject 6, results show that although the HMC methods perform better then non-HMC methods, results are relatively similar and head movement affects all methods equally. For this case, observe that the size of the displacement vector obtained by

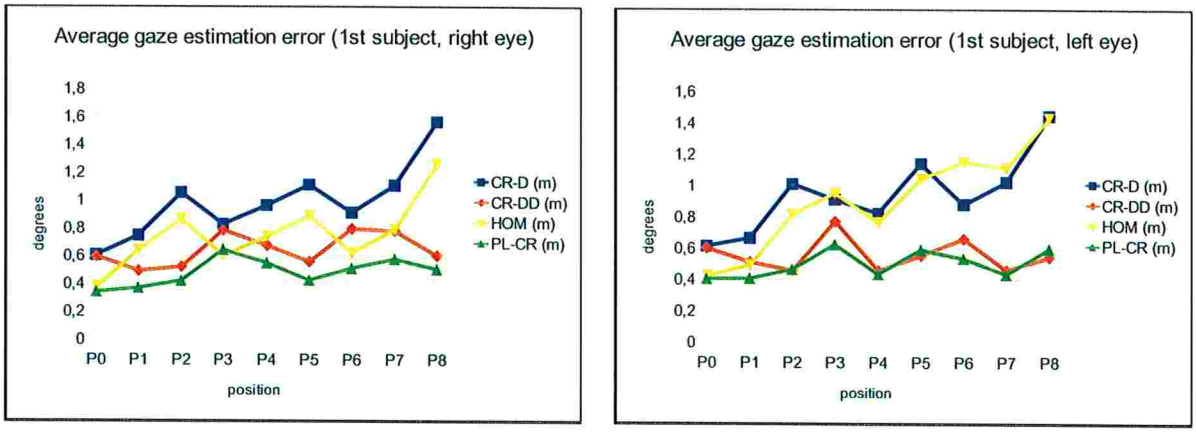

Figure A.39: *Average gaze estimation error (measured in degrees of visual angle), for left and right eyes of 1st subject, at test positions P_0 through P_8. The right eye is the dominant eye for this subject.*

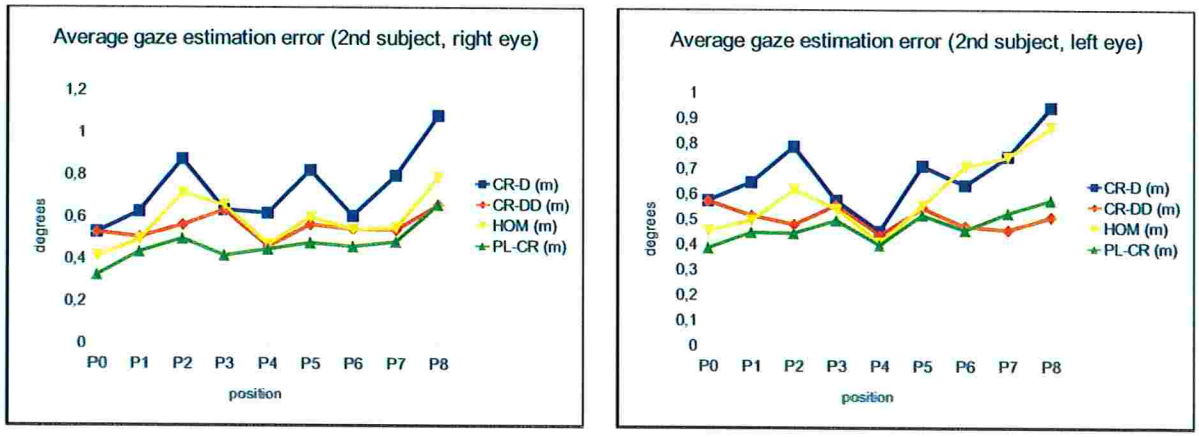

Figure A.40: *Average gaze estimation error (measured in degrees of visual angle), for left and right eyes of 2nd subject, at test positions P_0 through P_8. The left eye is the dominant eye for this subject.*

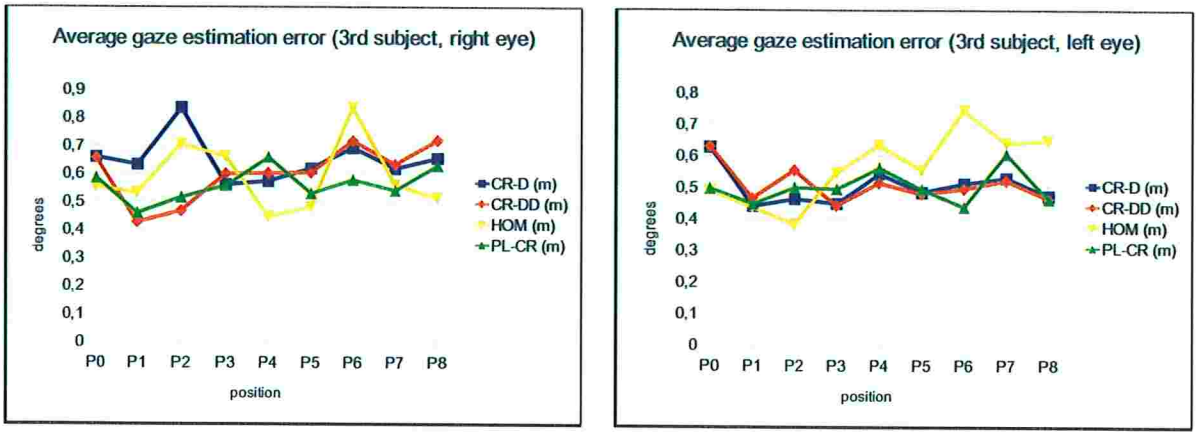

Figure A.41: *Average gaze estimation error (measured in degrees of visual angle), for left and right eyes of 3rd subject, at test positions P_0 through P_8. The right eye is the dominant eye for this subject.*

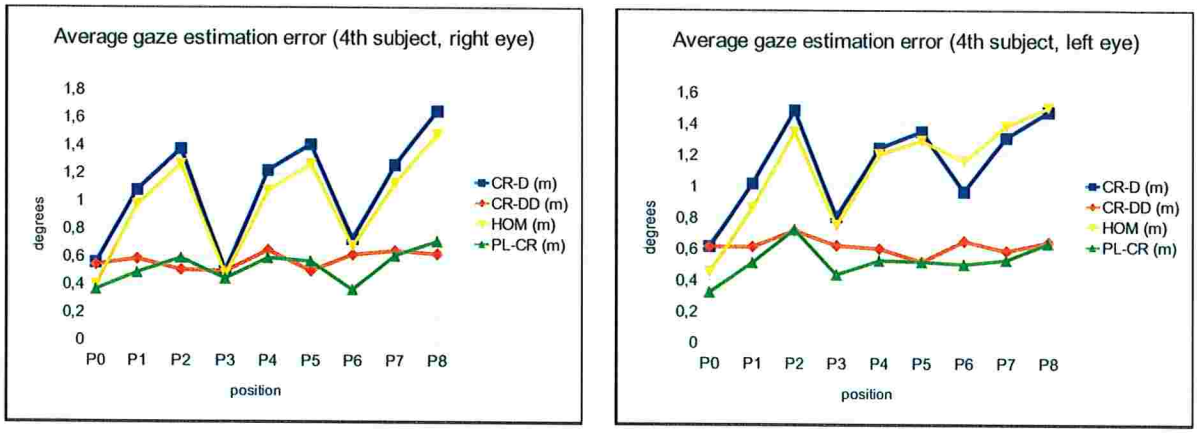

Figure A.42: *Average gaze estimation error (measured in degrees of visual angle), for left and right eyes of 4th subject, at test positions P_0 through P_8. The right eye is the dominant eye for this subject.*

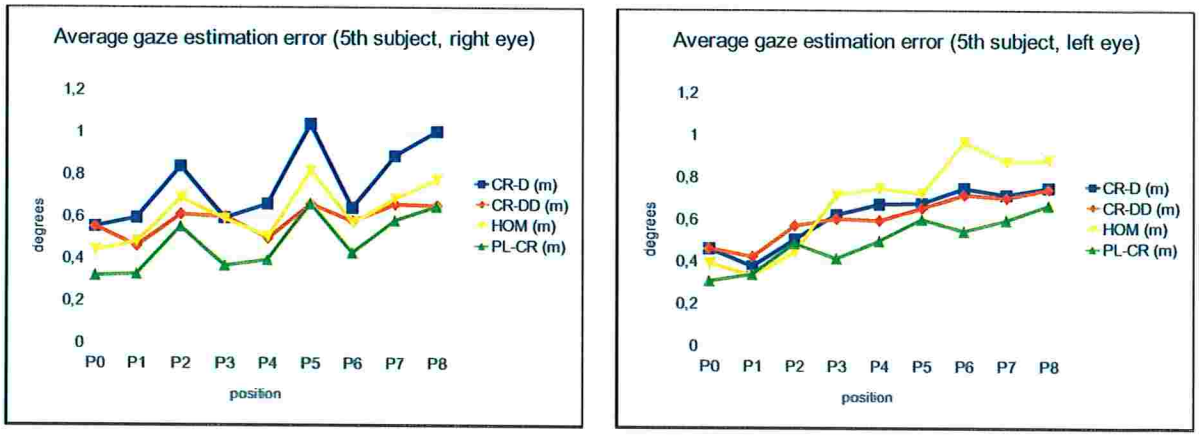

Figure A.43: *Average gaze estimation error (measured in degrees of visual angle), for left and right eyes of 5th subject, at test positions P_0 through P_8. The right eye is the dominant eye for this subject.*

calibration of the **CR-D** method is relatively small compared to the size of the right eye and other subjects as can be seen in Figure A.46. This indicates a smaller κ, as predicted in section A.4.1.

For subject 7's right eye, HMC methods do not achieve a smaller error than non-HMC methods at some positions. For the left eye, however, the result meets the expectation, showing a clear distinction between the two groups of methods.

For subject 3, a clear distinction in performance can not be observed between HMC and non-HMC methods. Also gaze estimation error of non-HMC methods are not affected by head movements as expected. For example, for this subject's right eye, the gaze estimation error observed at positions P_0 and P_8 are very similar, despite the fact of position P_8 being the farthest from P_0. As for the left eye, inspection of the displacement vector length also

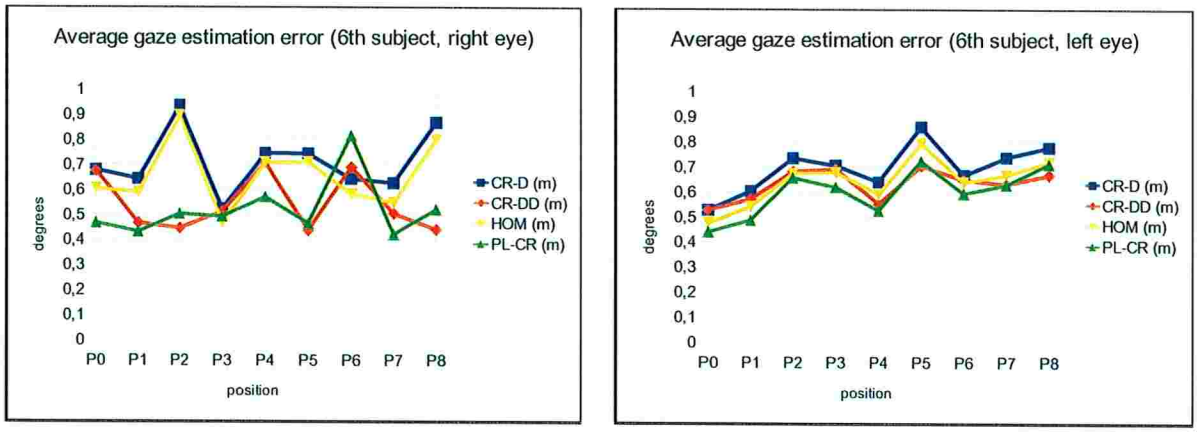

Figure A.44: *Average gaze estimation error (measured in degrees of visual angle), for left and right eyes of 6th subject, at test positions P_0 through P_8. The left eye is the dominant eye for this subject.*

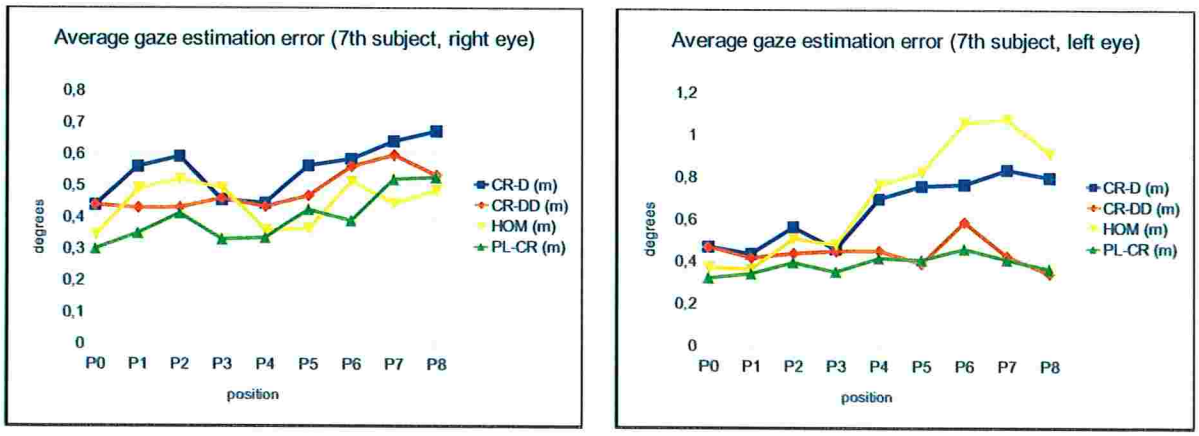

Figure A.45: *Average gaze estimation error (measured in degrees of visual angle), for left and right eyes of 7th subject, at test positions P_0 through P_8. The right eye is the dominant eye for this subject.*

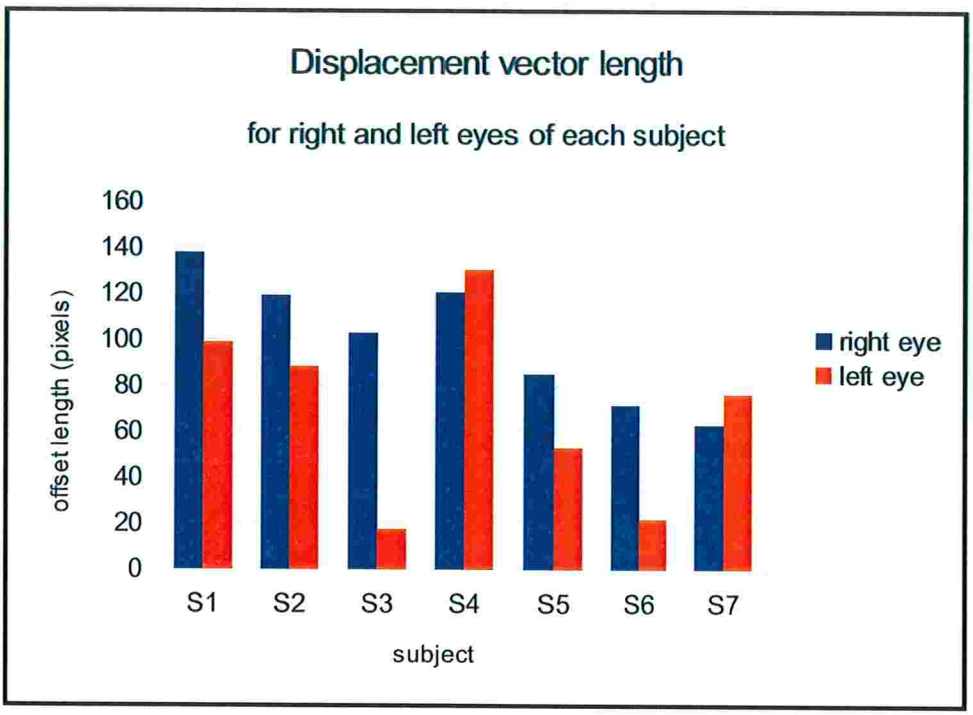

Figure A.46: *Size of displacement vector used by methods CR-D and CR-DD, obtained after calibration at* P_0.

suggests a small κ as the reason why the results are not the expected.

Tables A.1, A.2, A.3, and A.4 shows the calibrated parameters (v_x, v_y, c_z) obtained by calibration of the **PL-RC** method at each head position. When comparing the calibrated parameter sets with the parameter set for P_0, it is possible to observe some variations in their values. In theory, the model used for the **PL-CR** method is invariant to head motion, and as such, the calibrated parameters set (v_x, v_y, c_z) should be the same regardless of the head position. The variation that is observed, may explain some of the cases where performance of the **PL-CR** methods is not the expected and may also indicate that some subjects may adapt more to the eye model used than others.

The combined result for all subjects is shown in Figure A.47. At each position the average visual angle error (in degrees) for all subjects is presented. Results show that, on average, HMC methods achieve lower error than non-HMC methods. Also, error variation between all positions is smaller for the HMC methods. Average error ranges from 0.49^o to 0.62^o for the **CR-DD** method, 0.38^o to 0.59^o for the **PL-CR** method, 0.56^o to 1.01^o for the **CR-D** method and 0.44^o to 0.93^o for the **HOM** method. Table A.7.7 summarizes the percentage improvement of each HMC method compared to both non-HMC methods. As expected, we

v_x (right eye)							
	S1	S2	S3	S4	S5	S6	S7
P0	31.75	28.43	28.17	15.75	19.93	6.84	15.68
P1	31.52	28.13	27.19	15.74	19.26	5.89	16.63
P2	30.71	26.78	28.05	15.16	18.09	6.95	15.38
P3	32.62	28.1	24.89	14.84	17.85	6.33	13.66
P4	31.8	28.87	24.8	18.06	19.16	6.52	15.36
P5	34.09	27.53	23.49	16.17	20.1	4.94	16.06
P6	33.01	28.44	25.46	14.51	19.94	8.24	15.53
P7	33.79	26.1	24.88	15.79	19.68	6.93	16.77
P8	34.69	25.59	24.44	16.16	19.1	5.03	15.89
average	32.66	27.55	25.71	15.8	19.23	6.41	15.66
stddev	1.33	1.15	1.68	1.02	0.8	1.03	0.91
v_y (right eye)							
	S1	S2	S3	S4	S5	S6	S7
P0	-13.32	-6.96	0.49	-23.61	-5.3	-13.8	3.04
P1	-10.8	-8.58	-3.67	-28.1	-6.67	-17.6	1.67
P2	-15.4	-11.7	-4.79	-30.72	-10.56	-19.65	0.36
P3	-11.16	-4.88	-3.83	-20.87	-5.31	-14.89	5.52
P4	-13.93	-8.56	-2.63	-30.54	-9.61	-13.48	1.65
P5	-14.91	-12.31	-3.69	-31.14	-12.7	-17.81	-0.41
P6	-12.08	-12.09	-4.59	-27.4	-9.72	-13.99	1.55
P7	-12.61	-11.28	-0.93	-30.73	-13.13	-16.15	-0.57
P8	-14.61	-13.13	-2.53	-30.91	-12.62	-20.51	-2.06
average	-13.2	-9.94	-2.91	-28.22	-9.51	-16.43	1.19
stddev	1.65	2.82	1.73	3.7	3.11	2.61	2.23

Table A.1: *Estimated parameters v_x and v_y for the model used by the PL-CR method. This table shows the parameters for the right eye of all subjects (S_1 trough S_7) obtained by calibration at each position (P_0 trough P_8) and also their average and standard deviation. Values expressed in units of cornea radius/1000.*

c_z (right eye)							
	S1	S2	S3	S4	S5	S6	S7
P0	601.9	598.3	612.68	585.73	573.58	539.01	603.9
P1	600.2	593.06	602.84	581.03	570.94	531.64	605.36
P2	604.07	591.3	598.22	580.07	573.44	527.33	601.26
P3	597.98	601.12	601.43	586.51	575.8	534.12	606.51
P4	592.73	592.23	601.73	580.52	568.85	537.4	600.83
P5	586.35	586.66	600.84	577.72	565.09	524.84	601.67
P6	601.44	587.33	605.33	577.16	567.04	525.69	608.14
P7	591.25	590.21	605.02	579.59	562.64	526.27	608.35
P8	589.32	592.62	603.06	581.54	566.4	512.91	597.86
average	596.14	592.54	603.46	581.1	569.31	528.8	603.76
stddev	6.34	4.69	4.08	3.19	4.42	7.92	3.61

Table A.2: *Estimated parameter c_z for the model used by the PL-CR method. This table shows the parameters for the right eye of all subjects (S_1 trough S_7) obtained by calibration at each position (P_0 trough P_8) and also their average and standard deviation. Values expressed in units of cornea radius/1000.*

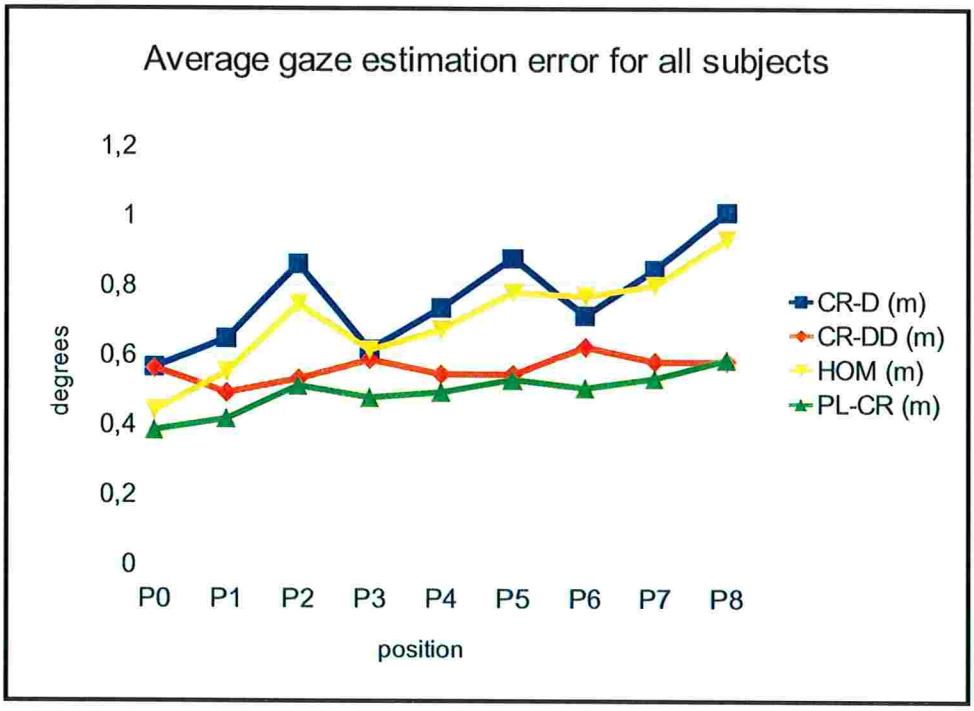

Figure A.47: *Average gaze estimation for all users, at test positions P_0 through P_8.*

observed more improvements for eye positions that reflect larger head movements.

A.7.8 Discussion of experimental results

Considering the average results for all subjects for each experiments, we observed that HMC methods achieved lower gaze estimation error when compared to the non-HMC meth-

v_x (left eye)							
	S1	S2	S3	S4	S5	S6	S7
P0	-22.29	-17.63	-3.55	-34.76	-11.41	3.18	-19.7
P1	-20.93	-16.34	-0.94	-36.41	-10.66	2.15	-18.18
P2	-22.4	-14.56	0.59	-36.18	-12.29	1.43	-19.01
P3	-23.69	-17.58	-4.48	-34.44	-13	5.02	-19.69
P4	-22.67	-15.25	-2.38	-36.55	-14.08	4.43	-21
P5	-21.21	-13.28	-1.46	-36.63	-13.41	1.89	-20.63
P6	-24.8	-17.59	-4.02	-37.14	-14.64	2.61	-22.32
P7	-23.2	-16.36	0.22	-37.32	-13.69	2.76	-22.01
P8	-24.75	-15.9	-3.38	-34.68	-14.84	0.23	-20.32
average	-22.88	-16.06	-2.15	-36.01	-13.11	2.63	-20.32
stddev	1.38	1.5	1.85	1.1	1.43	1.47	1.34
v_y (left eye)							
	S1	S2	S3	S4	S5	S6	S7
P0	-11.2	-11.93	4.82	-5.29	9.49	-2.04	2.54
P1	-11.84	-15.55	5.91	-9.67	6.01	-3.33	1.53
P2	-14.78	-15.21	2.99	-13.91	5.35	-6.21	1
P3	-10.19	-8.3	6.81	-6.38	9.52	-5.75	0.66
P4	-13.66	-12.66	1.65	-10.52	6.26	-5.98	-0.34
P5	-16.88	-13.3	2.46	-11.93	3.31	-8.62	-0.69
P6	-13.73	-14.77	5.97	-3.55	4.73	-6.18	-0.83
P7	-15.51	-15.97	6.48	-11.7	3.62	-8.38	-0.36
P8	-18.6	-17.33	3.84	-12.47	2.64	-9	0.25
average	-14.04	-13.89	4.55	-9.49	5.66	-6.17	0.42
stddev	2.73	2.7	1.89	3.59	2.5	2.35	1.12

Table A.3: *Estimated v_x and v_y parameters for the model used by the **PL-CR** method. This table shows the parameters for the left eye of all subjects (S_1 trough S_7) obtained by calibration at each position (P_0 trough P_8) and also their average and standard deviation. Values expressed in units of cornea radius/1000.*

c_z (left eye)							
	S1	S2	S3	S4	S5	S6	S7
P0	605.68	592.84	606.89	579.9	572.68	532.59	609.1
P1	602.02	591.75	612.21	570.56	567.21	527.82	610.11
P2	604.79	592.37	607.98	572.42	566.83	529.2	615.04
P3	602.04	595.75	603.63	577.53	578.63	530.31	606.47
P4	596.69	594.2	590.72	574.26	568.96	531.8	608.68
P5	599.8	589.6	592.71	565.41	564.61	524.06	608.56
P6	596.45	588.79	604.11	571.56	566.87	530.05	607.61
P7	596.17	589.43	603.76	570.23	566.06	520.67	604.09
P8	584.43	586.63	602.16	566.99	564.12	517.45	603.35
average	598.67	591.26	602.69	572.1	568.44	527.1	608.11
stddev	6.41	2.88	6.92	4.64	4.58	5.24	3.45

Table A.4: *Estimated parameter c_z for the model used by the **PL-CR** method. This table shows the parameters for the left eye of all subjects (S_1 trough S_7) obtained by calibration at each position (P_0 trough P_8) and also their average and standard deviation. Values expressed in units of cornea radius/1000.*

	CR-DD/CR-D	CR-DD/HOM	PL-CR/CR-D	PL-CR/HOM
P0	0.65%	-27.23%	32.04%	12.96%
P1	24.14%	10.87%	35.66%	24.40%
P2	38.31%	28.75%	40.59%	31.39%
P3	4.61%	4.45%	22.41%	22.28%
P4	25.95%	18.96%	32.94%	26.61%
P5	37.95%	30.09%	39.76%	32.13%
P6	12.63%	19.01%	29.12%	34.30%
P7	31.35%	27.42%	37.02%	33.42%
P8	42.55%	37.69%	42.01%	37.11%

Table A.5: *For each position, the average percentage of error reduction for all users when CR-DD and PL-CR methods are compared to the CR-D and HOM.*

ods. Despite some differences in the results obtained for both experiments the relative performance between the compared methods were similar for both.

A comparison between the non-HMC methods shows a better performance of the **HOM** method over the **CR-D**. This result is again expected since the **CR-D** employs just two basic transformations: a scale (of p, the observed pupil in the image) and a translation (addition of the displacement vector). The displacement vector is obtained so that gaze estimation error over the entire screen is minimized on average, but is not optimal for individual screen points. The **HOM** method, in contrast, uses a homography normalization to compensate sources of errors of the cross-ratio method. The homography is more flexible (has more degrees of freedom) and as a result the correction is better for different regions on the screen, meaning lower gaze estimation error for each screen point and overall lower average error than the **CR-D** method.

As for the HMC methods, the **PL-CR** performs better than the **CR-DD** mainly for positions where the head is closer to the screen. As the head gets farther from the screen the performance difference between these two methods gets smaller until the observed gaze estimation errors are almost the same. This is explained by the **CR-DD** method's approach to compensate head movements that just considers distance variation, ignoring eye rotation. Because of this, at distances closer to the screen (where the eye needs to rotate more), the head movement compensation is not as affective as the one employed by the **PL-CR** method.

One reason that explains variation of results between different subjects is relative to κ. When a subject's κ is small, the effects of head movements for the methods that do not explicitly compensates them are smaller and consequently the new proposed methods will not show much accuracy improvement. The magnitude of κ can be inferred by the size displacement vector used by the **CR-D** and **CR-DD** methods. Figure A.46 shows the sizes of the displacement vectors obtained by calibration of the **CR-D** method, for all subjects of the second experiment. It is possible to see that the left eyes of subjects 3 and 6 have the smallest κ and some of the lowest improvements ratio for the **CR-DD** and **PL-CR** methods. A small κ is also the case of subject 8 in the first experiment.

Comparing results of experiments 1 and experiment 2, we can notice that gaze estimation

accuracy for the second experiment is better. This is a result of the experiment redesign that reduced potential sources of errors. For the second experiment the IR LEDs were better aligned to the screen plane, and measures were taken to increase reliability of captured data. Also, the position P_0 used in experiment 1 was too close to the screen, affecting corneal reflection formation.

Regarding the HMC methods, since they rely on extra computation to compensate head movements, inaccuracies of detected feature points will be propagated to the head compensation schemes used by each. An important note about the **CR-DD** method in particular is that the method considers distance variation relative to the screen, but in fact what is being computed is distance variation relative to the camera. In the experiments the camera was positioned about 10-15 cm in front of the screen. This difference may introduce some error in the estimation of distance variation. Potential sources of error for the **PL-CR** includes eventual discrepancies between the model used and subject's eyes. It is known for example that the radius of the cornea surface changes from its central region to its borders [NIK+10].

Eye torsion was also ignored for both HMC methods. Although Guestrin and Eizenman [GE10] argued that practical effect of this kind of eye movement (considering distances of 60-70 cm from the screen) in gaze estimation results are very small, it would be interesting to further investigate and consider this kind of movement in the model in a future work.

A.8 Implementation

Our implementation was constraint to the hardware that we had available in our lab. Our gaze tracking device consists of an analog black and white video camera, a USB video capture card, a desktop computer, and several light sources that are required by the cross-ratio methods.

Light sources consists of infrared LEDs and are divided into two sets. One set correspond to the L_i points and are attached around the monitor. They generate corneal reflections G_i that are imaged as g_i. The second set is attached around the camera's optical axis. This set generate the reference corneal reflection G_0 imaged as g_0. A filter is also used in the camera to filter light in the visible spectrum.

A circuit that process the analog video signal controls activation of these two sets of LEDs. While the even field is scanned, the first set of LEDs (screen) is activated. Besides generation of g_i, the pupil appears dark in the even lines of the image (as we usually see it). While the odd is field scanned, the second set of LEDs (camera) is activated. Besides generation of g_0, light is reflected from the back of eye, and the pupil appears bright at the odd lines of the image.

Since the pupil is usually the only image element that exhibit a large contrast between the two illumination conditions, this alternating illumination scheme facilitates pupil detection during image processing.

A.8.1 Software

The gaze tracking software was developed for the Linux platform and uses the OpenCV library for image processing. It works in both real time and offline modes, and implements the **PCR**, **CR**, **CR-D**, **CR-DD**, **HOM**, and **PL-CR** methods for remote eye gaze tracking. The gaze tracking software is responsible for video acquisition (when operating in real time mode), image processing, and detection of eye features which are then passed to the implementations of the gaze estimation methods. Image processing and feature detection mainly consists of pupil and corneal reflection detection. For the **PL-CR** method, in particular, iris detection is also performed.

Pupil detection is based on the differential method [Ebi95]. The first step consists in deinterlacing of the input image, producing a *bright* and a *dark* pupil image, followed by subtraction of the *dark* image from the *bright* one. The resulting image, $diff$, is then thresholded to segment its high contrast regions, resulting in $diffT$. To avoid considering very bright regions (corneal reflections or specular reflections over glasses) as pupil candidates, or as part of the pupil, the brightest areas of both *dark* and *bright* images are segmented by two more threshold operations, resulting in $darkB$ and $brightB$. To make pupil detection more robust, a threshold is also applied to the *dark* image to select its darkest regions, resulting in $darkT$. A binary image containing regions that are pupil candidates (i.e., present high contrast between *dark* and *bright* images, appear as dark regions in the *dark* image and are not extremely bright in any of the *dark* and *bright* images) is given as the result of following boolean operations:

$$candidate = diffT \wedge darkT \wedge (!darkB) \wedge (!brightB) \tag{A.57}$$

After the *candidate* image is computed, connected component regions are extracted and analyzed to select one as the best pupil candidate. Ideally we expect to have just one candidate blob but, in some situations, especially for people wearing glasses, it is possible to have more than one. The best candidate is selected as largest blob that satisfies some conditions: the aspect ratio of the bounding box around the blob must be $>= 0.5$ and $<= 2.0$, and also the fill ratio (area of the blob divided by the area of the bounding rectangle) must be > 0.5 (just for comparison the fill rate of a circular shape is approximately 0.785).

After the best candidate is selected, its contour is extracted into the *contour* image. Care must be taken here, because if a corneal reflection is formed over the pupil edge, the pupil contour will be corrupted by part of the reflection contour. We eliminate this interference by dilating $darkB$ and $brightB$ and subtracting both of them from the *contour* image. The resulting pixels in the contour are then used to fit an ellipse, that is taken as the pupil, and p is taken as the ellipse center.

Since corneal reflections appear as bright spots in the images of the eye, we detect them by segmenting bright regions in both the *dark* and *bright* images. In fact this step is executed

for pupil detection, resulting in $darkB$ and $brightB$. The remaining blobs in both $darkB$ and $brightB$ are ordered according to their distances from the pupil center previously computed. In $brightB$ we expect to find one corneal reflection, so the closest blob to the pupil center is taken as g_0. For the $darkB$ image, the 5 closest blobs to the pupil center are selected and combinations of 4 blobs are tested against a rectangularity criteria. Given 4 points that form a quadrilateral, and its internal angles $\hat{a}_i$, the rectangularity is the sum of $|\hat{a}_i - 90|$, for $i \in \{1, 2, 3, 4\}$. The smaller this sum is, the closest the quadrilateral is to a rectangle. The combination that has the smaller rectangularity score is taken as the set of corneal reflections g_i.

For the **PL-CR** method, in addition to the pupil and corneal reflections, the iris must also be detected. Instead of actually detecting the iris contour, we compute an approximation of it based on the detected pupil. It is assumed that the iris is imaged as an ellipse with the same center and shape of the pupil. The iris approximation is computed by scaling the detected pupil so that the resulting ellipse's contour best matches the contour of the actual iris.

The computation of the approximate iris based on the scale of the pupil eliminates the need to detect the full iris contour, but at least one point from the contour must be detected in order to determine by how much the pupil needs to be scaled. To increase the chance of successful detection of such contour point, we look for it in the horizontal line passing through p. This search starts at p and follows in the direction towards the corneal reflection g_0. This strategy ensures that the iris contour point belonging to the horizontal search line will not be occluded by eyelids or eye corners.

The gradient vector $\vec{gr}$ of an arbitrary point ic that belongs to the iris contour is expected to have a large magnitude value and to also point in the approximate direction given by $\vec{i} = (ic - p)$. If ic belongs to the same horizontal line as p, $\vec{i}$ can be expressed by $\vec{i} = (1, 0)$ or $\vec{i} = (-1, 0)$ depending on the search direction. This way, the point in the horizontal search line that maximizes the following score

$$score = \left(\frac{\vec{gr}}{\|\vec{gr}\|} \cdot \frac{\vec{i}}{\|\vec{i}\|} \right) \|\vec{gr}\| \tag{A.58}$$

is taken as a point belonging to the iris contour, which will then be used to scale the pupil.

For the user experiments, in particular, a manual approach to iris detection was used. The manual approach for iris detection also assumes that the iris will have the same center and shape as the detected pupil. Based on the detected pupil on a given image, the user select an amount of scale so that the scaled pupil contour best matches the iris contour. A simple user interface is provided so that the user can evaluate the amount of scale needed. To avoid the manual selection of iris in thousands of images, detection of iris was just performed when subjects were looking at the mid point of the bottom row of test targets. The choice of this point is justified by the fact that in this condition iris are almost circular and a good estimate for its diameter can be measured (in image pixels) and stored. Keeping the same head position, when the eye rotates and the iris is viewed as an ellipse, its major axis length will be equivalent to the stored iris diameter. This way, when the eye gazes at arbitrary test points on the screen, the stored diameter for the respective head position can be used to scale the detected pupil so that the resulting ellipse's major axis has the same length as the stored value.

The use of an iris approximation in both the automatic and manual detection approaches is reasonable since we chose to compute the iris normal by the alternative method presented in section A.6.5 which does not rely on its contour but rather on image points p and g_0, and the c_z model parameter. The iris contour is, in fact, just used by the **PL-CR** method for the computation of the s scale factor as described in section A.6.3.

Our gaze tracker implementation is able to process each video frame in approximately 12 milliseconds using one core of a Xeon 2.8 GHz processor, which ensures the real time operation, an essential requirement when we have interactive applications in mind. This processing time was achieved for the **PL-CR** method, which is the one that demands more computation to estimate the gaze. This time also includes the time spent during the display of the captured video.

A.9 Conclusion

In this work we presented two new methods with the objective of improving head movement tolerance for cross-ratio based eye trackers: the **CR-DD** and the **PL-CR** methods.

The **CR-DD** method is an extension of the **CR-D** method in which the size of the displacement vector is adjusted dynamically according to the eye distance from the screen. Instead of absolutely computing the eye distance, we compute the eye distance variation relative to an initial eye position. This computation is done based on the observed size of the corneal reflection pattern. A problem with this approach is that what we are truly measuring is the distance variation from the camera, but we are taking it as the variation between the eye and screen. It may be acceptable depending on the placement of the camera, but this compensation is not 100% effective. Another problem is that we are just measuring variation in eye distance, but not eye rotation that also affects the displacement vector.

The **PL-CR** method compensates both sources of errors pointed by [GEKE08] by estimating the average Π_G plane where corneal G_i reflections are formed (or the plane over which L_i are projected, having the cornea center as projection center) and computing the intersection of the visual axis of the eye with this plane (V'). Once G_i and V' lies on a common plane and the true visual axis of the eye is being considered, the basic principle of the cross-ratio method can be applied directly. To keep the use of a single non-calibrated camera an orthographic camera model was used to compute Π_G and V'. We also used an eye model whose parameters (obtained via calibration) are invariant regardless of the eye location and orientation. In contrast to the **CR-DD** method, the approach taken for the **PL-CR** method better handles all kind of head movements since eye rotation is handled naturally by the computation of the visual axis intersection with Π_G.

These methods were evaluated and compared to the **CR-D** and **HOM** cross-ratio based methods. Both simulations and user experiments were conducted. Both simulation and experimental results confirmed the improvement in gaze estimation accuracy for the two proposed methods under the condition of head motion. We also showed that the amount of observed improvement is dependent on the magnitude of the angle between the visual and optical axis of the eye.

A.10 Future work

As a future work, we plan to develop a new gaze tracking framework that dynamically adapts the gaze estimation method according to the number of corneal reflections found on an eye image. The motivation for this future work is that we realized that it can be difficult to guarantee proper corneal reflection formation considering a large area of head movement. This way, it would be highly desirable if the tracker could be able to continue the gaze estimation under the condition of missing corneal reflections. To see what would be possible to compute with fewer reflections, an analysis of the homography transformation is given.

Let $p = (p_x, p_y, 1)$ and $q = (q_x, q_y, 1)$ two 2D points in homogeneous coordinates, and $\mathbf{H}$ be an homography transformation with the following form:

$$\mathbf{H} = \begin{pmatrix} h_1 & h_2 & h_3 \\ h_4 & h_5 & h_6 \\ h_7 & h_8 & 1 \end{pmatrix} \tag{A.59}$$

If $q = \mathbf{H}p$. Coordinates of q are given by:

$$\begin{aligned} q_x &= \frac{h_1 p_x + h_2 p_y + h_3}{h_7 p_x + h_8 p_y + 1} \\ q_y &= \frac{h_4 p_x + h_5 p_y + h_6}{h_7 p_x + h_8 p_y + 1} \end{aligned} \tag{A.60}$$

Rearranging the above equations we have:

$$\begin{aligned} q_x \left(h_7 p_x + h_8 p_y + 1\right) &= \left(h_1 p_x + h_2 p_y + h_3\right) \\ q_y \left(h_7 p_x + h_8 p_y + 1\right) &= \left(h_4 p_x + h_5 p_y + h_6\right) \end{aligned} \tag{A.61}$$

From (A.61) we can see that a pair of corresponding points $\{p, q\}$ defines two equations. Thus, to estimate an homography transformation, which has 8 parameters, at least four correspondences must be known. This is the reason why cross-ratio based methods requires

all four corneal reflections g_i to be properly detected.

In the scenario where just 3 reflections are detected, it is impossible to estimate a homography, but it is possible to estimate an affine transformation. An affine transformation is a particular case of the homography transformation where coefficients h_7 and h_8 (perspective coefficients) are set to zero. In this case, 3 corneal reflections define 6 equations that are enough to estimate the 6 coefficients of an affine transformation.

An alternative for when just 3 corneal reflections are detected is to continue using the planarization method since knowledge of 3 G_i points is enough to estimate the Π_G plane and compute the visual axis intersection with it. Moreover, given the geometry of the rectangle formed by L_i it would also be possible to estimate the missing reflection.

When just two corneal reflections are available, just 4 equations can be defined and just 4 coefficients can be estimated. This is enough to describe a similarity transformation:

$$\begin{pmatrix} s\,cos(\theta) & -s\,sin(\theta) & t_x \\ s\,sin(\theta) & s\,cos(\theta) & t_y \\ 0 & 0 & 1 \end{pmatrix} \tag{A.62}$$

where s represents a scale and θ a rotation.

Finally, when just one reflection is available, we can compensate for head translation and we have the situation of the **PCR** method. In this case a mapping estimated during the most recent successful detections of 2 or more reflections can be used to estimate gaze.

We expect that this adaptive approach should greatly improve gaze tracking robustness, also a very important characteristic for interactive applications. Besides development of such a gaze tracking device, future work would also include evaluation of gaze estimation accuracy for each kind of transformation.

Appendix B

Detailed experimental results

This appendix presents detailed results obtained for the user experiments 1 and 2. Results for user experiment 1 are presented in section B.1, while section B.2 shows results for the user experiment 2. Each figure in any section contains a set of heatmaps, where each heatmap corresponds to a head position. Also, each heatmap displays the distribution of the gaze estimation error across all test points of the screen. In each figure the set of heatmaps is organized following the corresponding layout of head positions used in each experiment. Due to the large number of figures, we opted to present them in this appendix separated from the main text.

B.1 Detailed results for experiment 1

In this section the detailed results for the experiment 1 are shown. Each figure displays the distribution of the gaze error in all positions for a given subject and gaze estimation method. Head positions for this experiment follows the layout of Figure A.13. In this experiment we had the participation of 9 subjects and just images of the subjects' right eye were used. Also, 4 gaze estimation methods were considered for the experiment: the **CR-D**, **CR-DD**, **HOM**, and **PL-CR** methods.

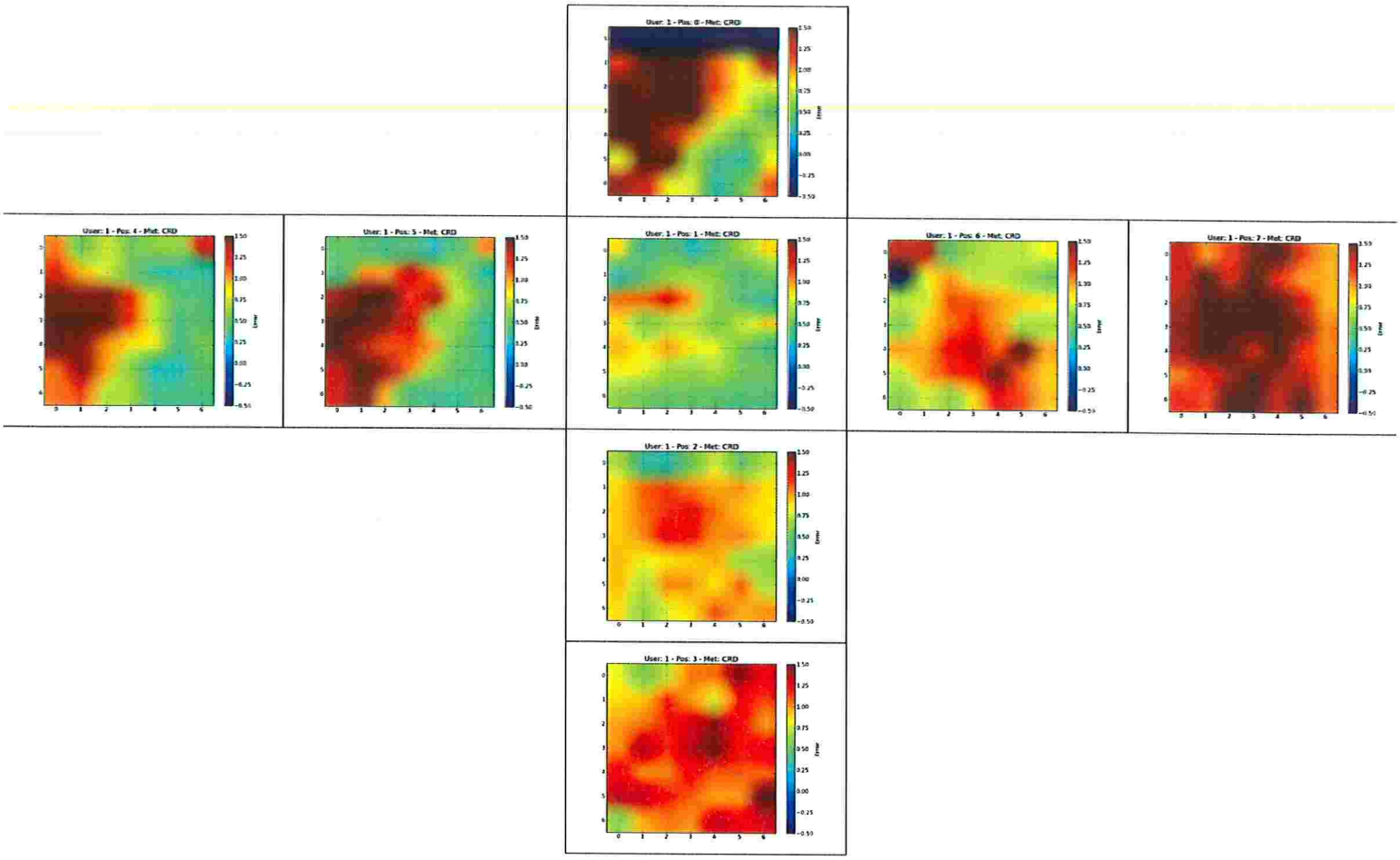

Figure B.1: *Distribution of the gaze error in all positions for participant 1 using method* ***CR-D****. Position 1 was used for calibration.*

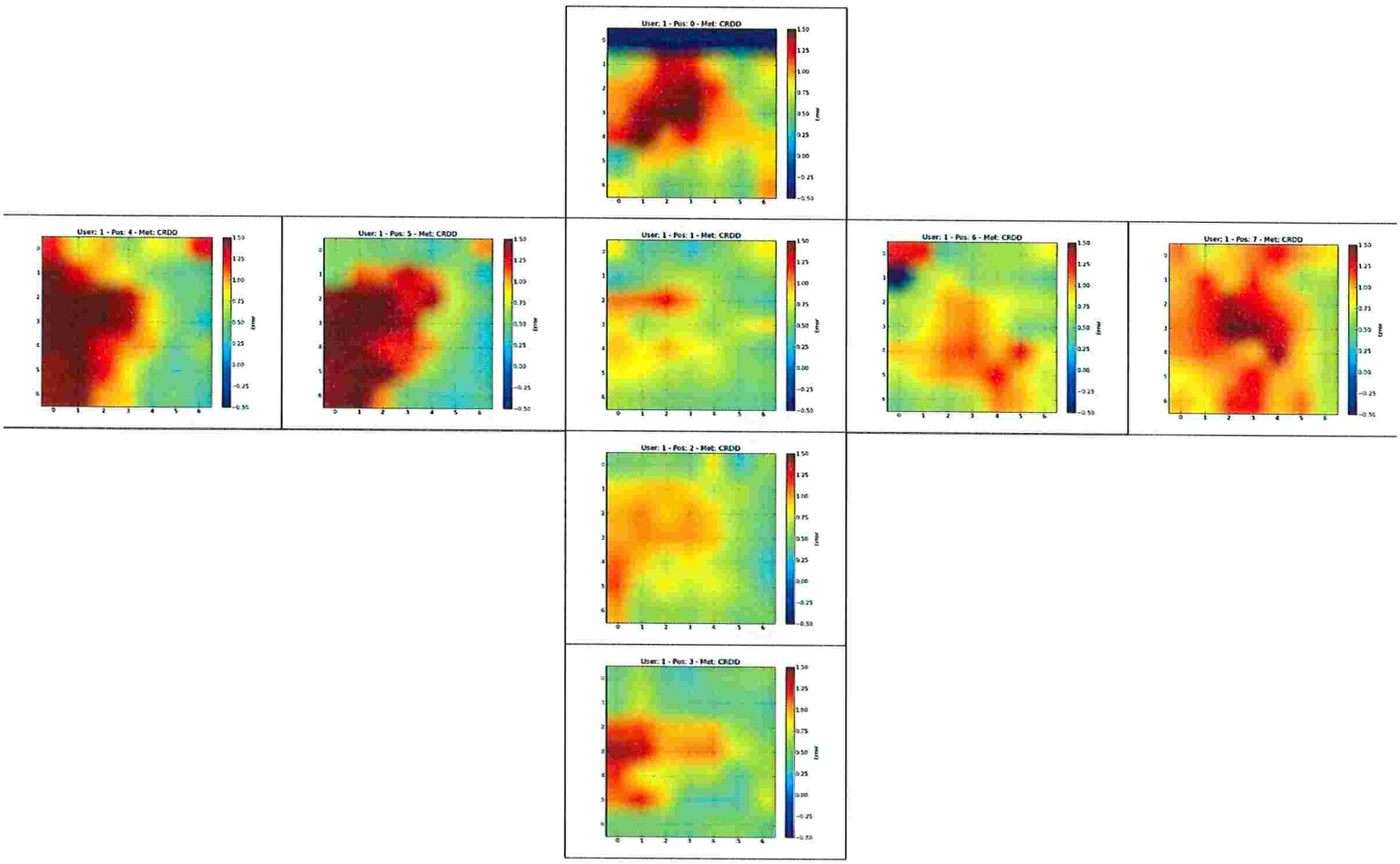

Figure B.2: *Distribution of the gaze error in all positions for participant 1 using method* ***CR-DD****. Position 1 was used for calibration.*

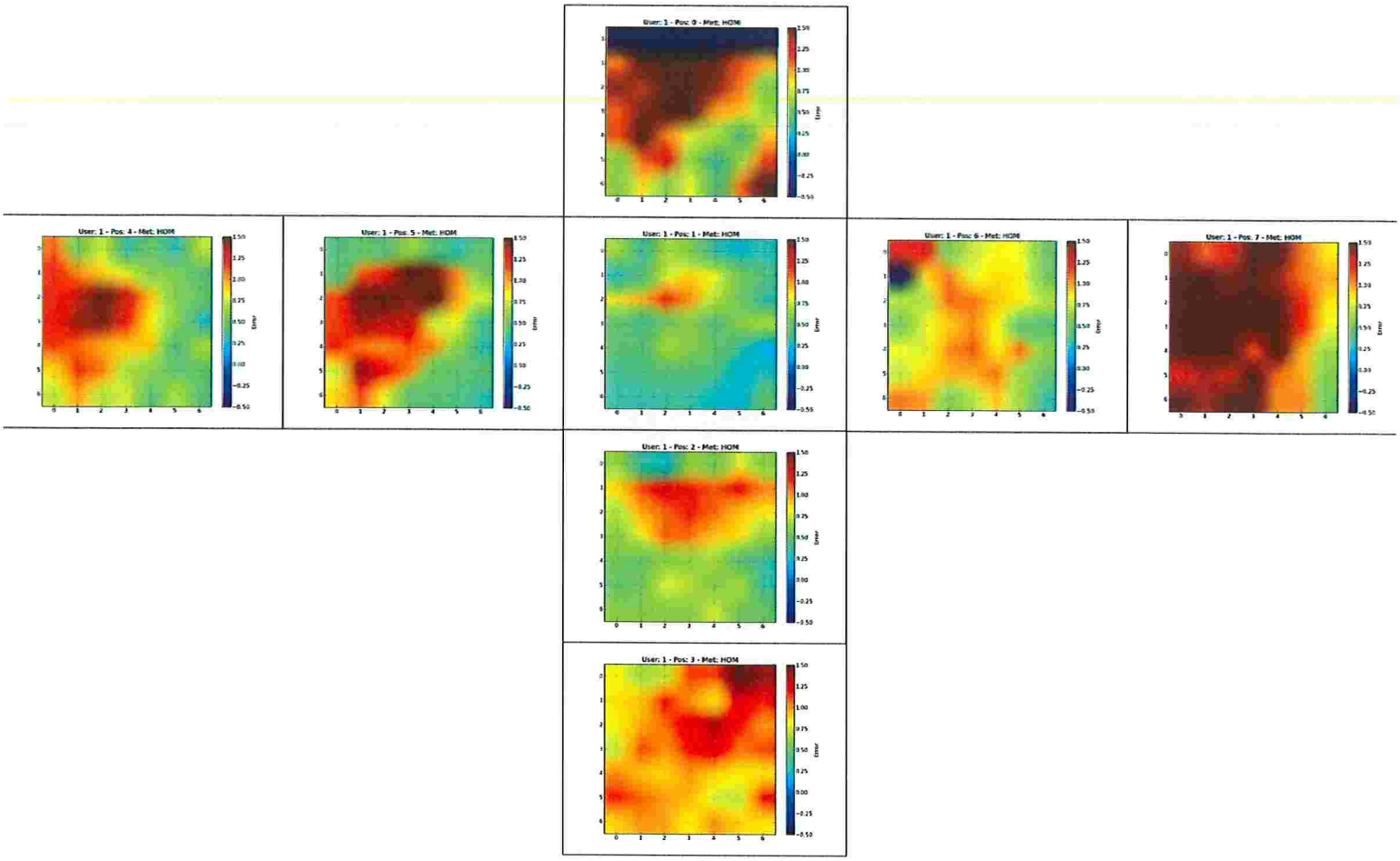

Figure B.3: *Distribution of the gaze error in all positions for participant 1 using method* ***HOM****. Position 1 was used for calibration.*

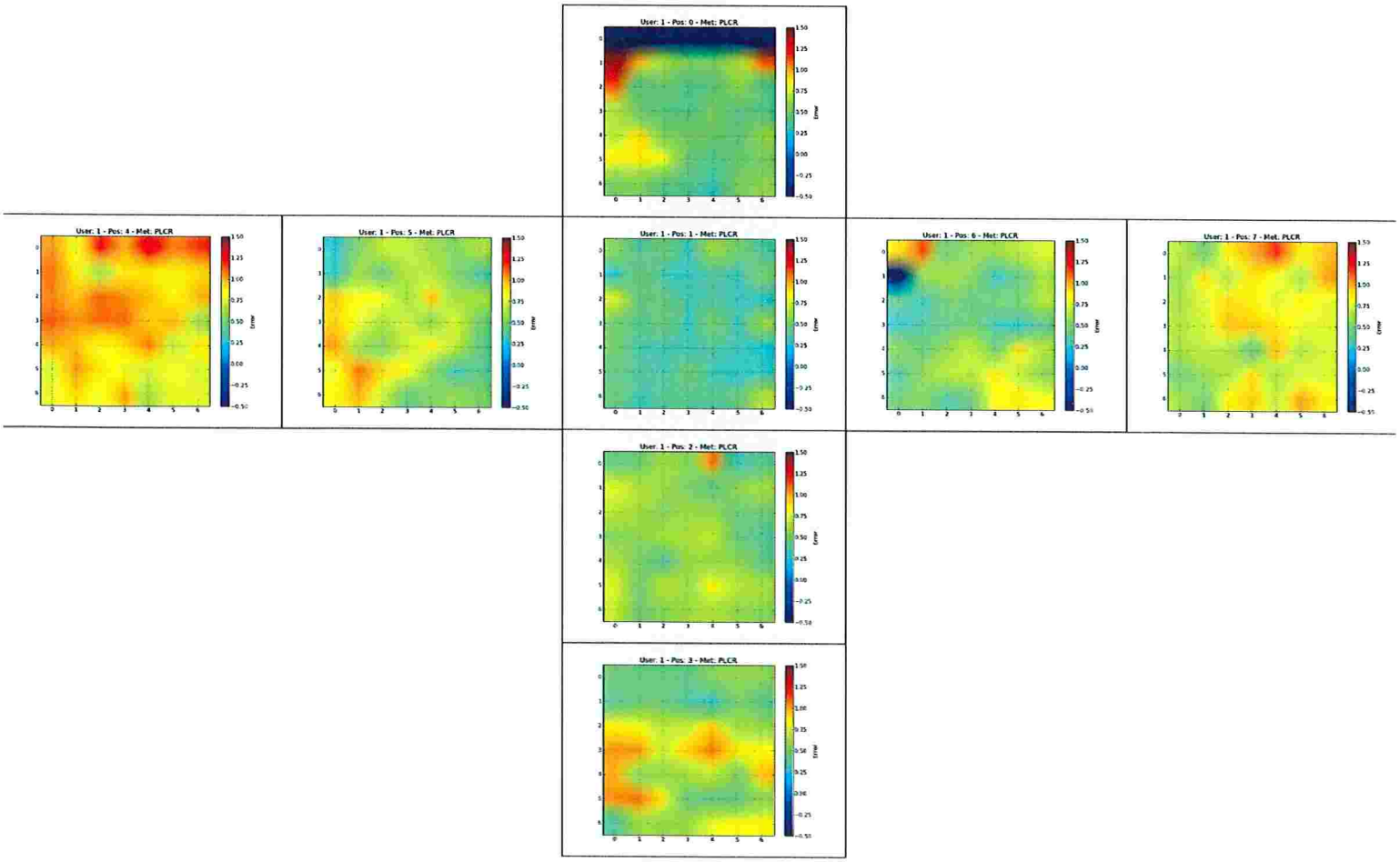

Figure B.4: *Distribution of the gaze error in all positions for participant 1 using method* ***PL-CR****. Position 1 was used for calibration.*

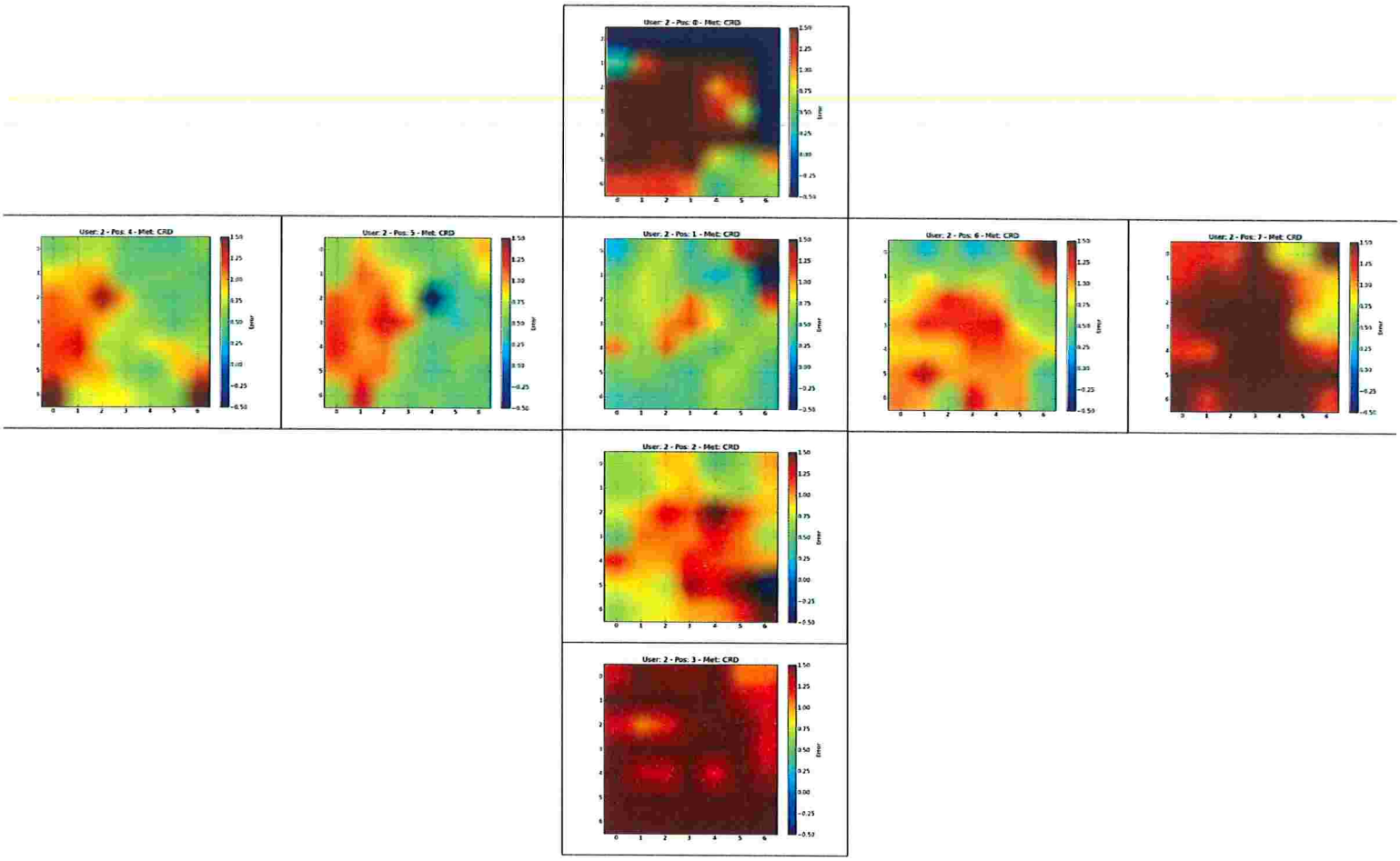

Figure B.5: *Distribution of the gaze error in all positions for participant 2 using method* ***CR-D****. Position 1 was used for calibration.*

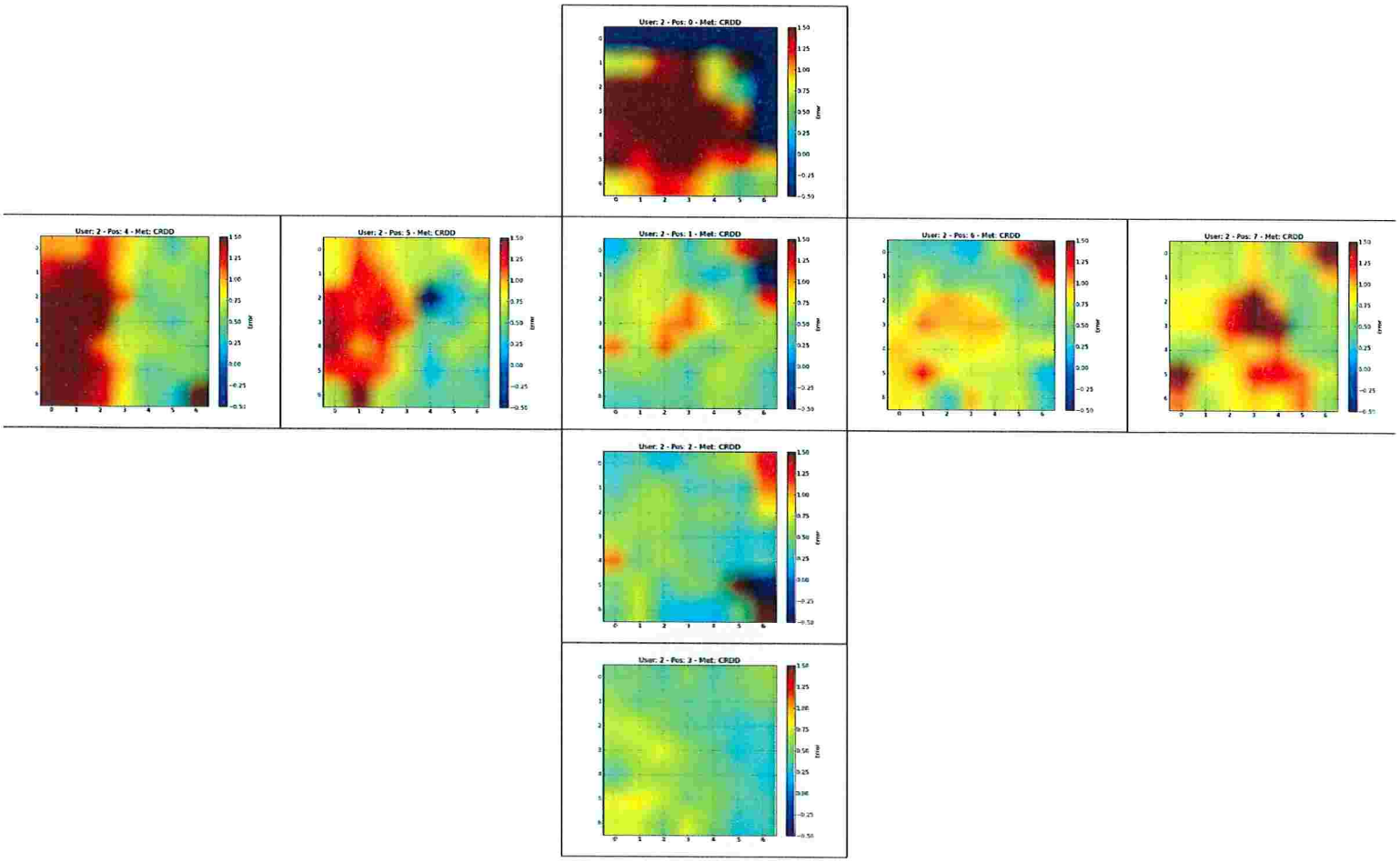

Figure B.6: *Distribution of the gaze error in all positions for participant 2 using method* ***CR-DD****. Position 1 was used for calibration.*

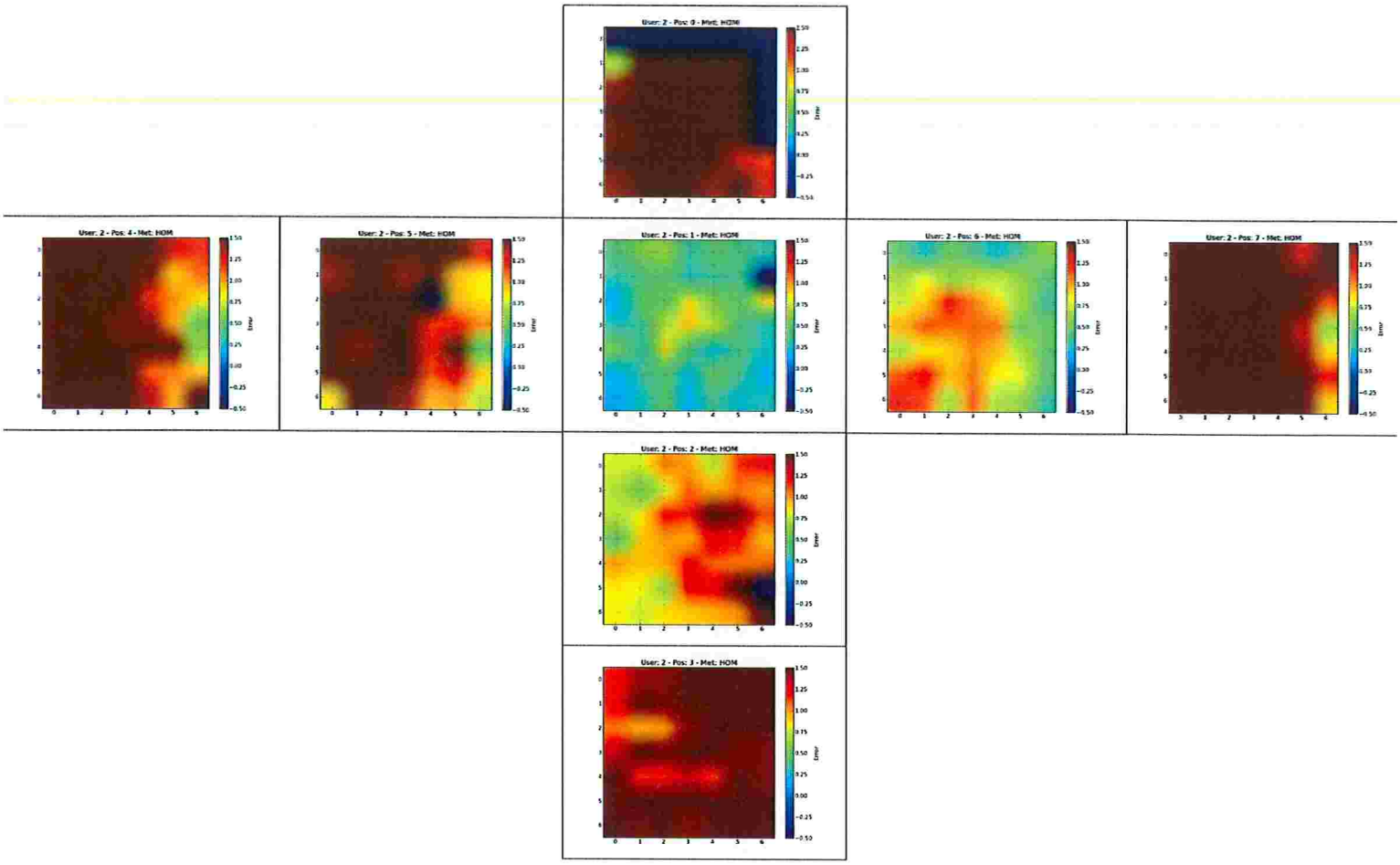

Figure B.7: *Distribution of the gaze error in all positions for participant 2 using method* ***HOM****. Position 1 was used for calibration.*

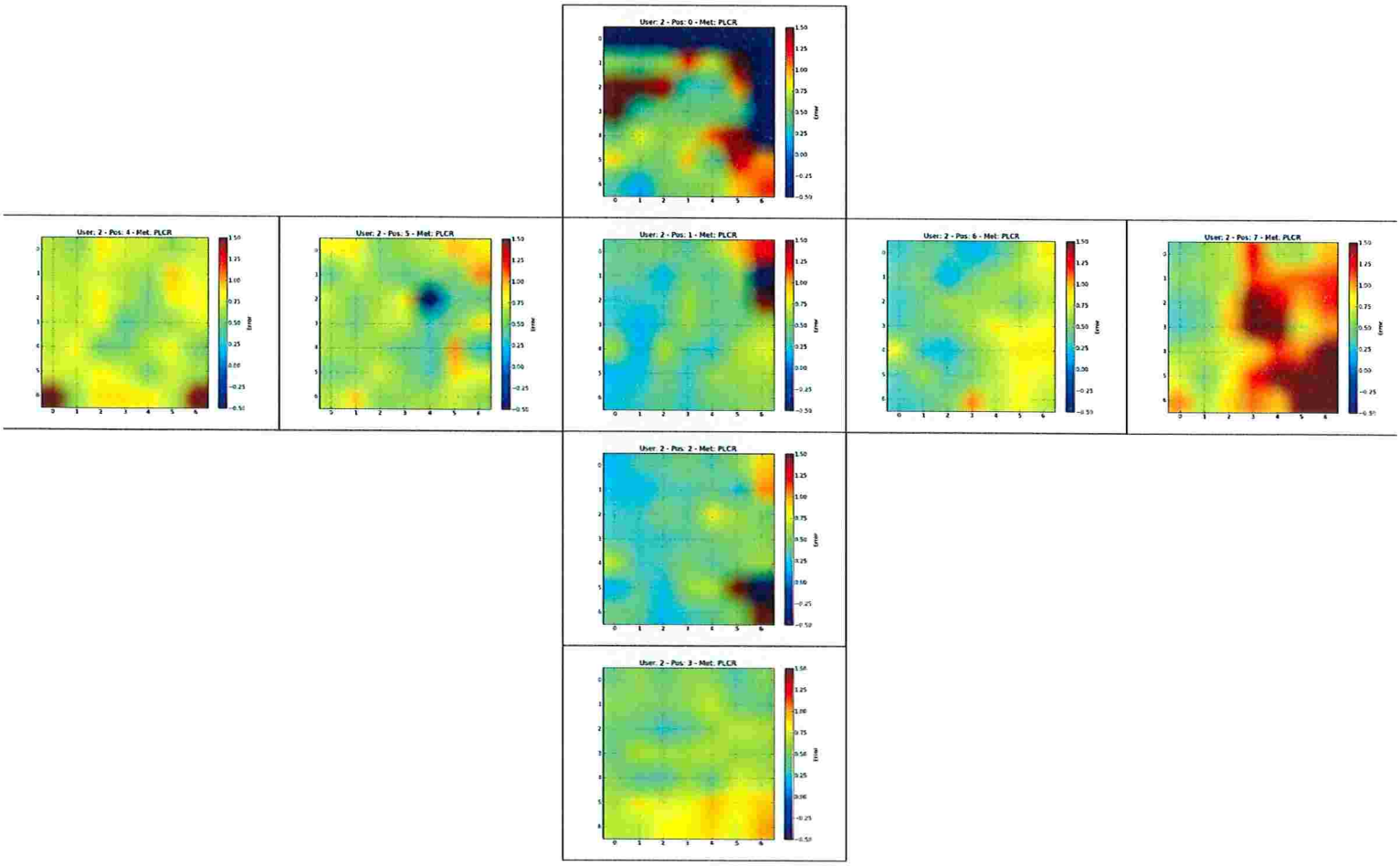

Figure B.8: *Distribution of the gaze error in all positions for participant 2 using method* ***PL-CR****. Position 1 was used for calibration.*

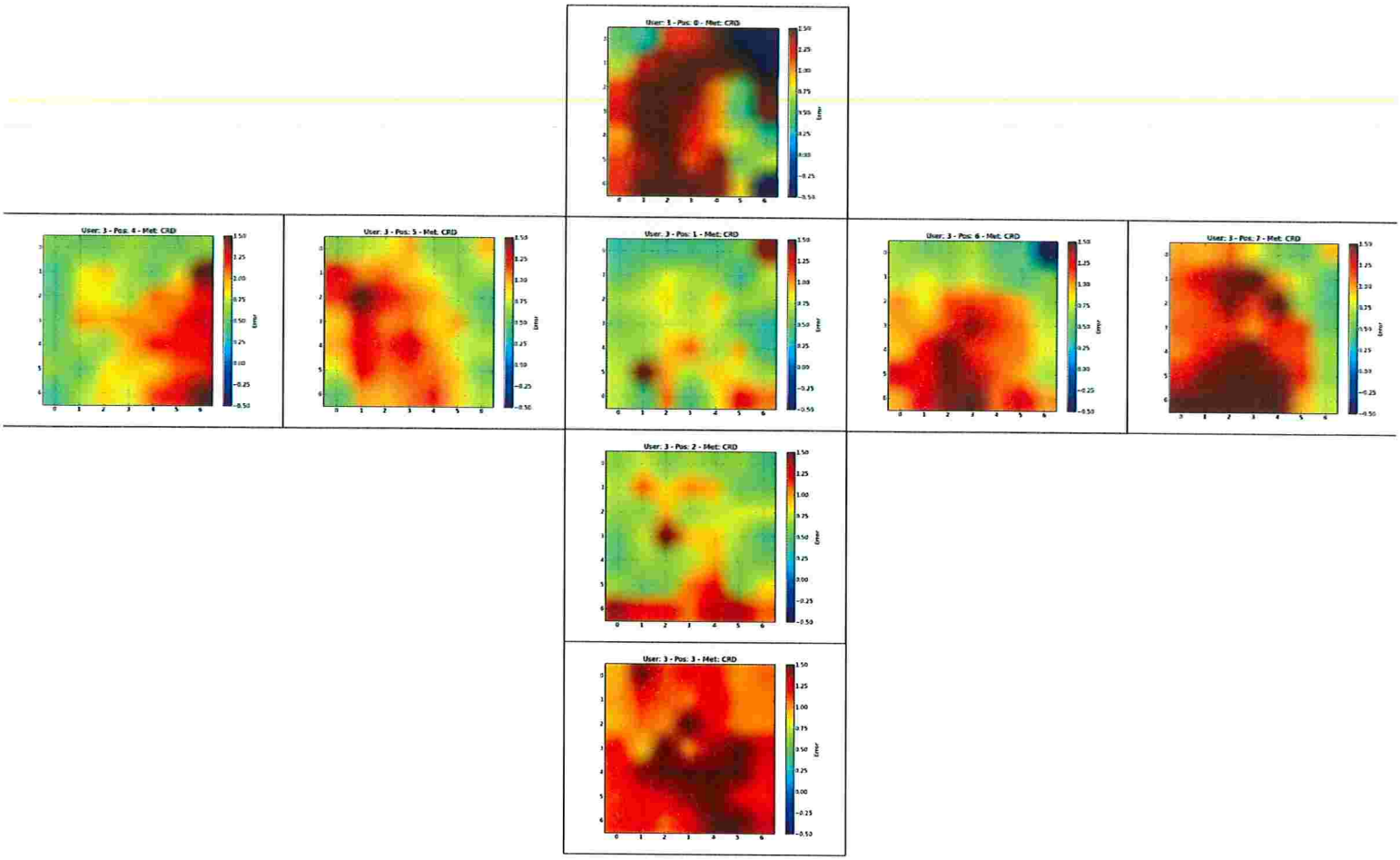

Figure B.9: *Distribution of the gaze error in all positions for participant 3 using method* ***CR-D****. Position 1 was used for calibration.*

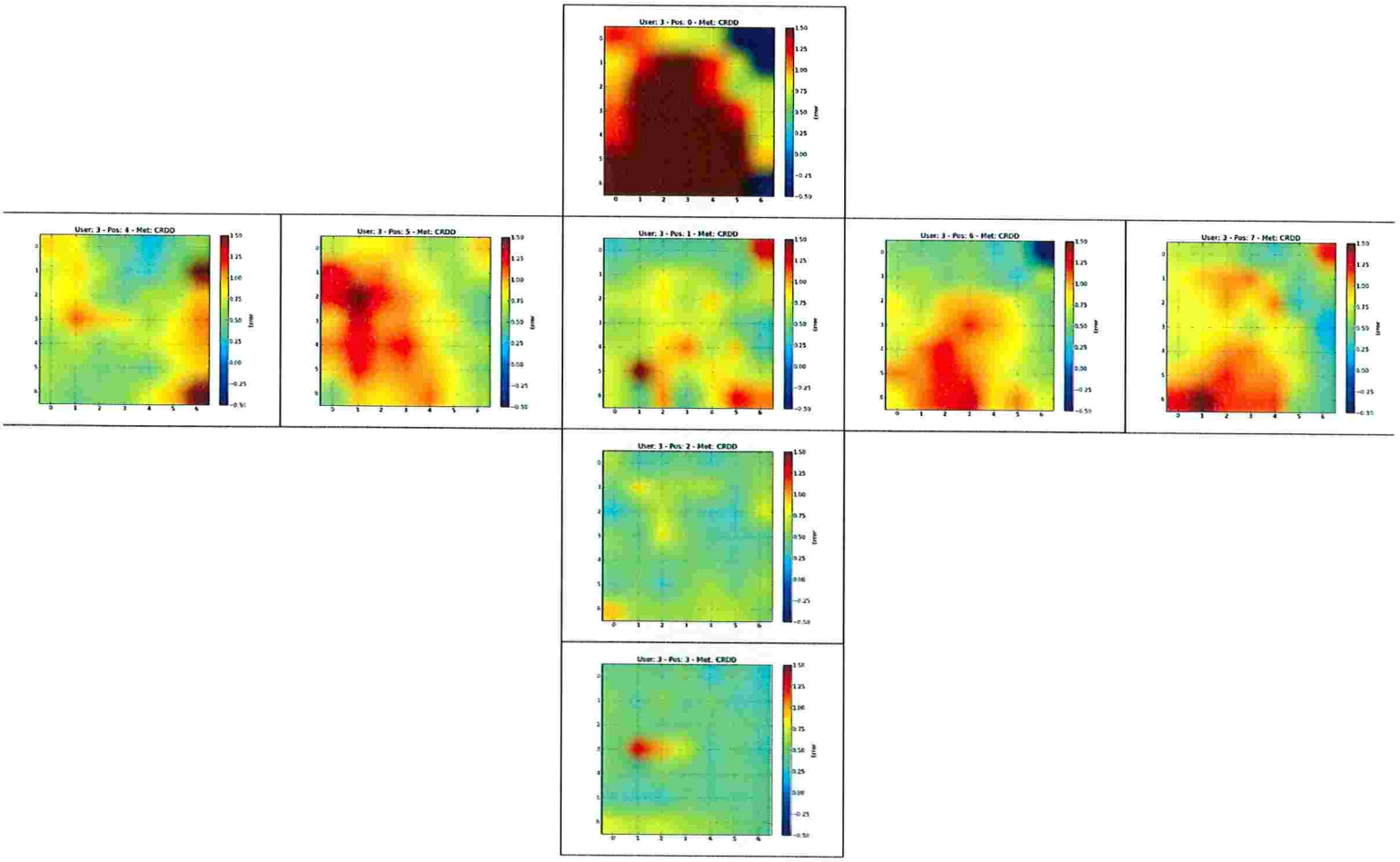

Figure B.10: *Distribution of the gaze error in all positions for participant 3 using method* ***CR-DD****. Position 1 was used for calibration.*

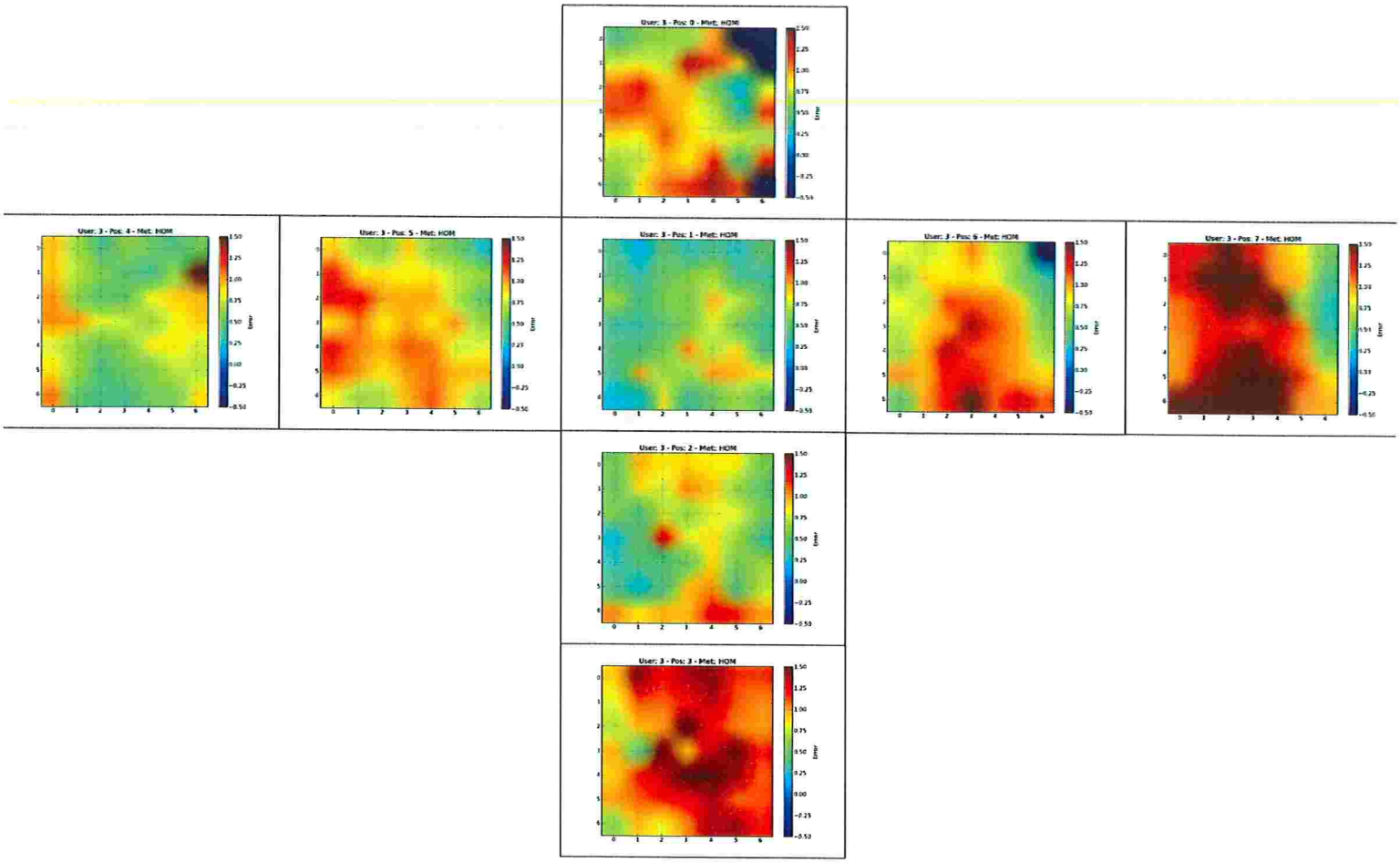

Figure B.11: *Distribution of the gaze error in all positions for participant 3 using method* ***HOM****. Position 1 was used for calibration.*

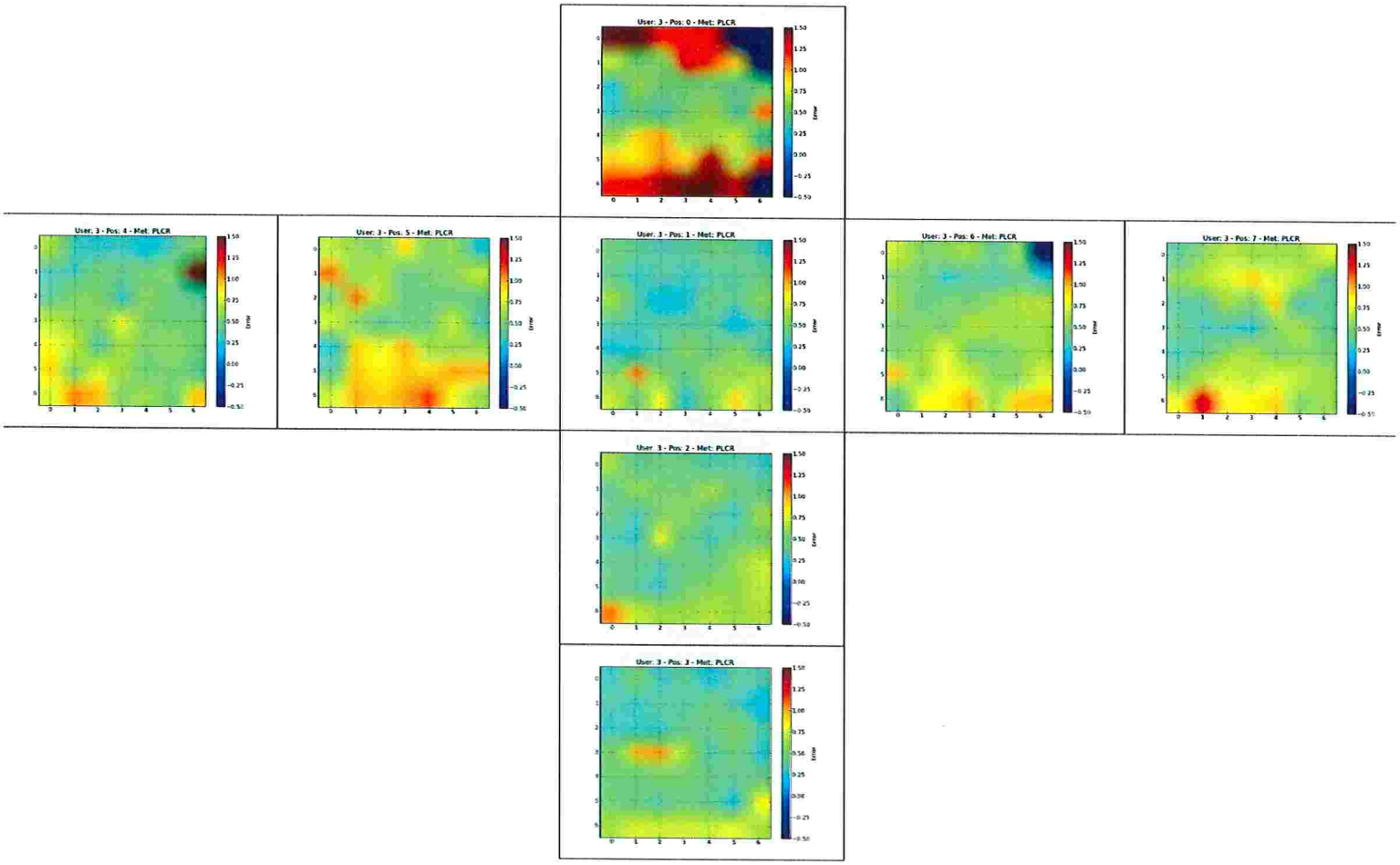

Figure B.12: *Distribution of the gaze error in all positions for participant 3 using method* ***PL-CR****. Position 1 was used for calibration.*

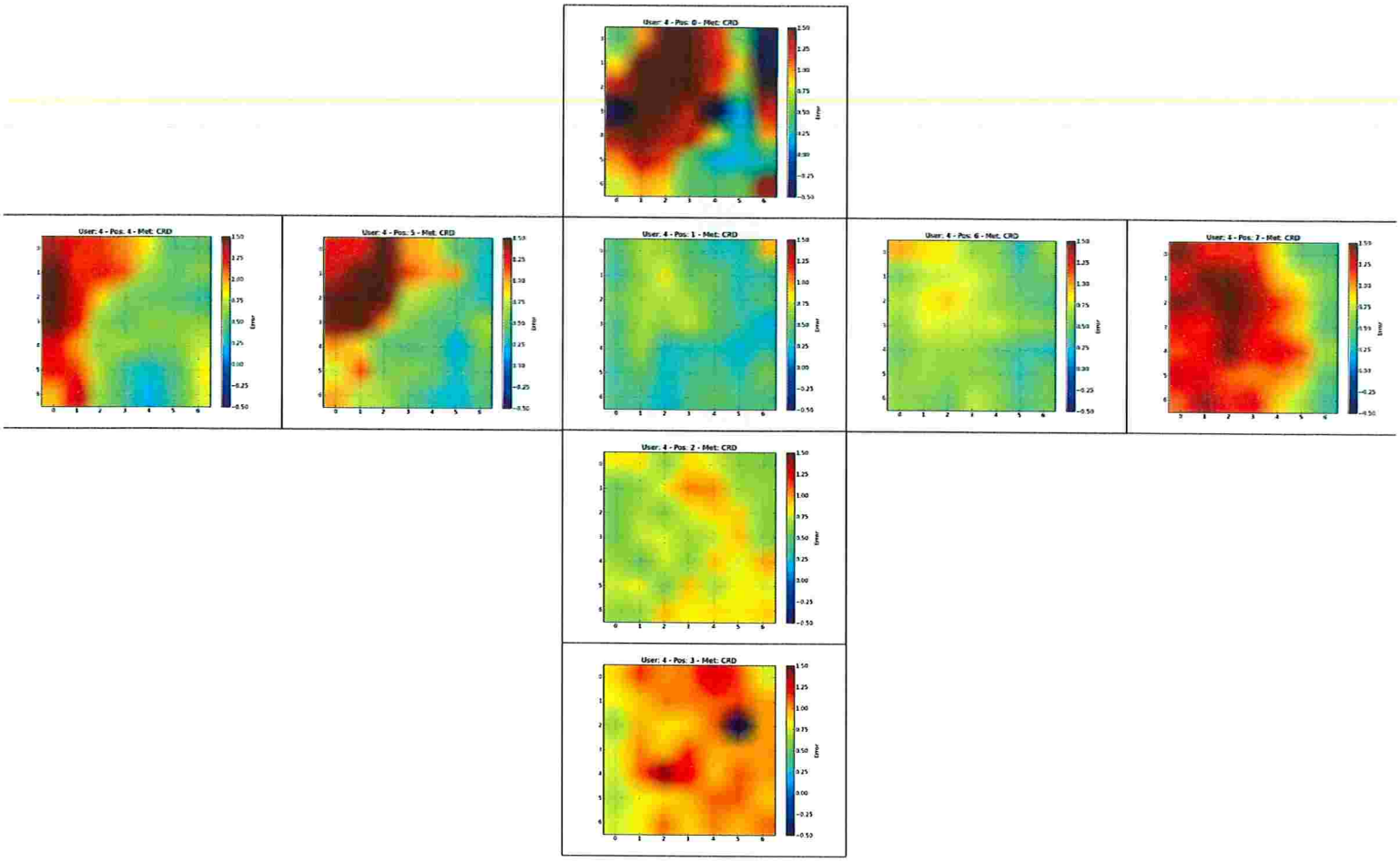

Figure B.13: *Distribution of the gaze error in all positions for participant 4 using method* ***CR-D****. Position 1 was used for calibration.*

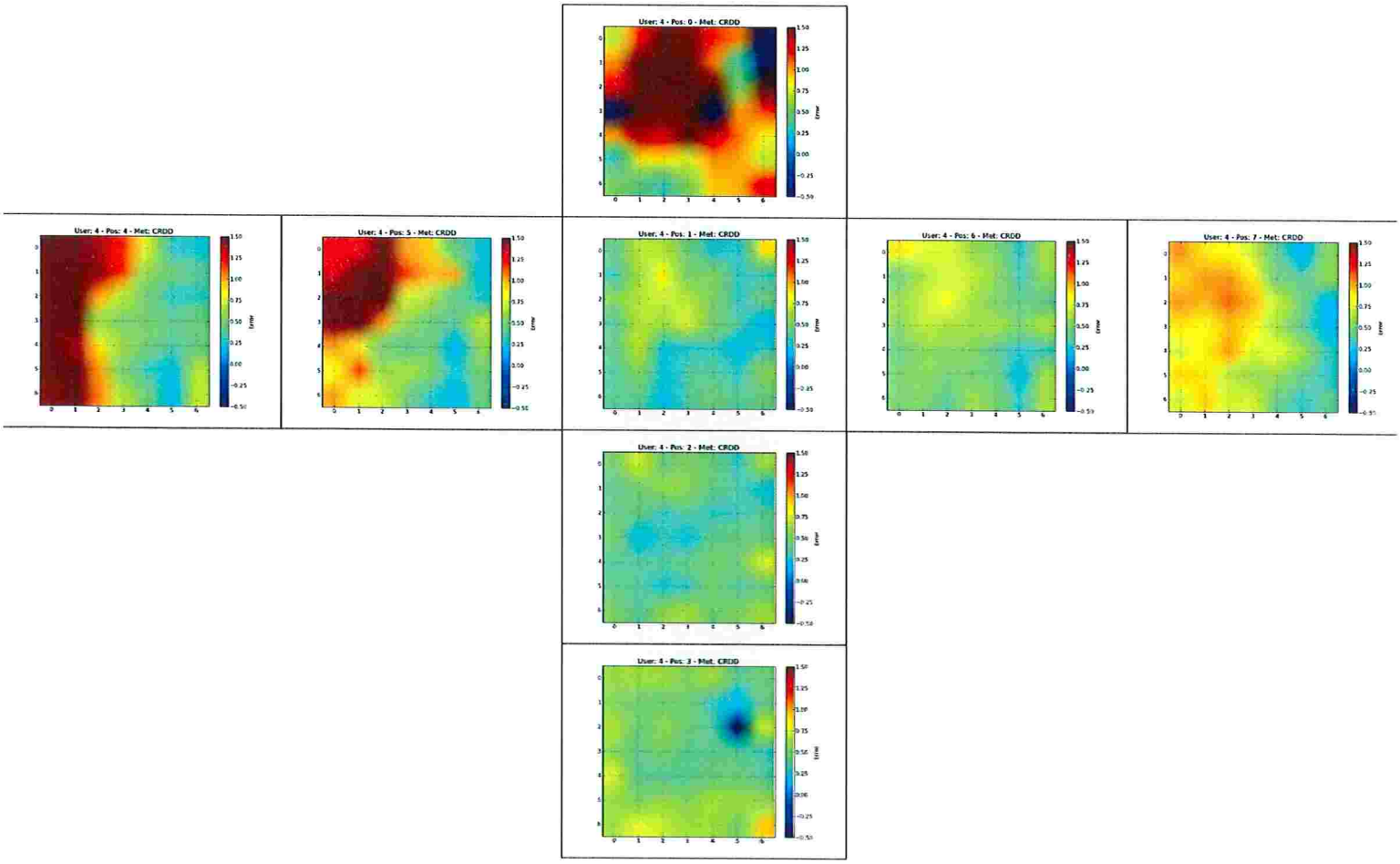

Figure B.14: *Distribution of the gaze error in all positions for participant 4 using method* ***CR-DD****. Position 1 was used for calibration.*

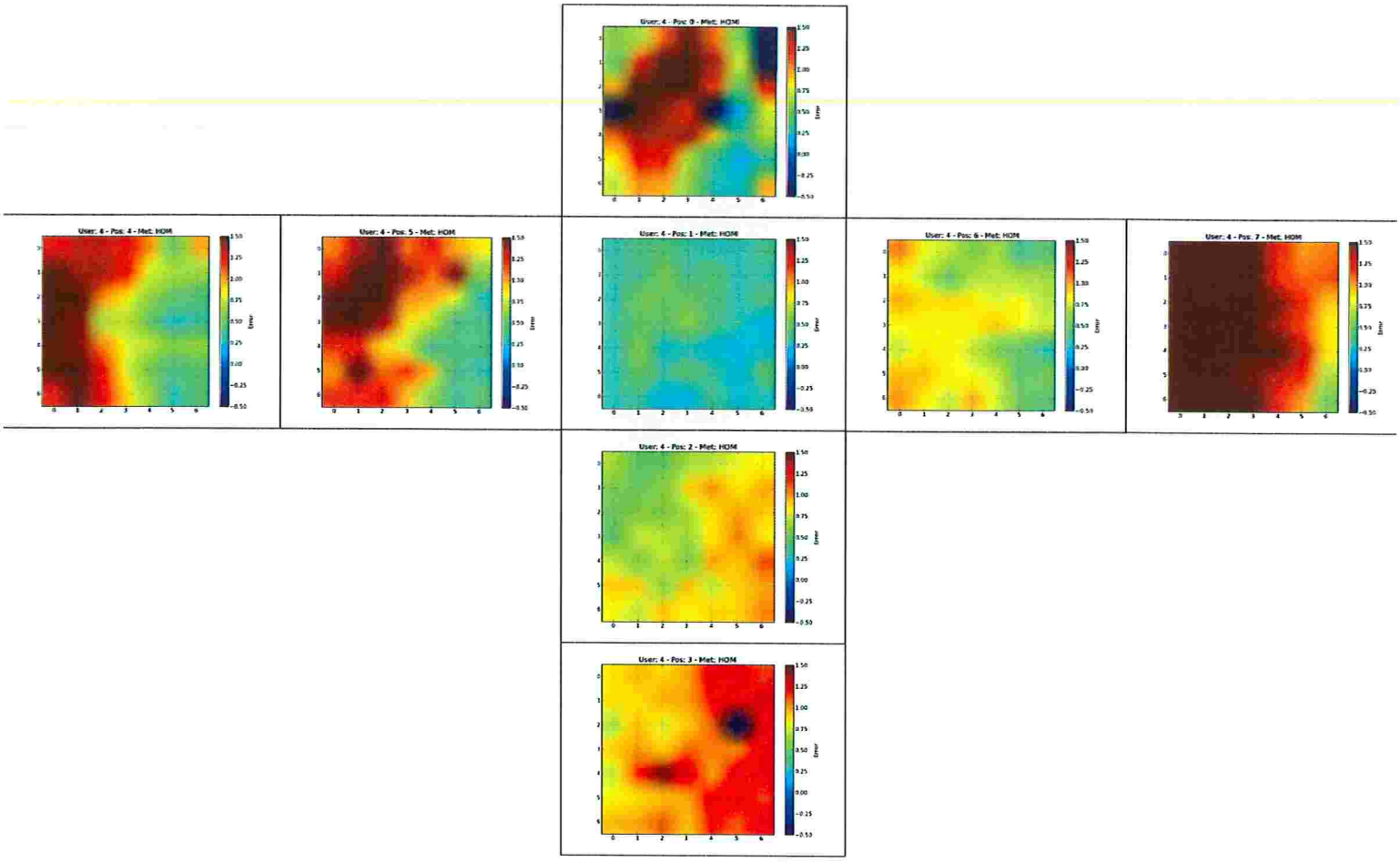

Figure B.15: *Distribution of the gaze error in all positions for participant 4 using method* ***HOM****. Position 1 was used for calibration.*

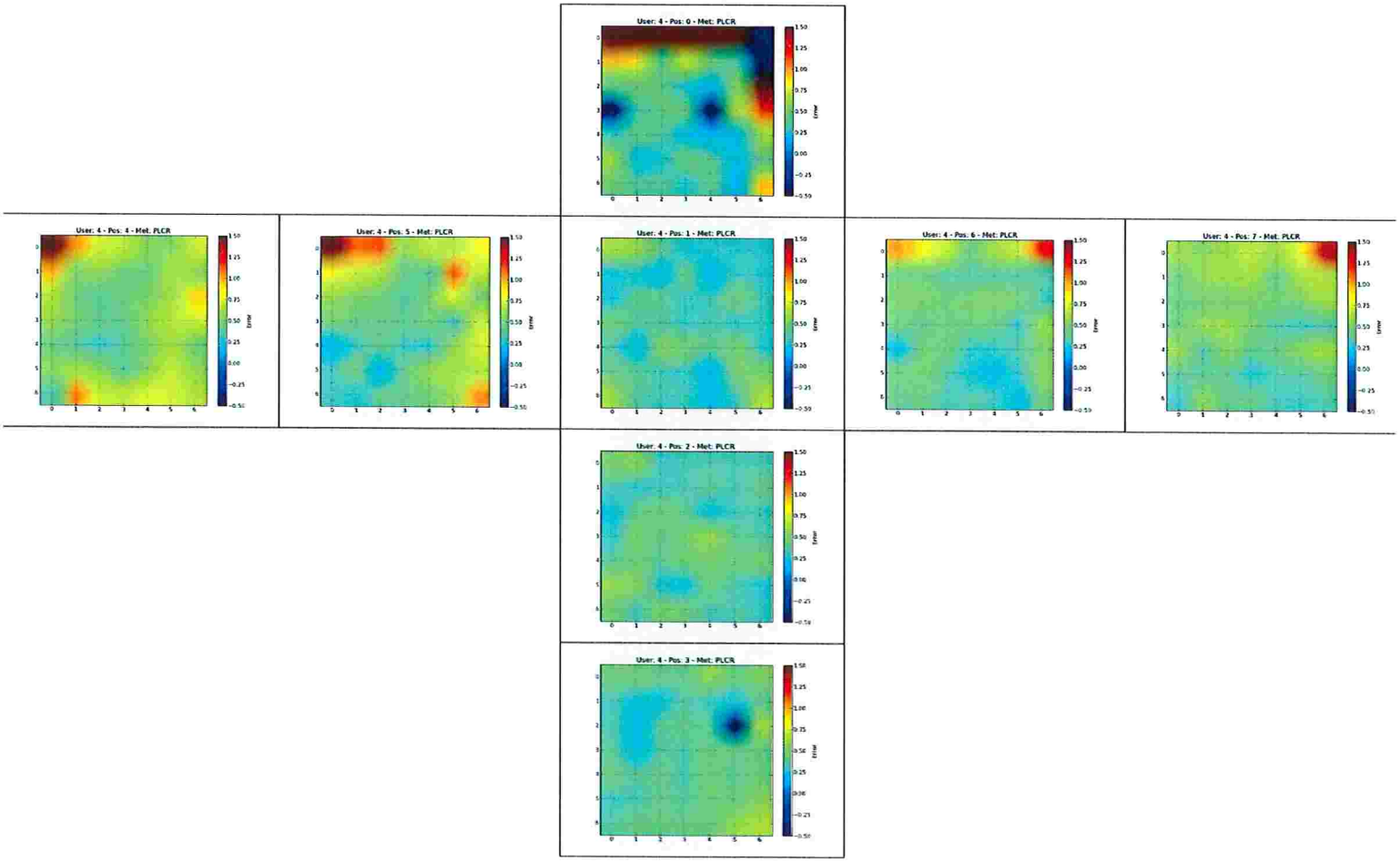

Figure B.16: *Distribution of the gaze error in all positions for participant 4 using method* ***PL-CR****. Position 1 was used for calibration.*

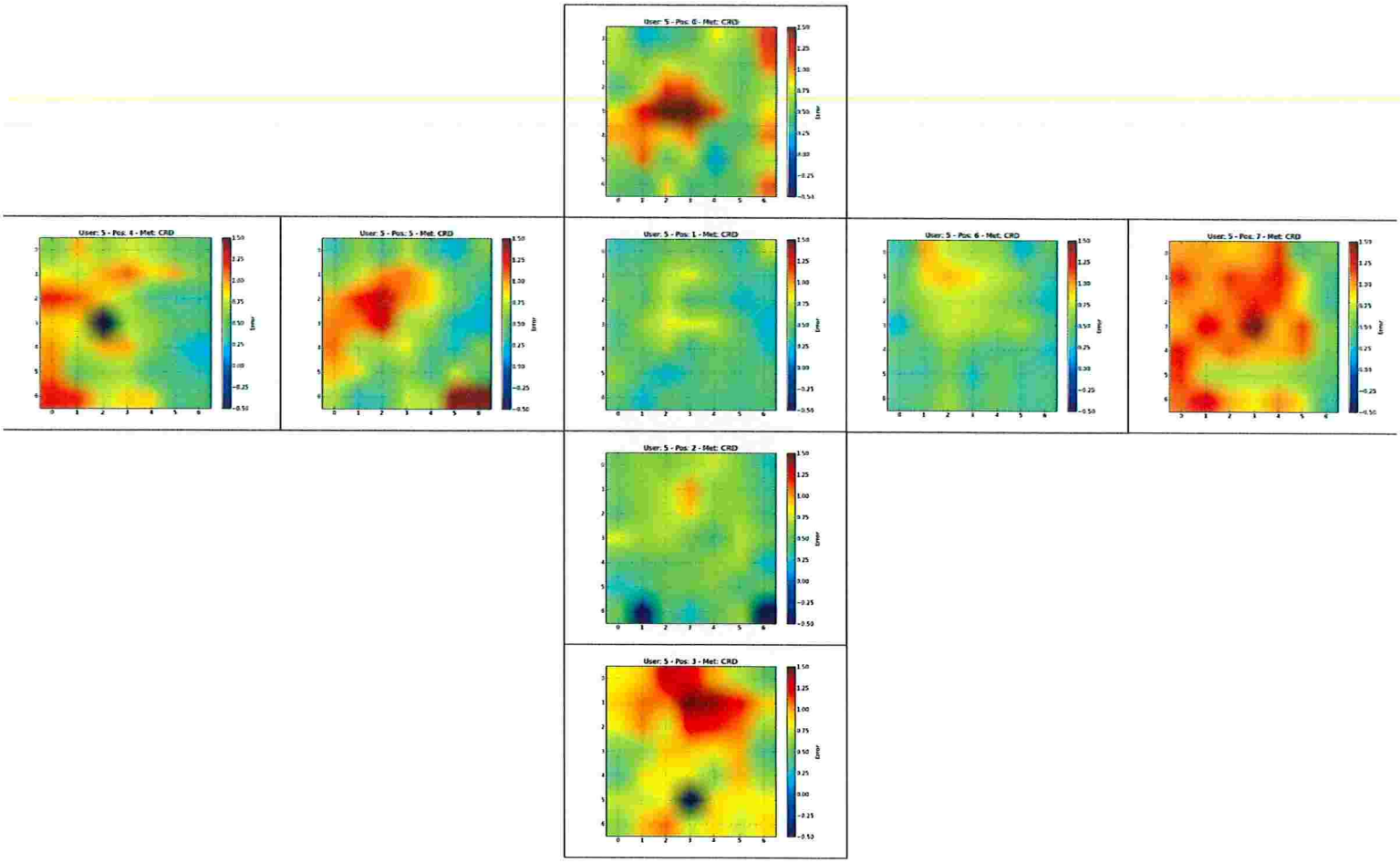

Figure B.17: *Distribution of the gaze error in all positions for participant 5 using method* ***CR-D****. Position 1 was used for calibration.*

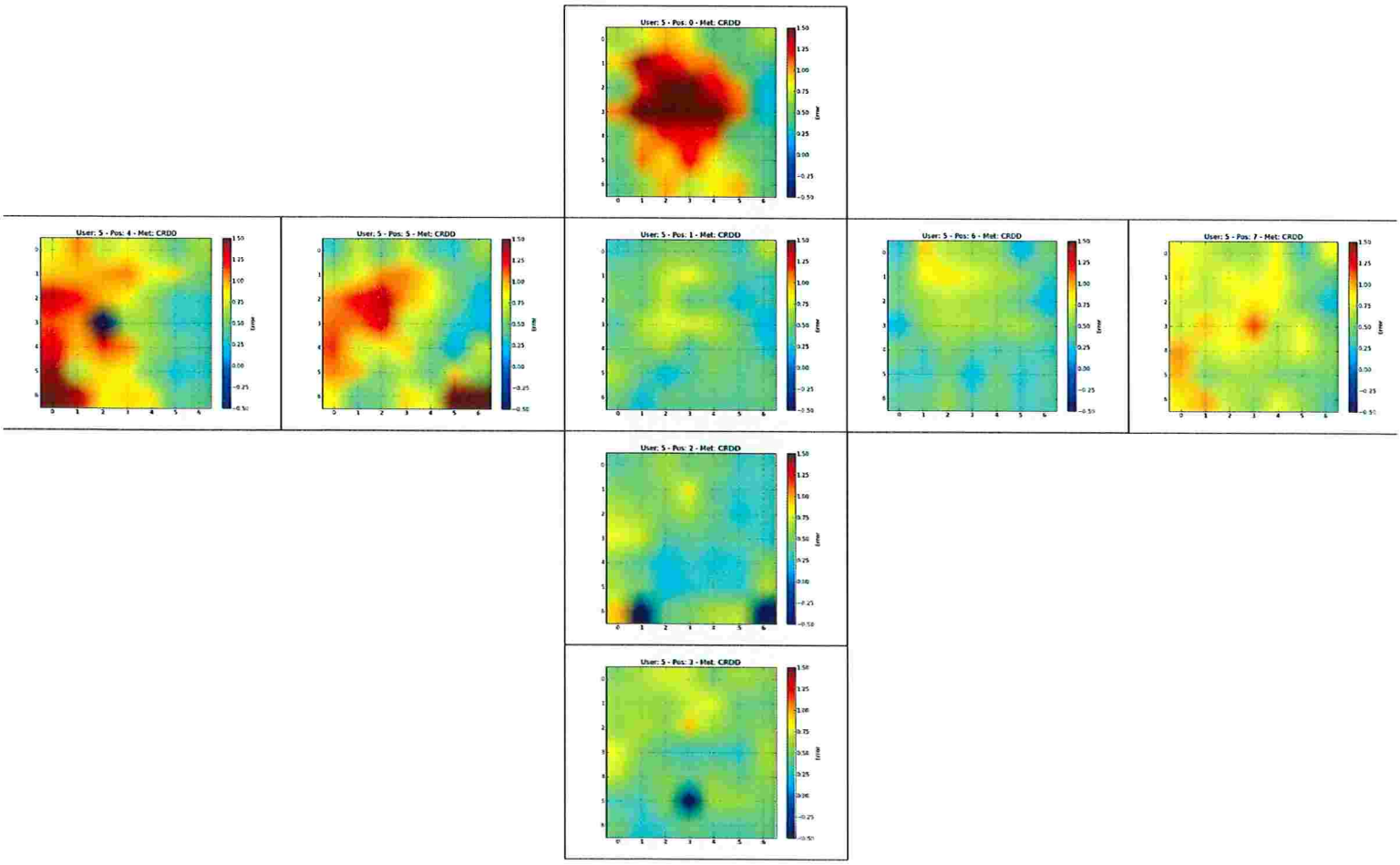

Figure B.18: *Distribution of the gaze error in all positions for participant 5 using method* ***CR-DD****. Position 1 was used for calibration.*

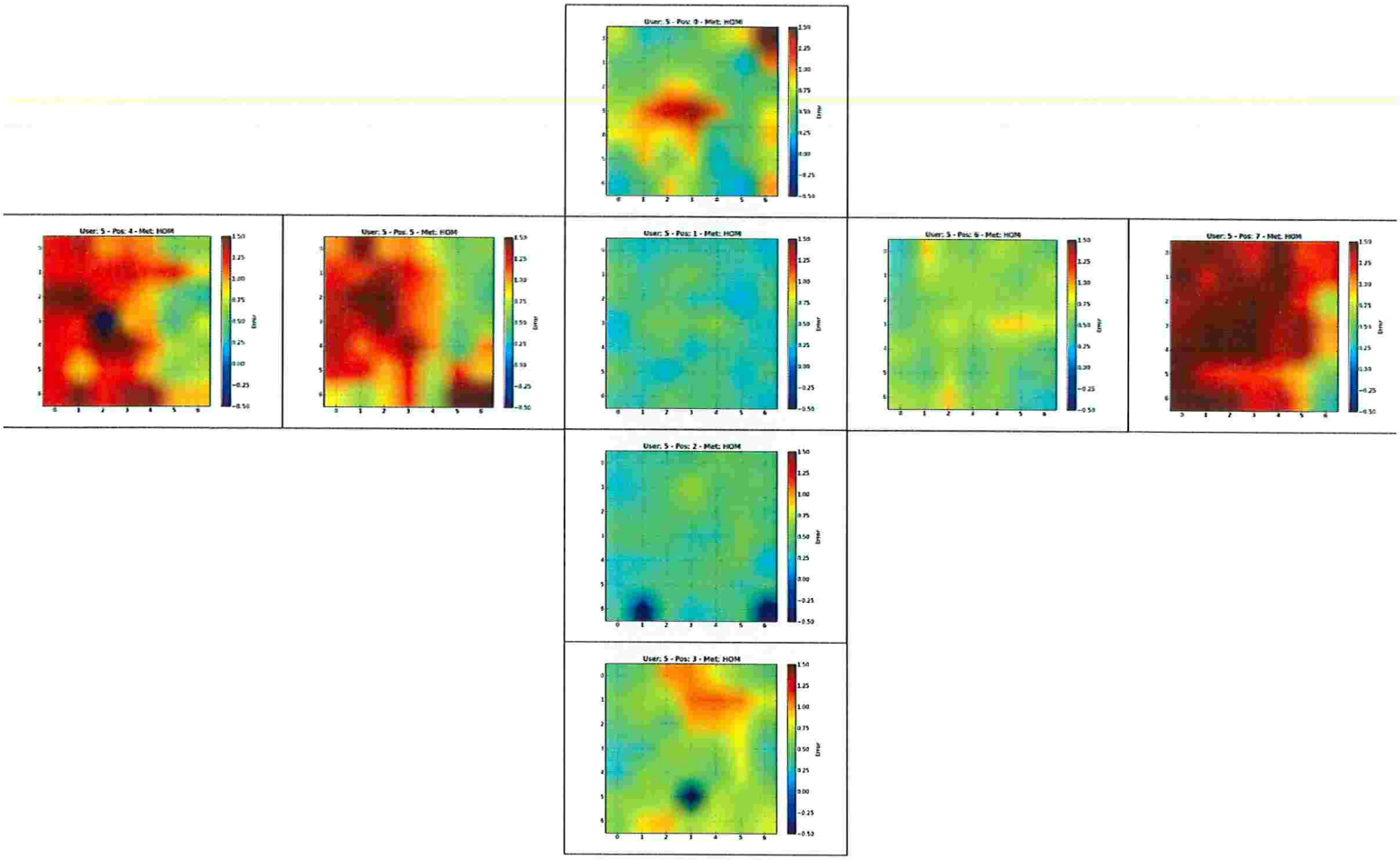

Figure B.19: *Distribution of the gaze error in all positions for participant 5 using method* ***HOM****. Position 1 was used for calibration.*

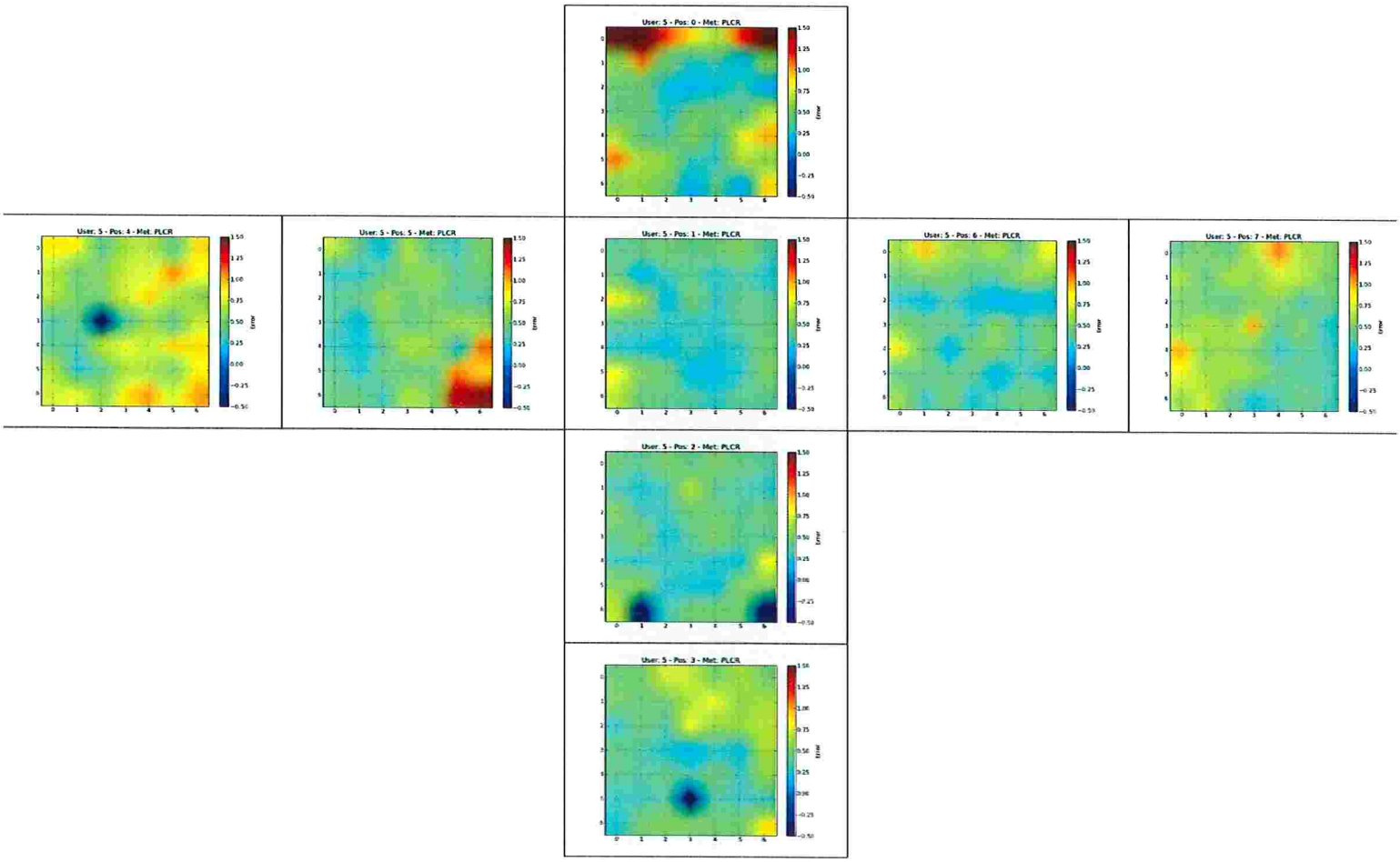

Figure B.20: *Distribution of the gaze error in all positions for participant 5 using method* ***PL-CR****. Position 1 was used for calibration.*

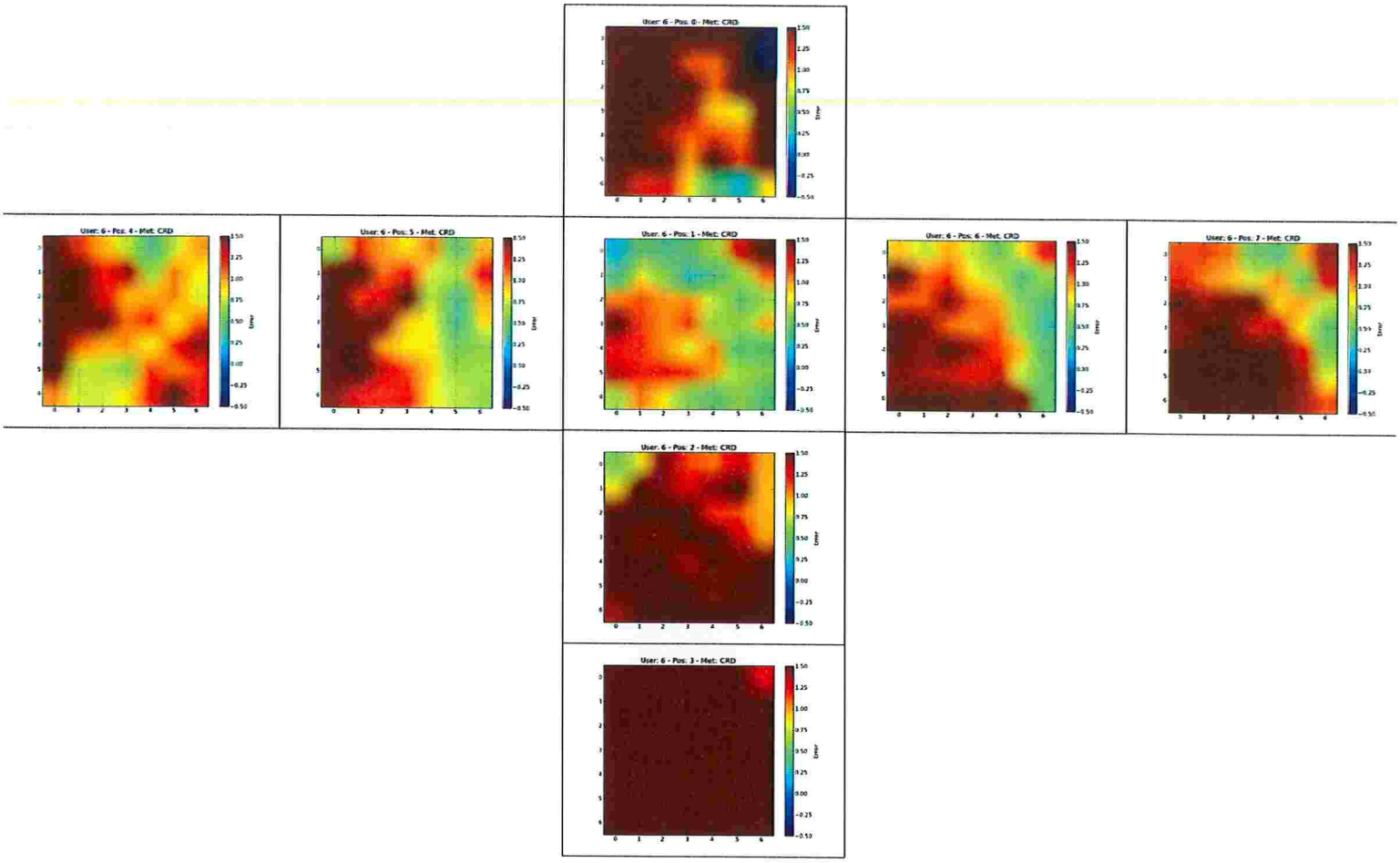

Figure B.21: *Distribution of the gaze error in all positions for participant 6 using method* ***CR-D****. Position 1 was used for calibration.*

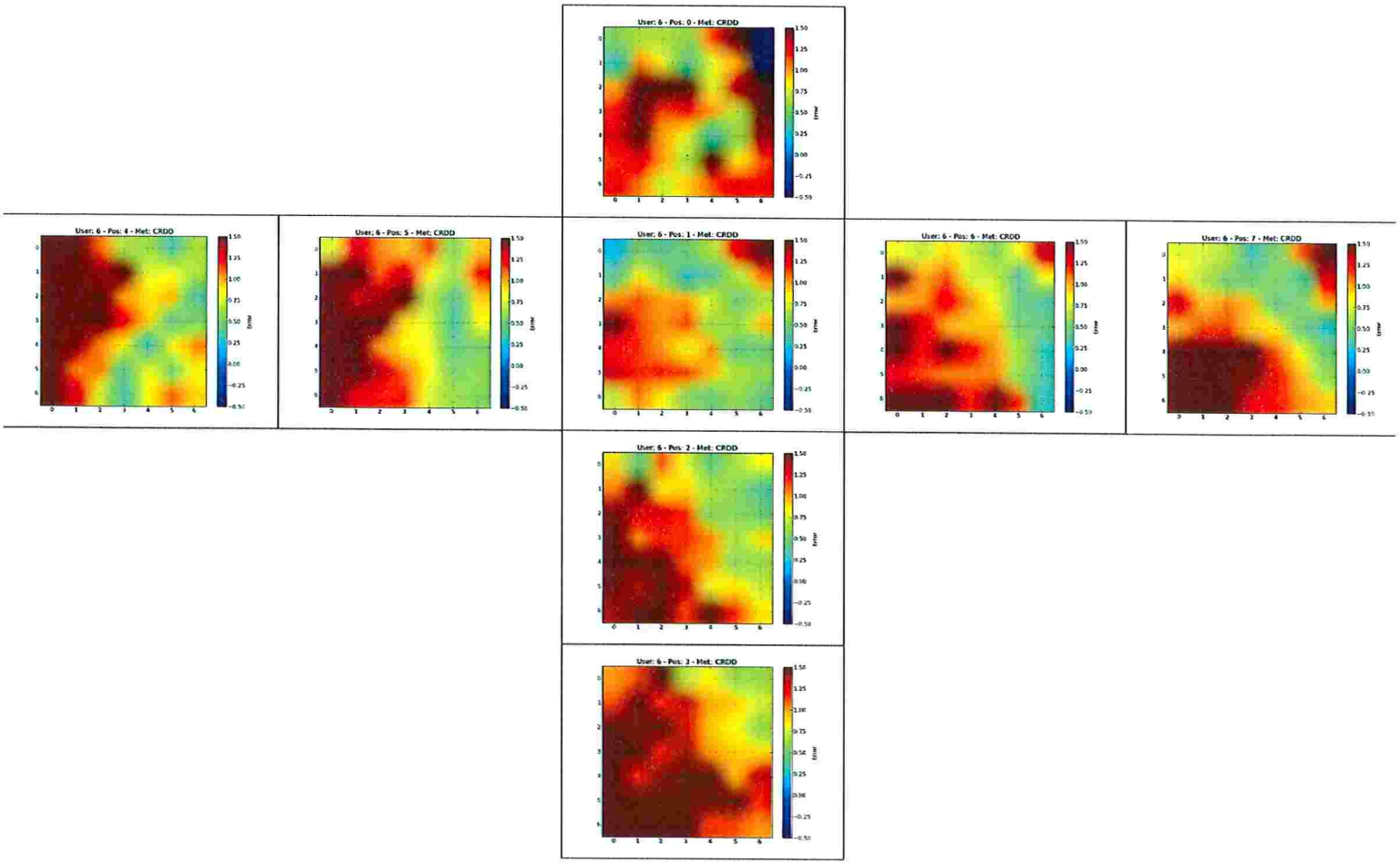

Figure B.22: *Distribution of the gaze error in all positions for participant 6 using method* ***CR-DD****. Position 1 was used for calibration.*

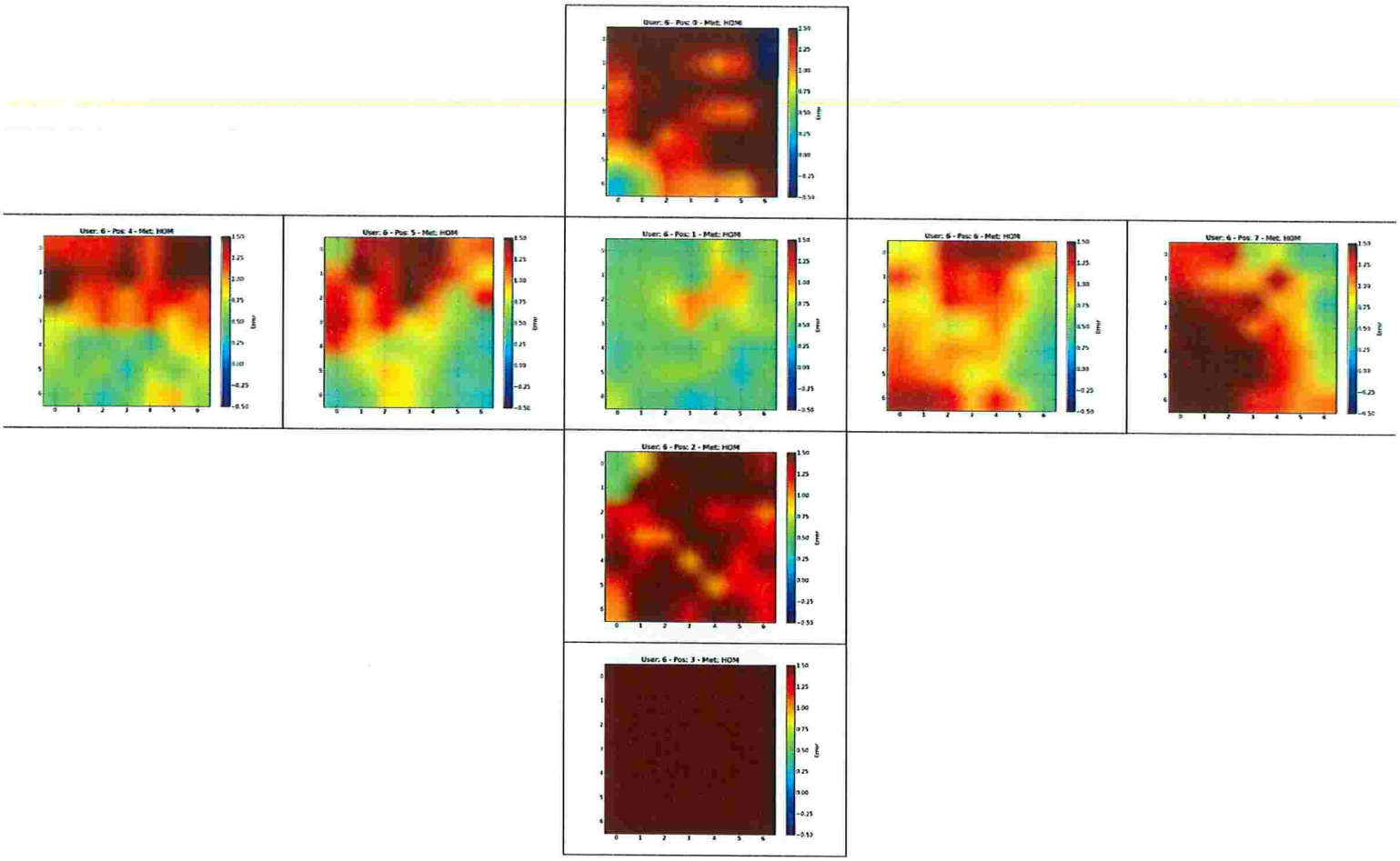

Figure B.23: *Distribution of the gaze error in all positions for participant 6 using method* ***HOM****. Position 1 was used for calibration.*

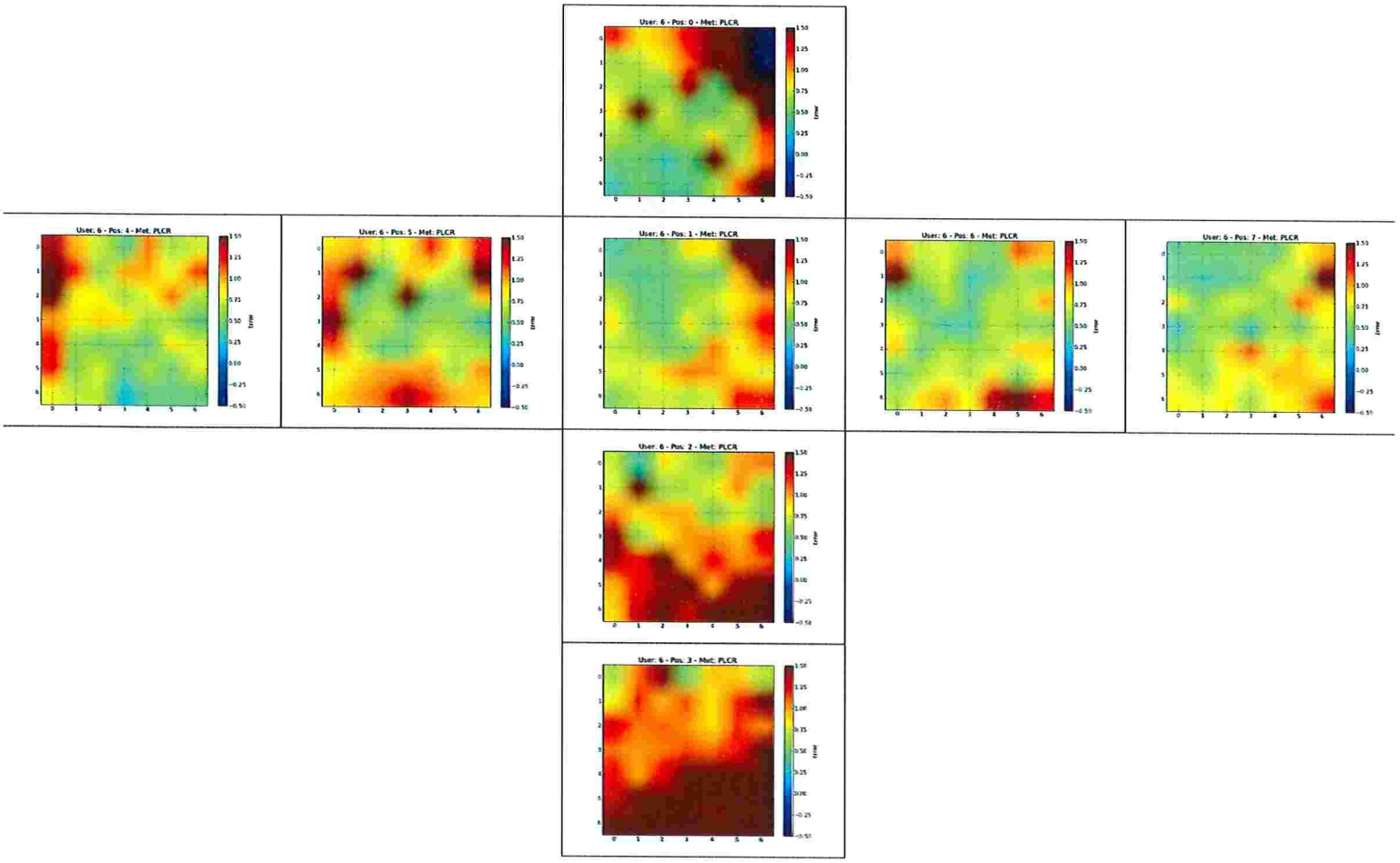

Figure B.24: *Distribution of the gaze error in all positions for participant 6 using method* ***PL-CR****. Position 1 was used for calibration.*

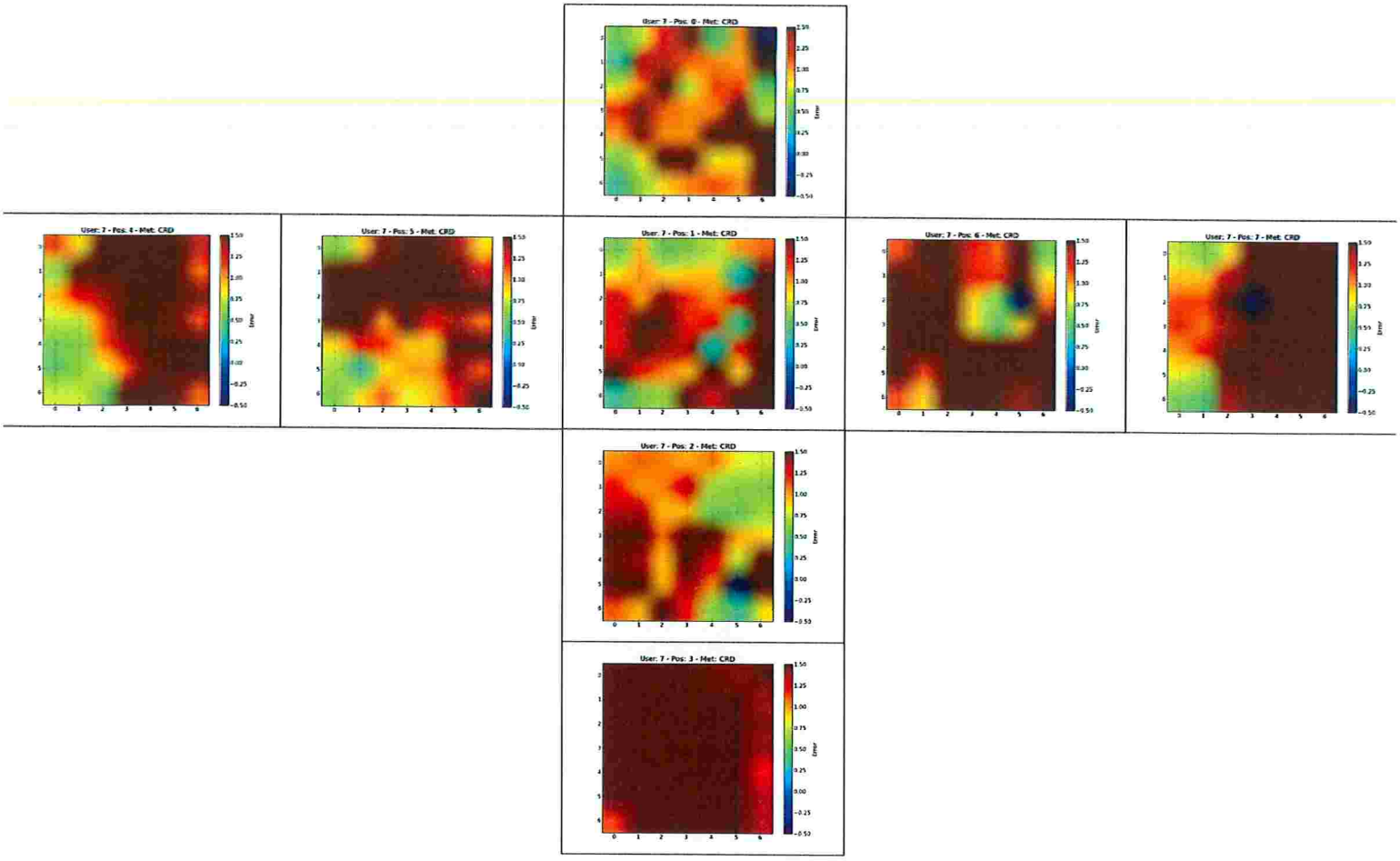

Figure B.25: *Distribution of the gaze error in all positions for participant 7 using method* ***CR-D****. Position 1 was used for calibration.*

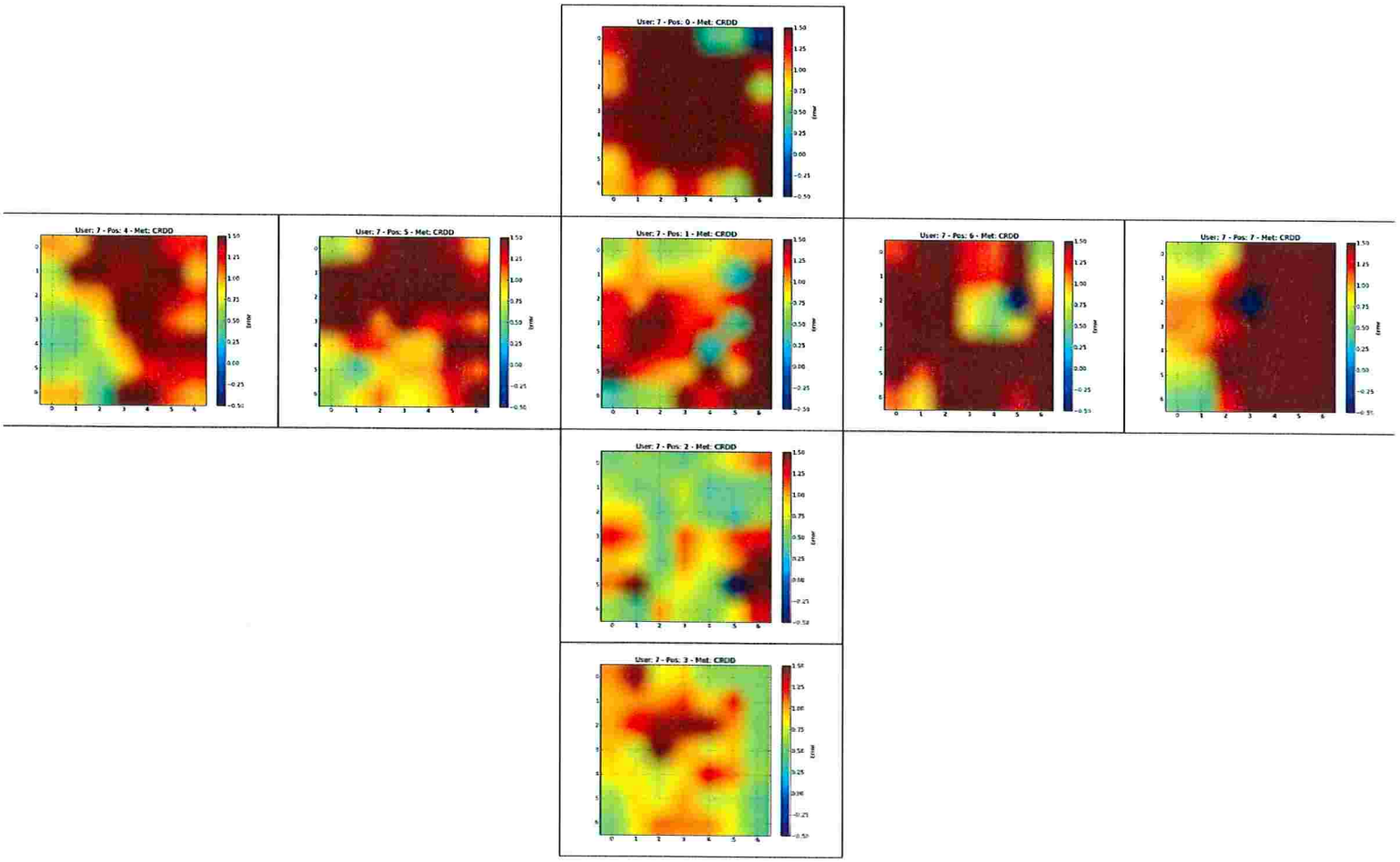

Figure B.26: *Distribution of the gaze error in all positions for participant 7 using method* ***CR-DD****. Position 1 was used for calibration.*

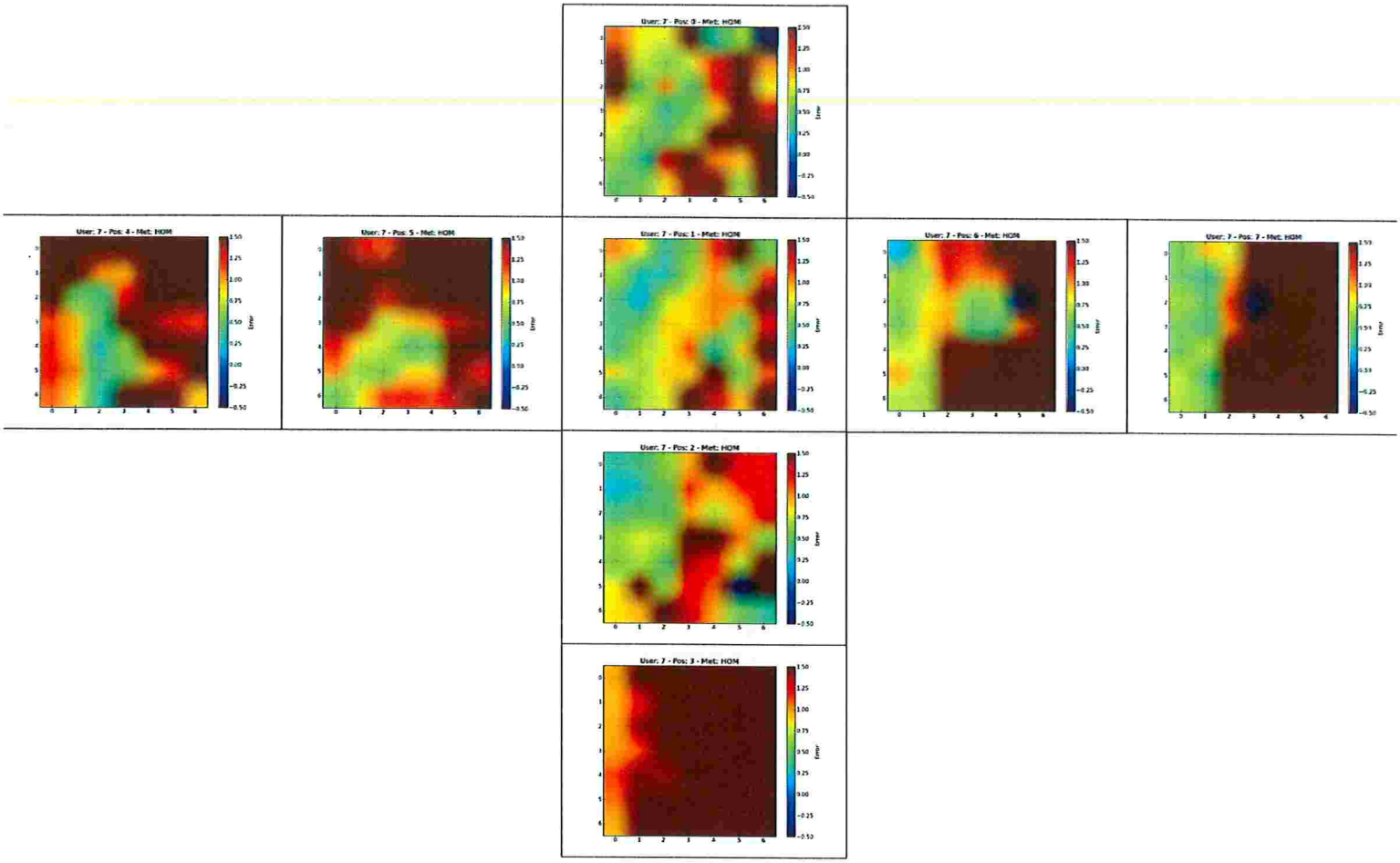

Figure B.27: *Distribution of the gaze error in all positions for participant 7 using method* ***HOM****. Position 1 was used for calibration.*

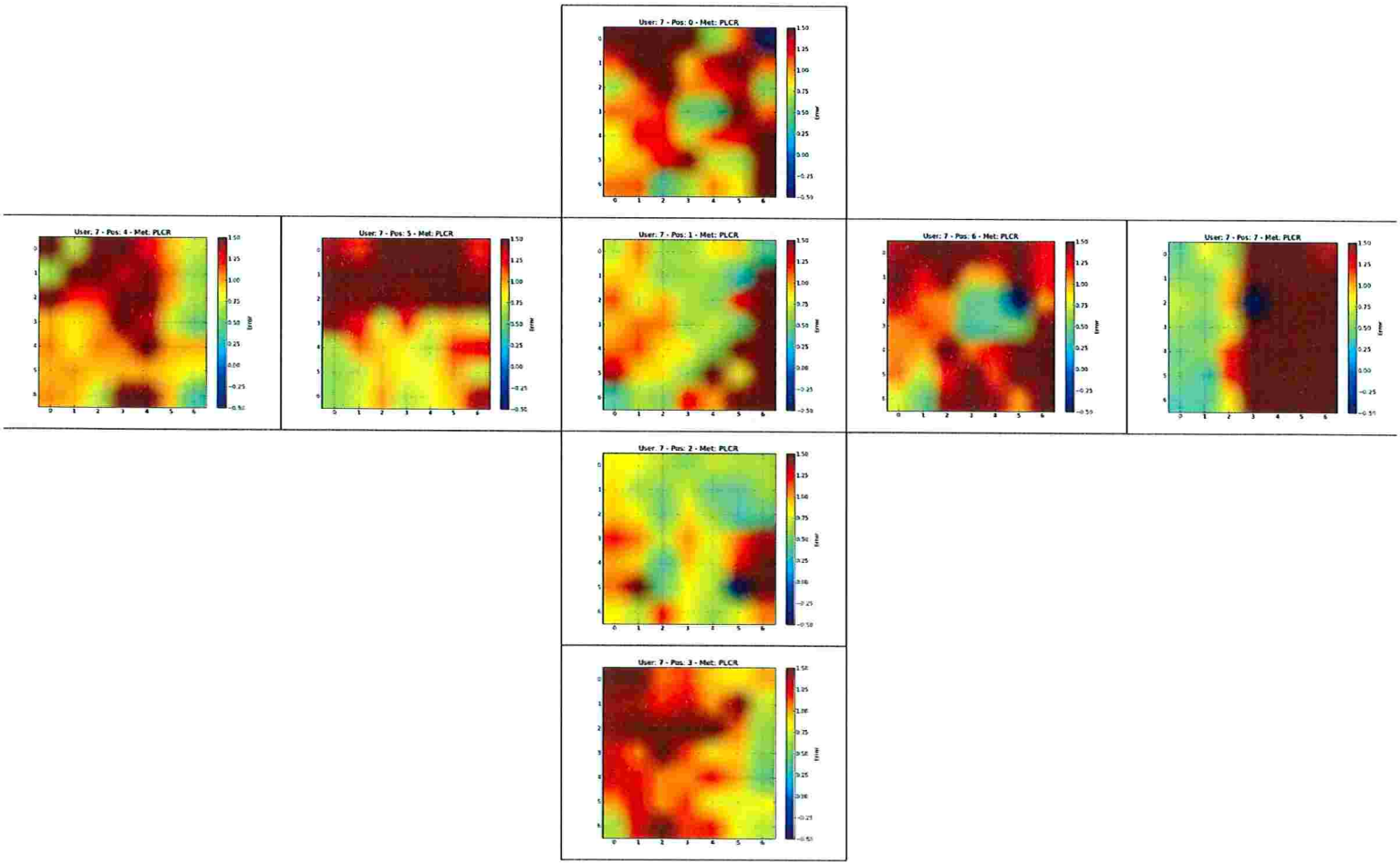

Figure B.28: *Distribution of the gaze error in all positions for participant 7 using method* ***PL-CR****. Position 1 was used for calibration.*

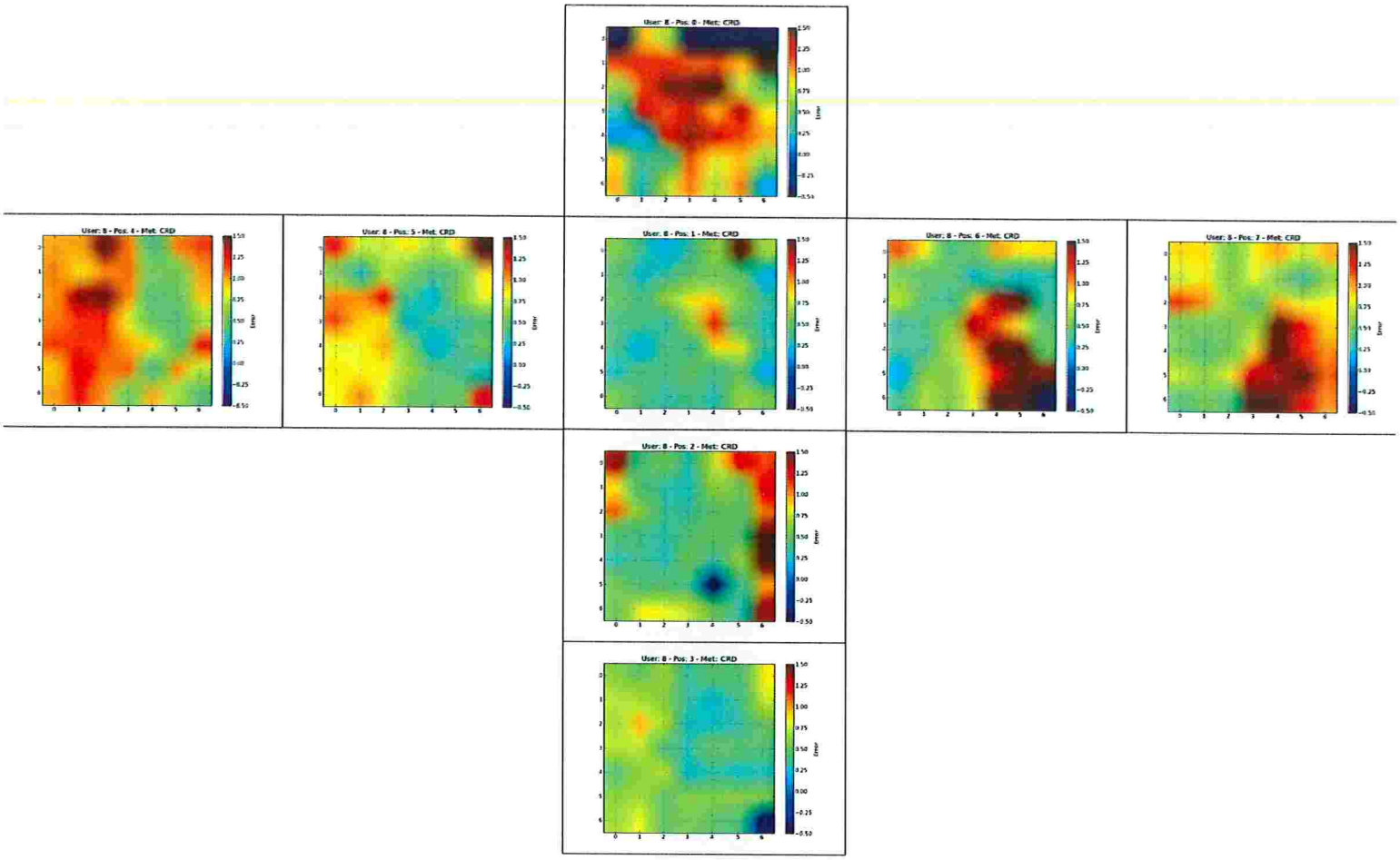

Figure B.29: *Distribution of the gaze error in all positions for participant 8 using method* ***CR-D****. Position 1 was used for calibration.*

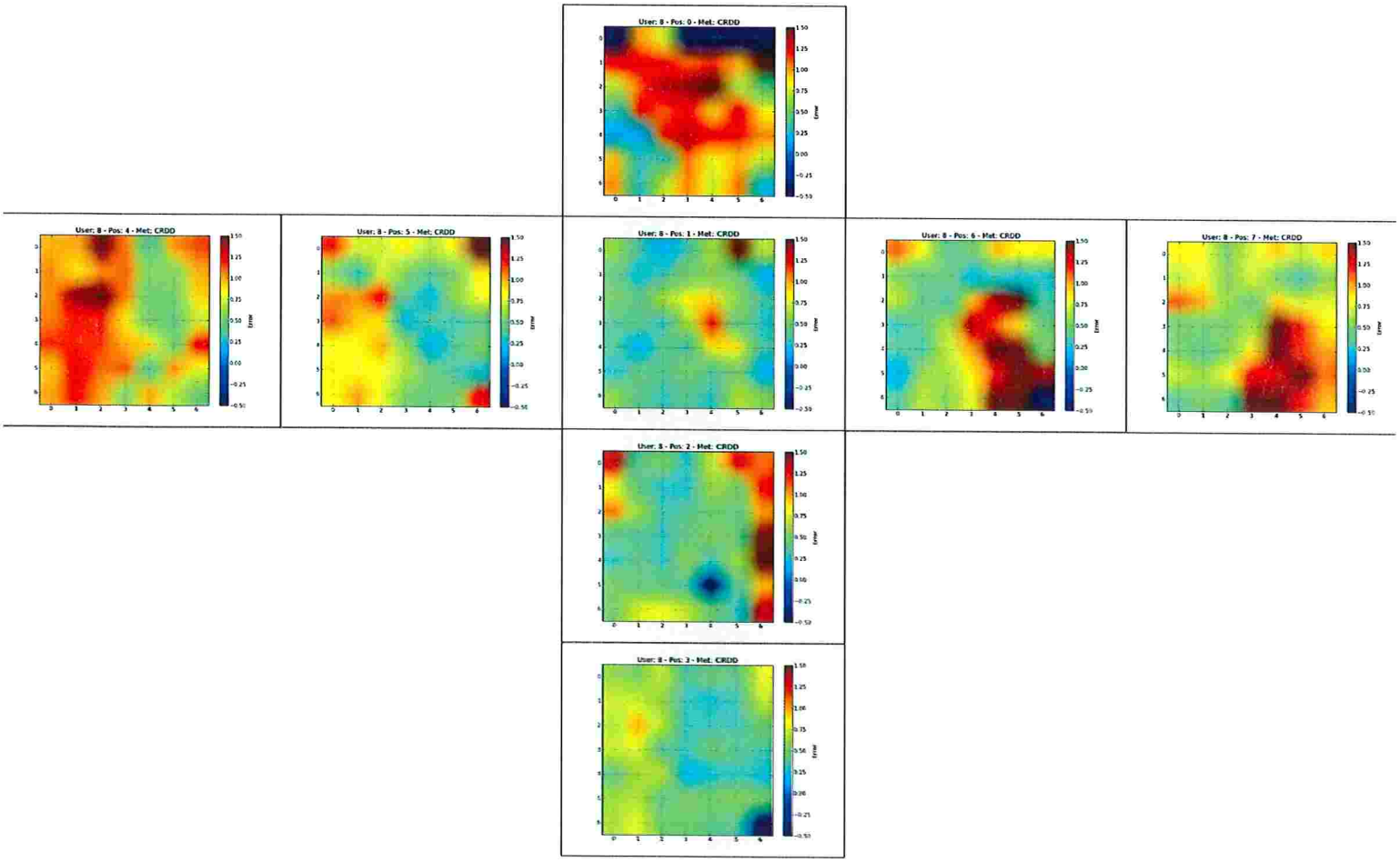

Figure B.30: *Distribution of the gaze error in all positions for participant 8 using method* ***CR-DD****. Position 1 was used for calibration.*

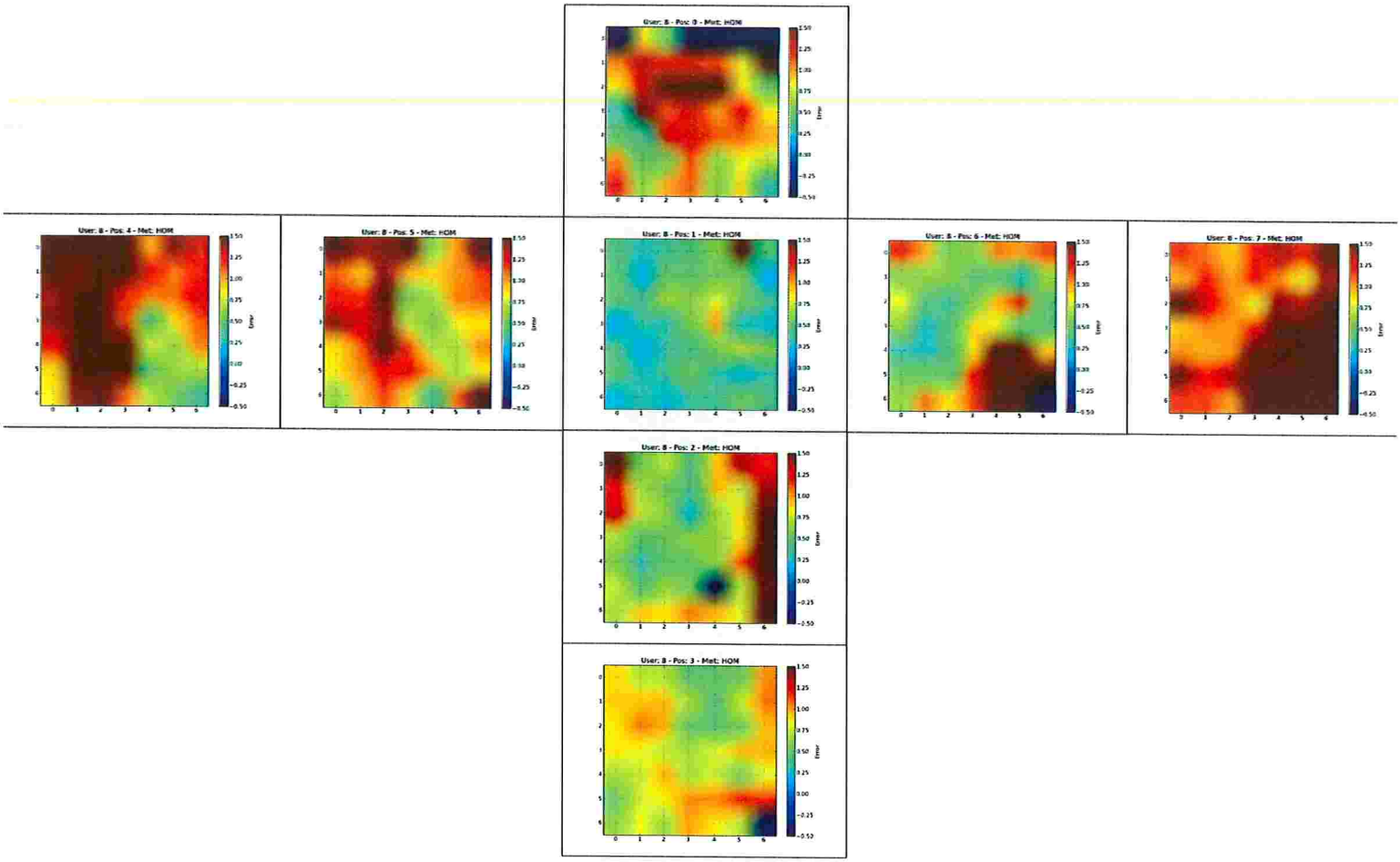

Figure B.31: *Distribution of the gaze error in all positions for participant 8 using method* ***HOM****. Position 1 was used for calibration.*

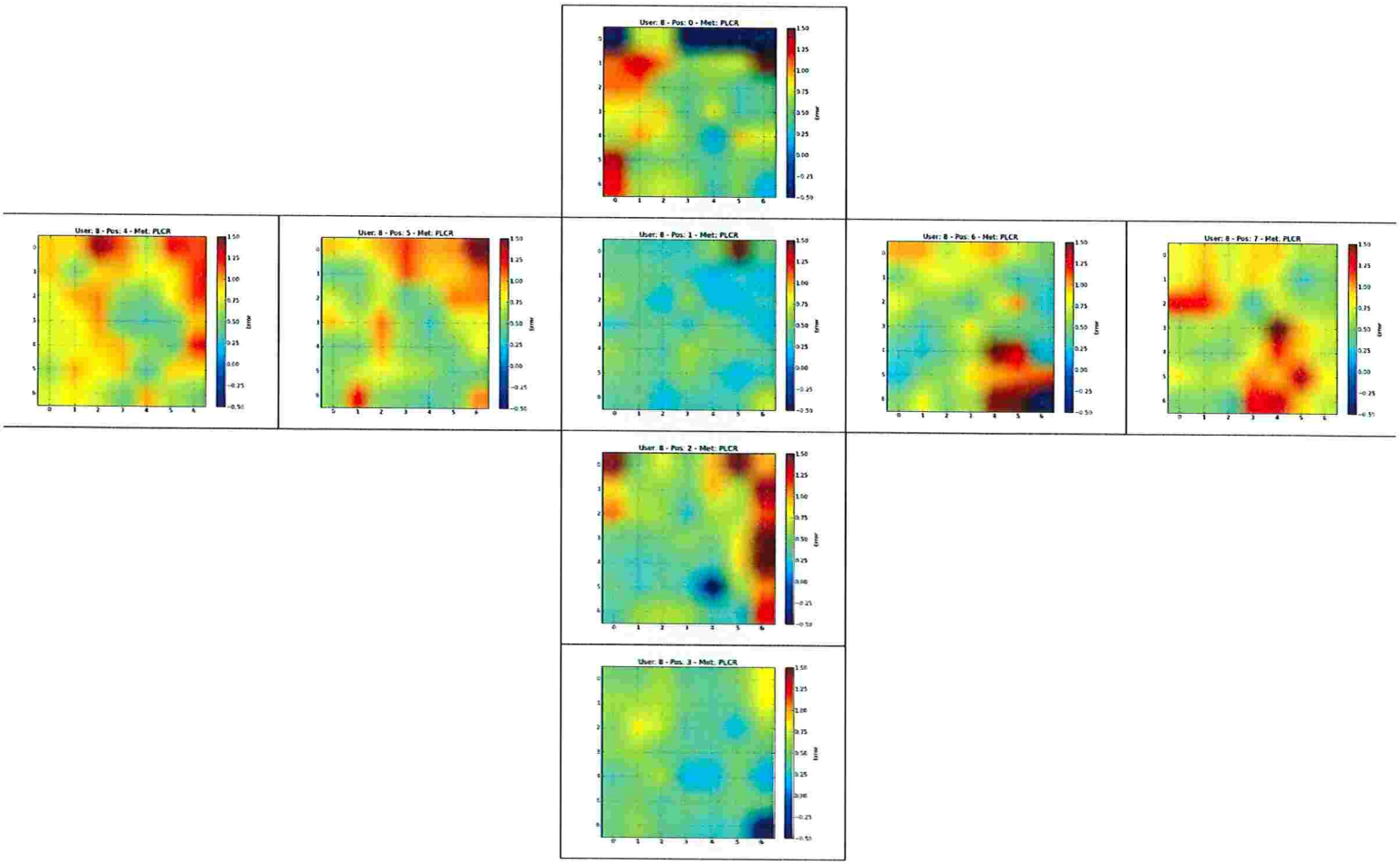

Figure B.32: *Distribution of the gaze error in all positions for participant 8 using method* ***PL-CR****. Position 1 was used for calibration.*

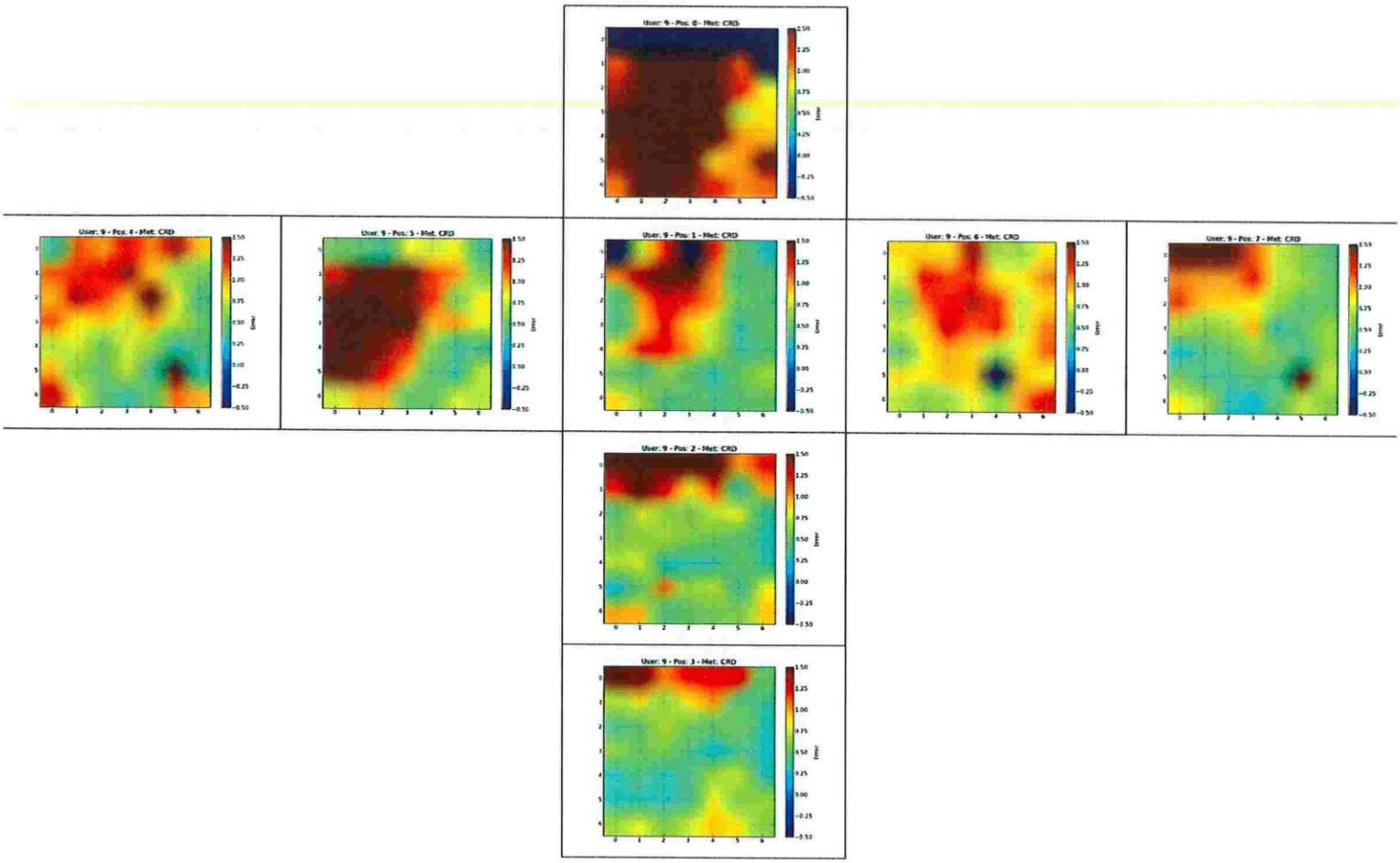

Figure B.33: *Distribution of the gaze error in all positions for participant 9 using method* ***CR-D****. Position 1 was used for calibration.*

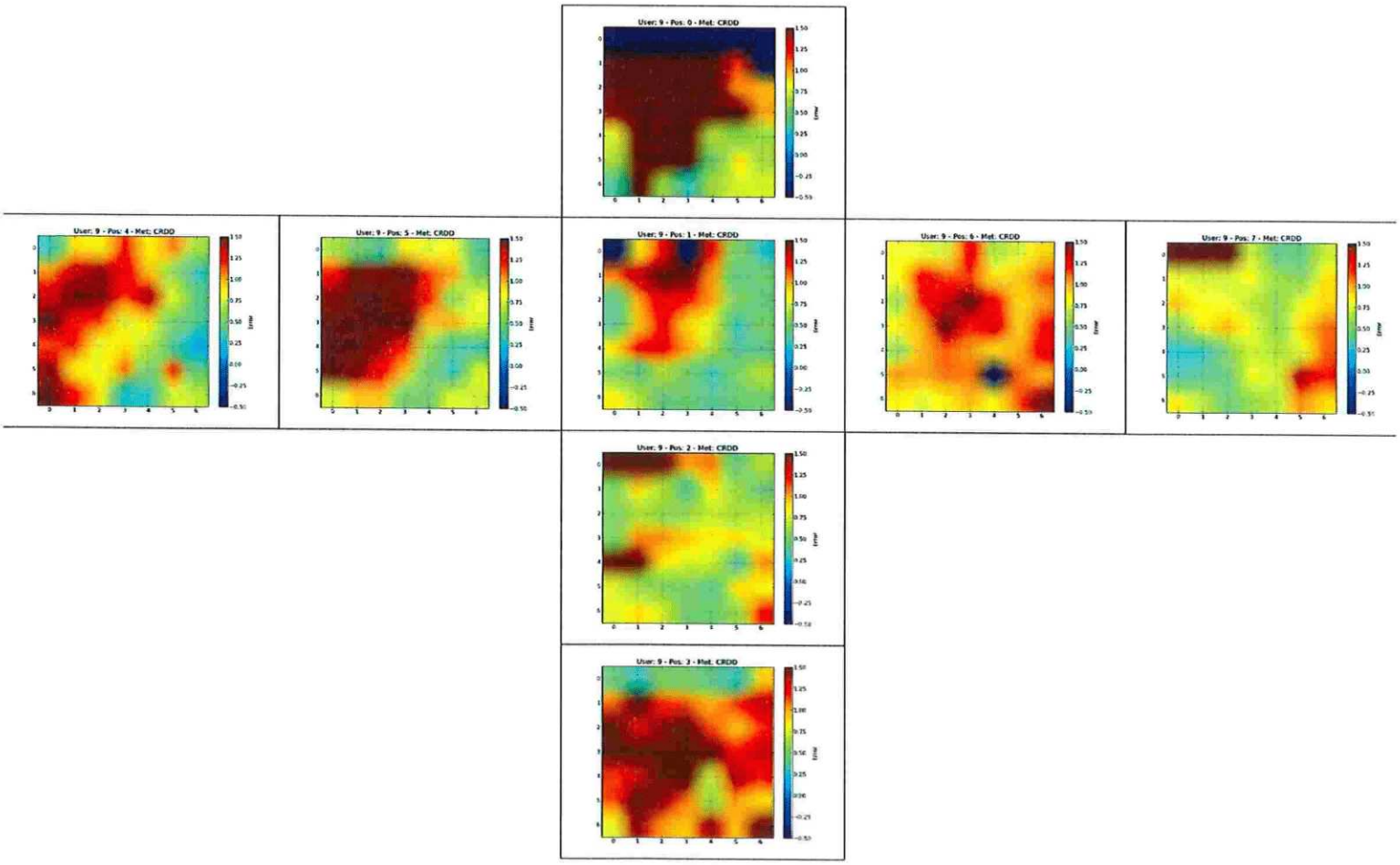

Figure B.34: *Distribution of the gaze error in all positions for participant 9 using method* ***CR-DD****. Position 1 was used for calibration.*

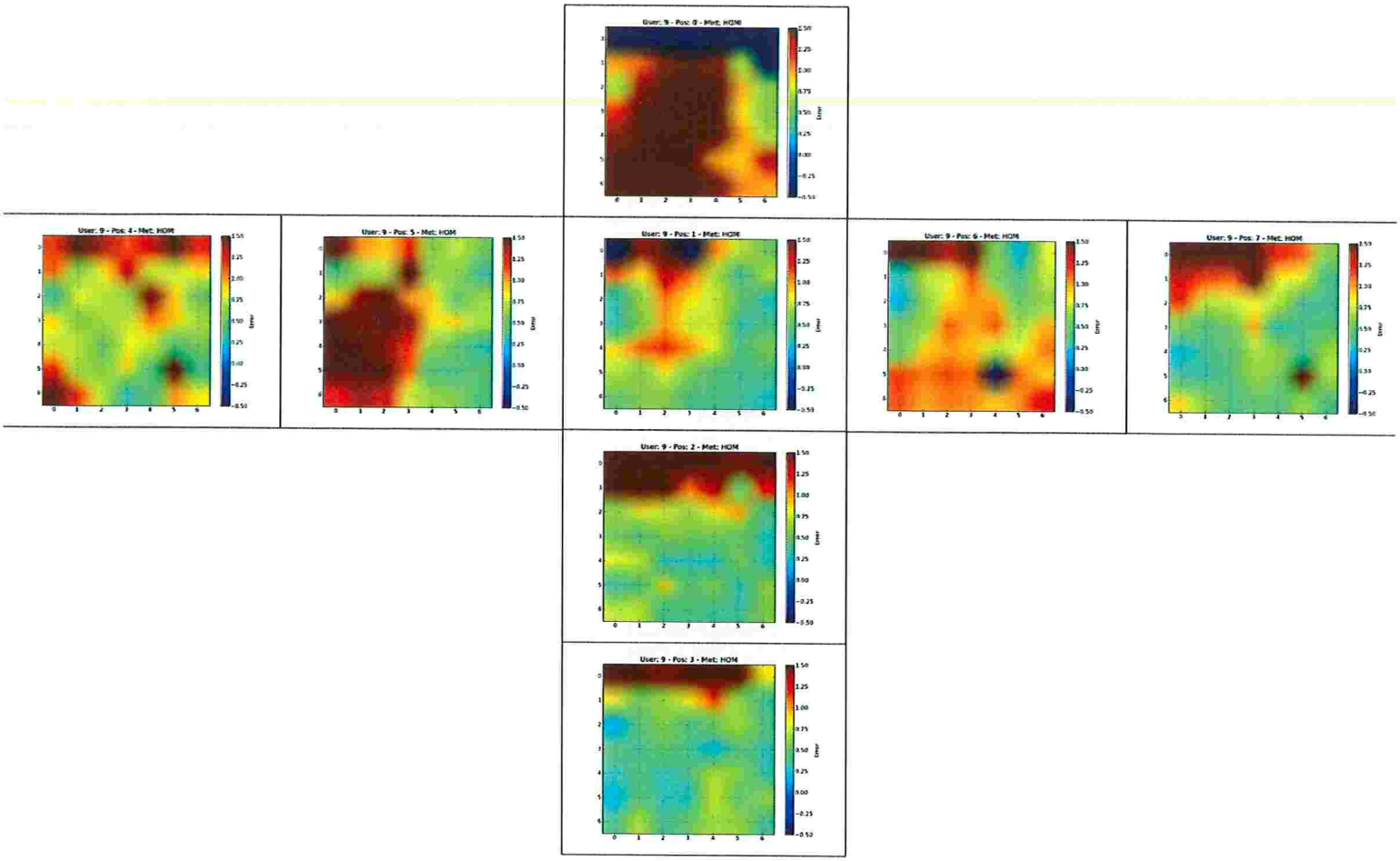

Figure B.35: *Distribution of the gaze error in all positions for participant 9 using method* ***HOM****. Position 1 was used for calibration.*

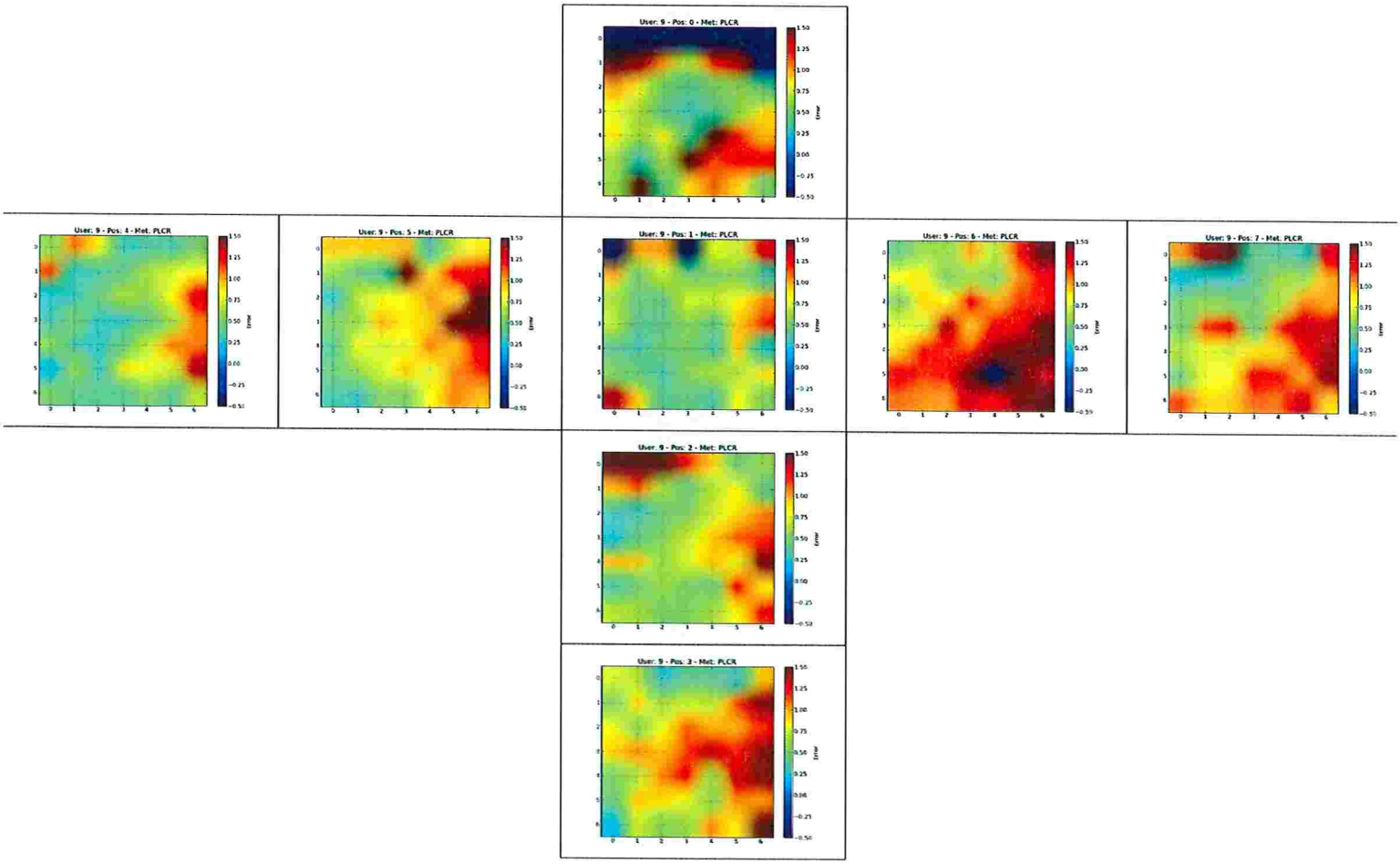

Figure B.36: *Distribution of the gaze error in all positions for participant 9 using method* ***PL-CR****. Position 1 was used for calibration.*

B.2 Detailed results for experiment 2

Detailed results for the experiment 2 are presented in this section. Each figure displays the distribution of the gaze error in all positions for a given subject's eye and gaze estimation method. Head positions for this experiment follows the layout of Figure A.14. A total of 7 subjects participated in this experiment. For each subject results for both right and left eyes were computed and are shown in this section. The gaze estimation methods considered for this second experiment were the **CR-D**, **CR-DD**, **HOM**, and **PL-CR** methods.

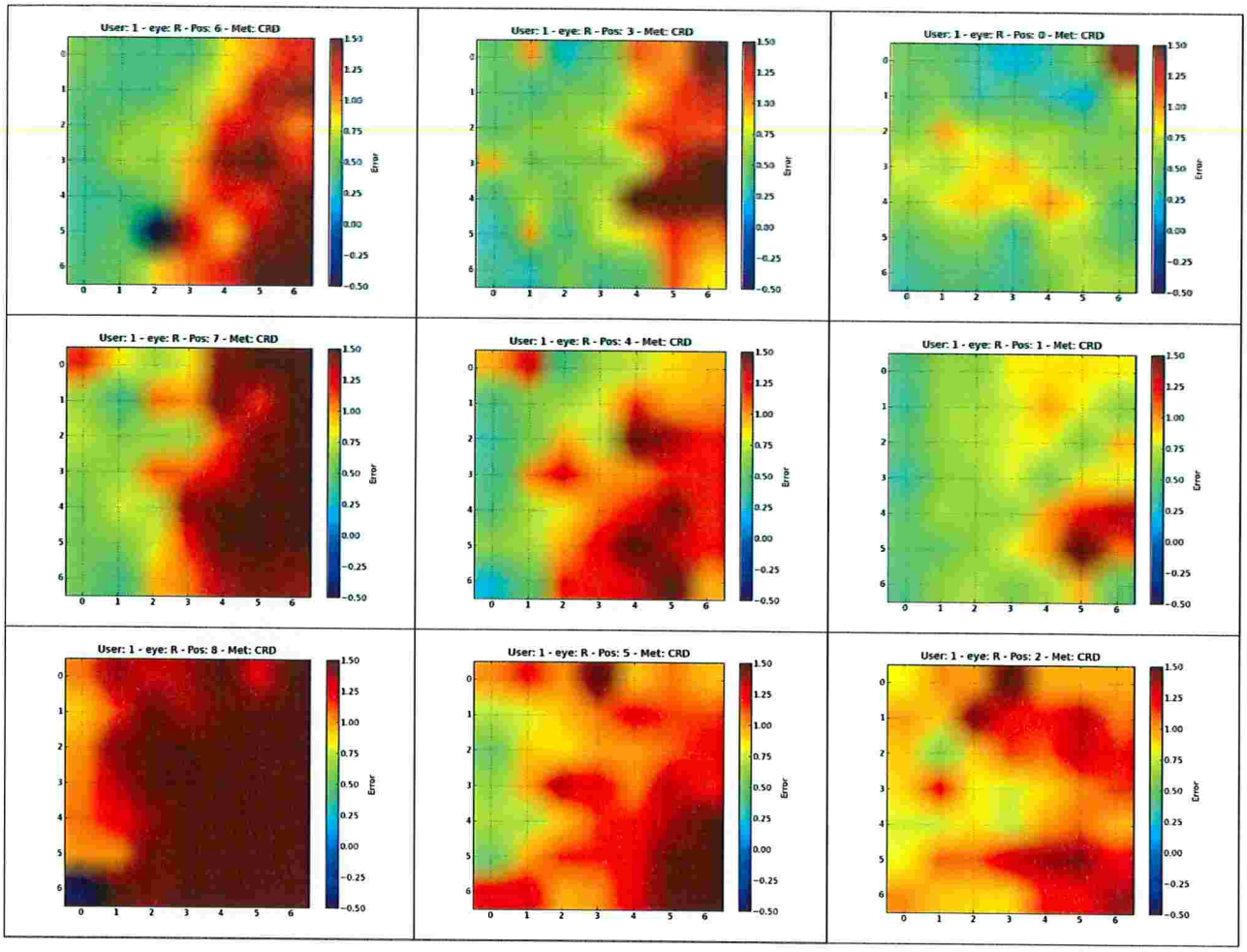

Figure B.37: *Distribution of the gaze error in all positions for the right eye of participant 1 using method* ***CR-D****. Position 0 was used for calibration.*

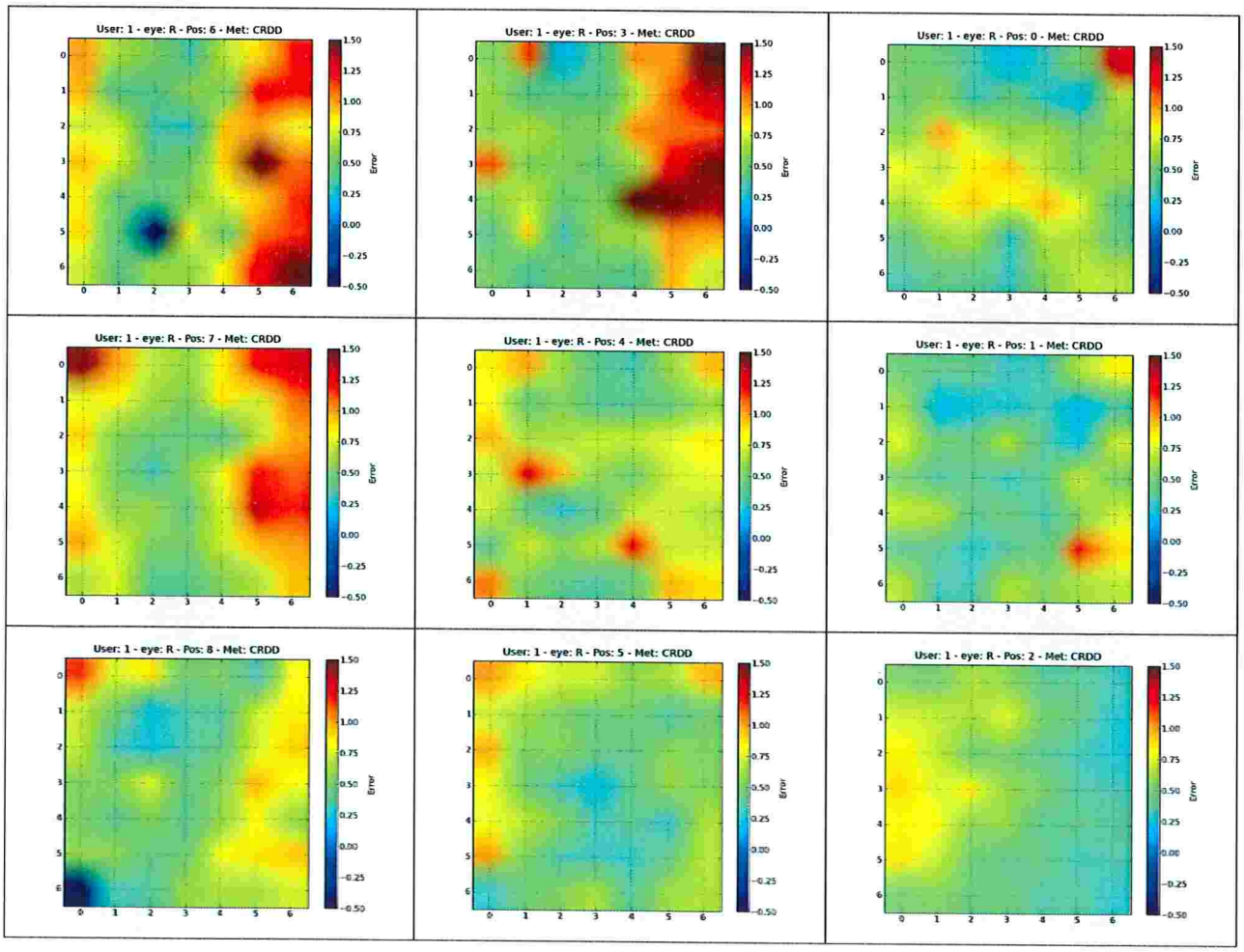

Figure B.38: *Distribution of the gaze error in all positions for the right eye of participant 1 using method* ***CR-DD****. Position 0 was used for calibration.*

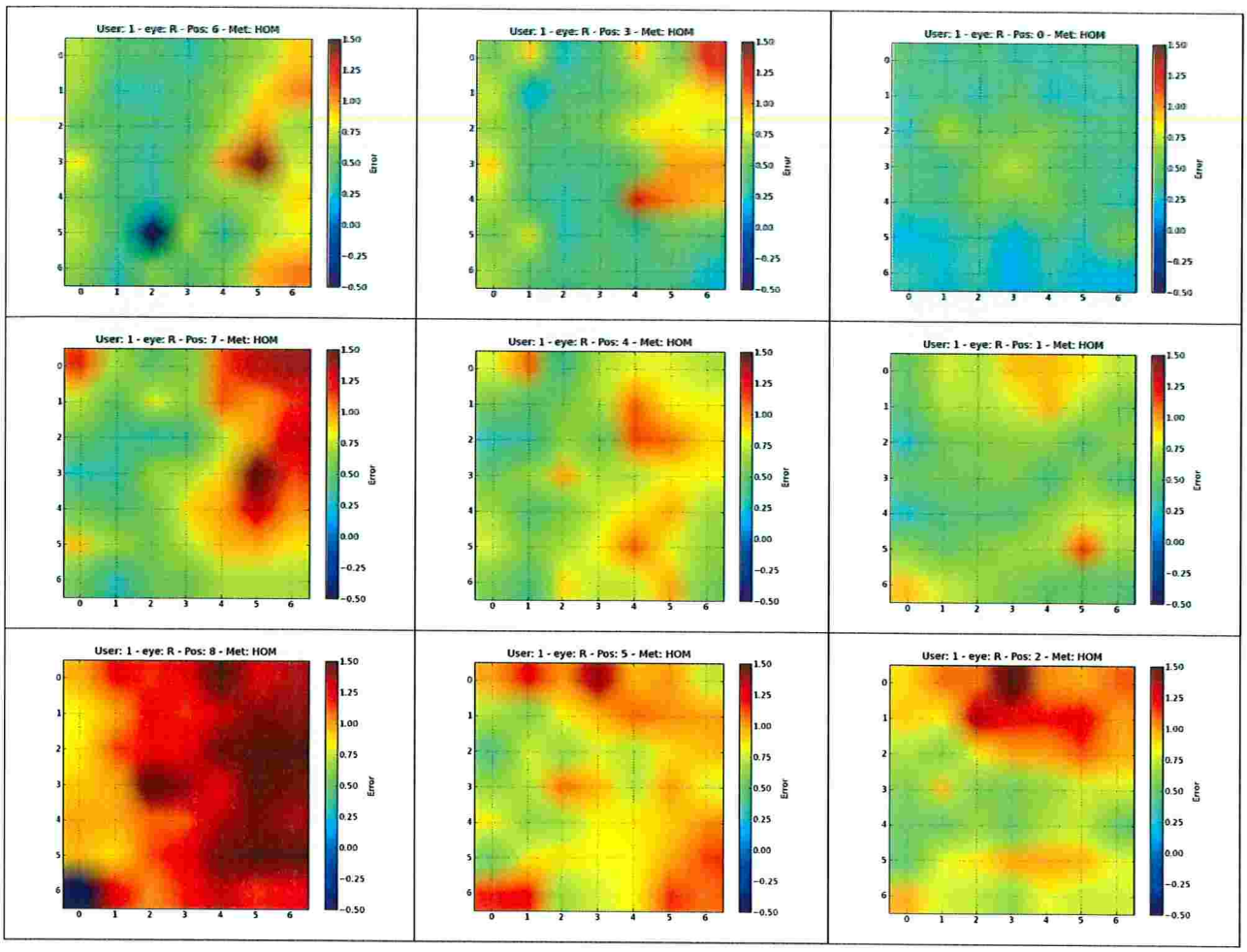

Figure B.39: *Distribution of the gaze error in all positions for the right eye of participant 1 using method* ***HOM****. Position 0 was used for calibration.*

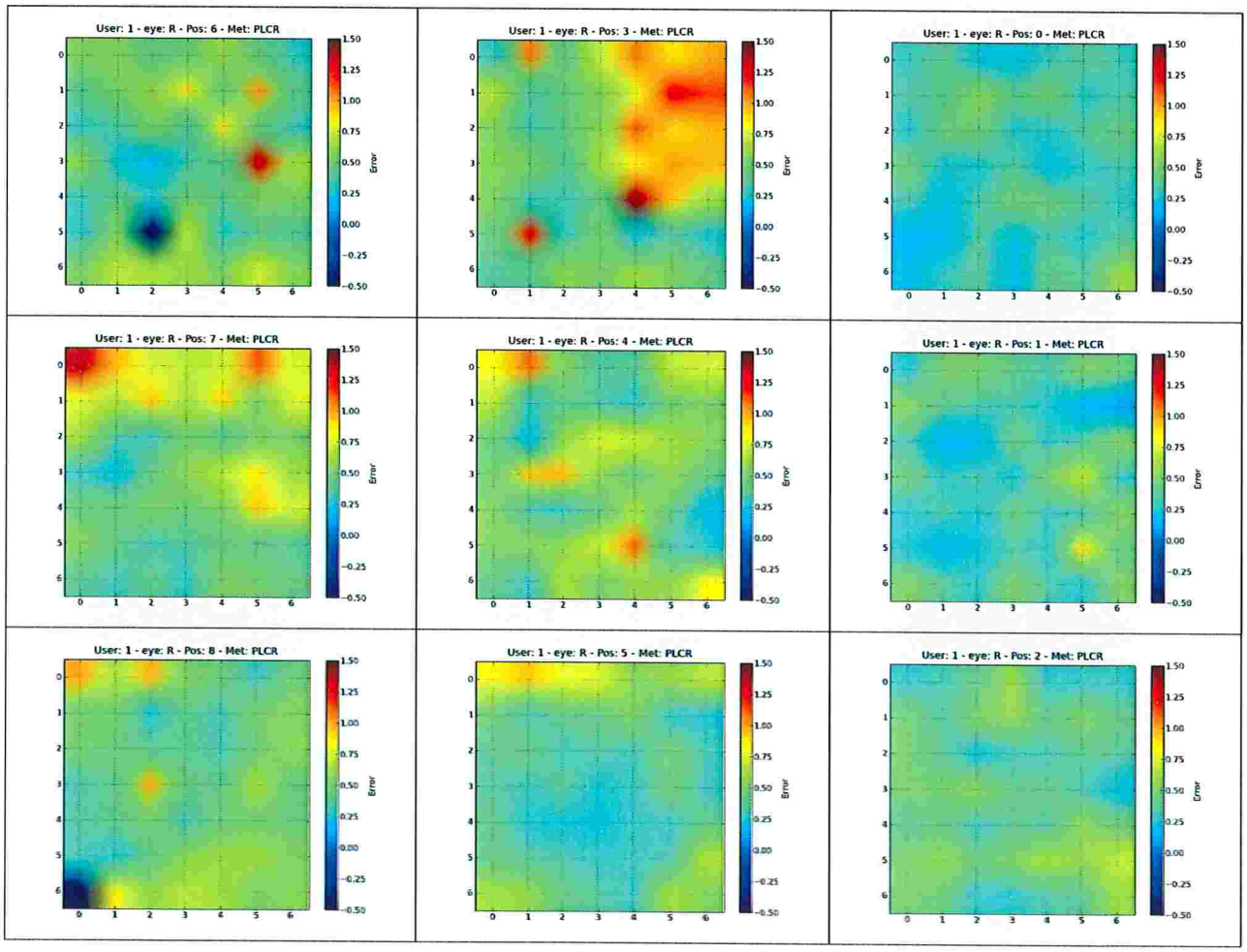

Figure B.40: *Distribution of the gaze error in all positions for the right eye of participant 1 using method* ***PL-CR****. Position 0 was used for calibration.*

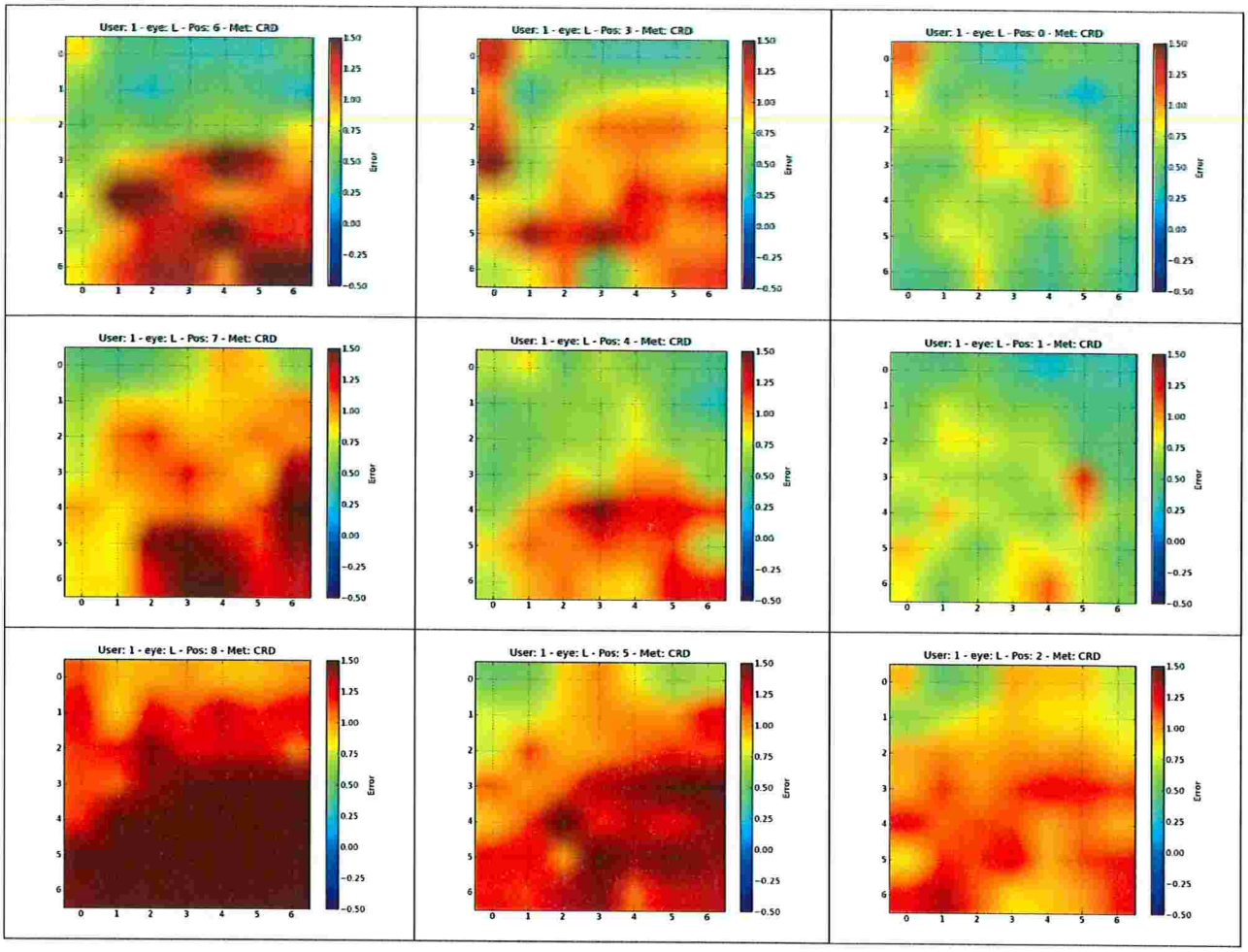

Figure B.41: *Distribution of the gaze error in all positions for the left eye of participant 1 using method* ***CR-D****. Position 0 was used for calibration.*

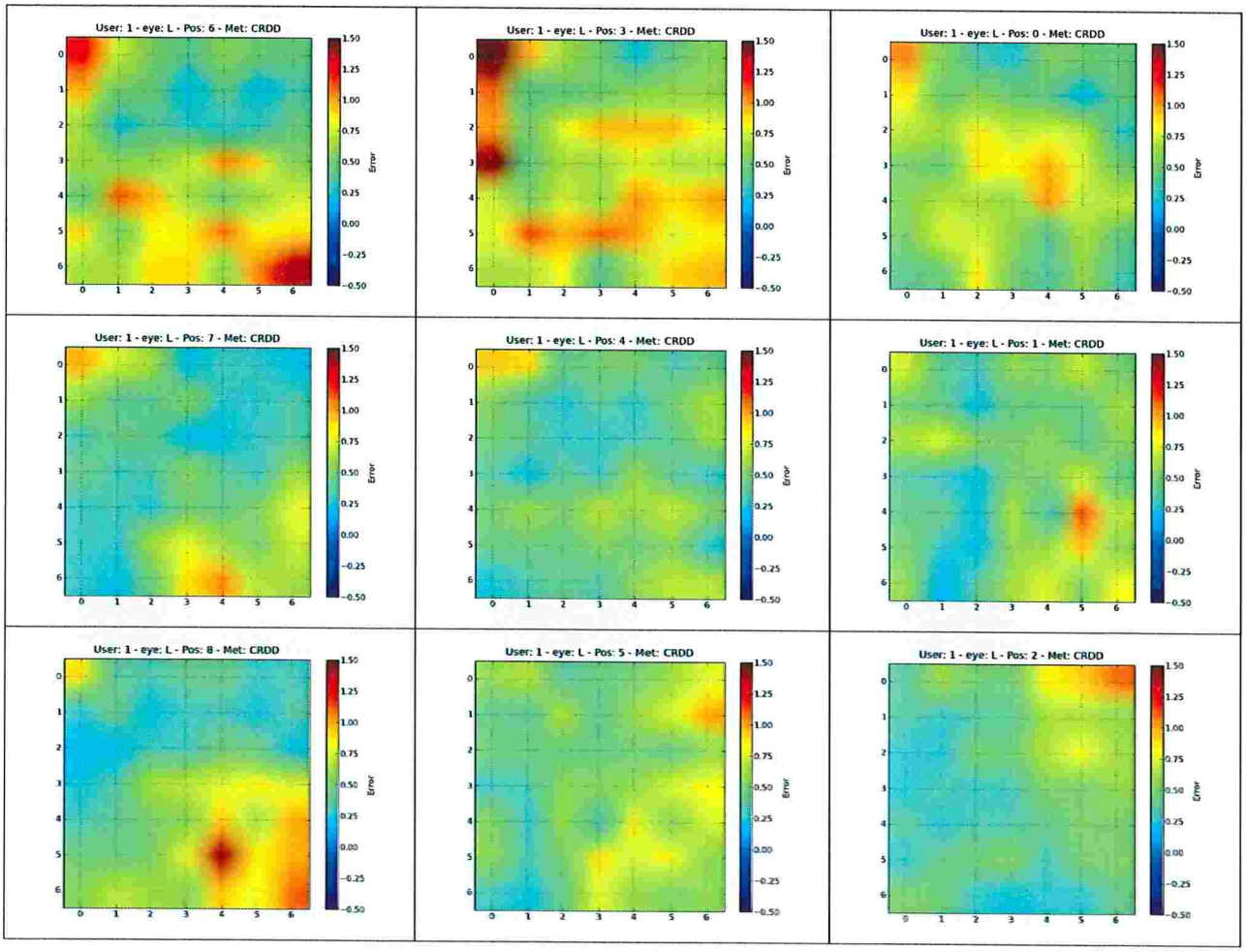

Figure B.42: *Distribution of the gaze error in all positions for the left eye of participant 1 using method* ***CR-DD****. Position 0 was used for calibration.*

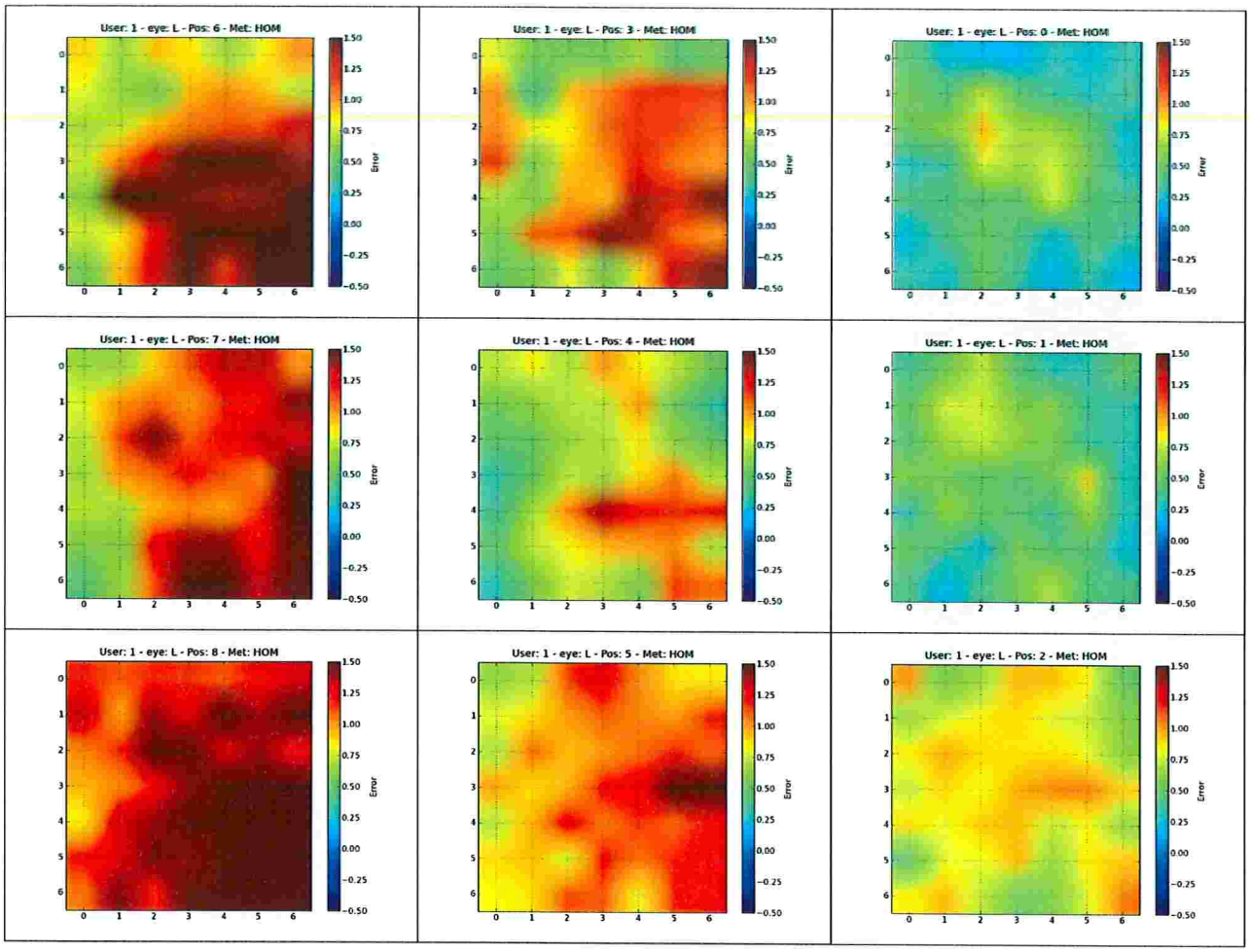

Figure B.43: *Distribution of the gaze error in all positions for the left eye of participant 1 using method* ***HOM****. Position 0 was used for calibration.*

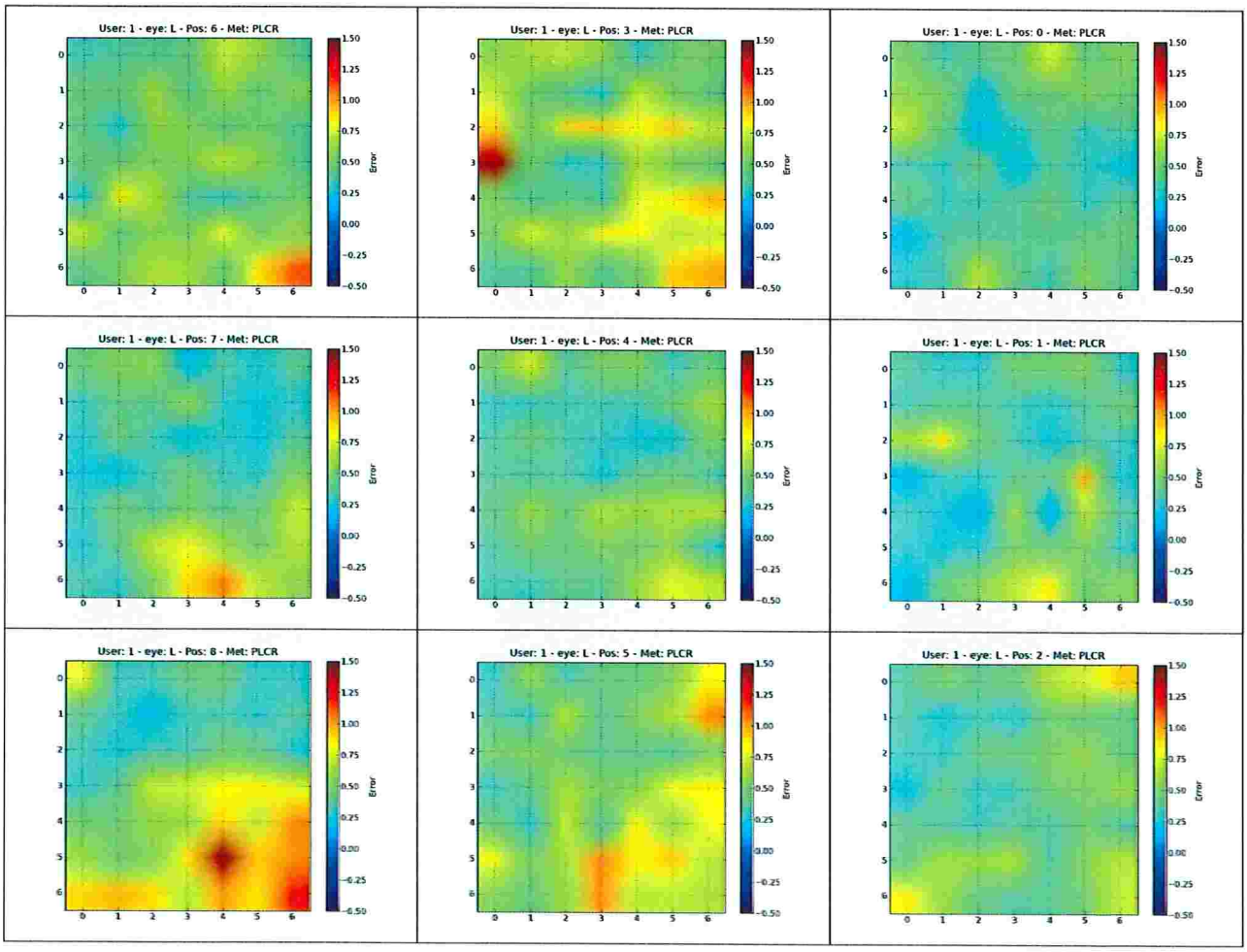

Figure B.44: *Distribution of the gaze error in all positions for the left eye of participant 1 using method* ***PL-CR****. Position 0 was used for calibration.*

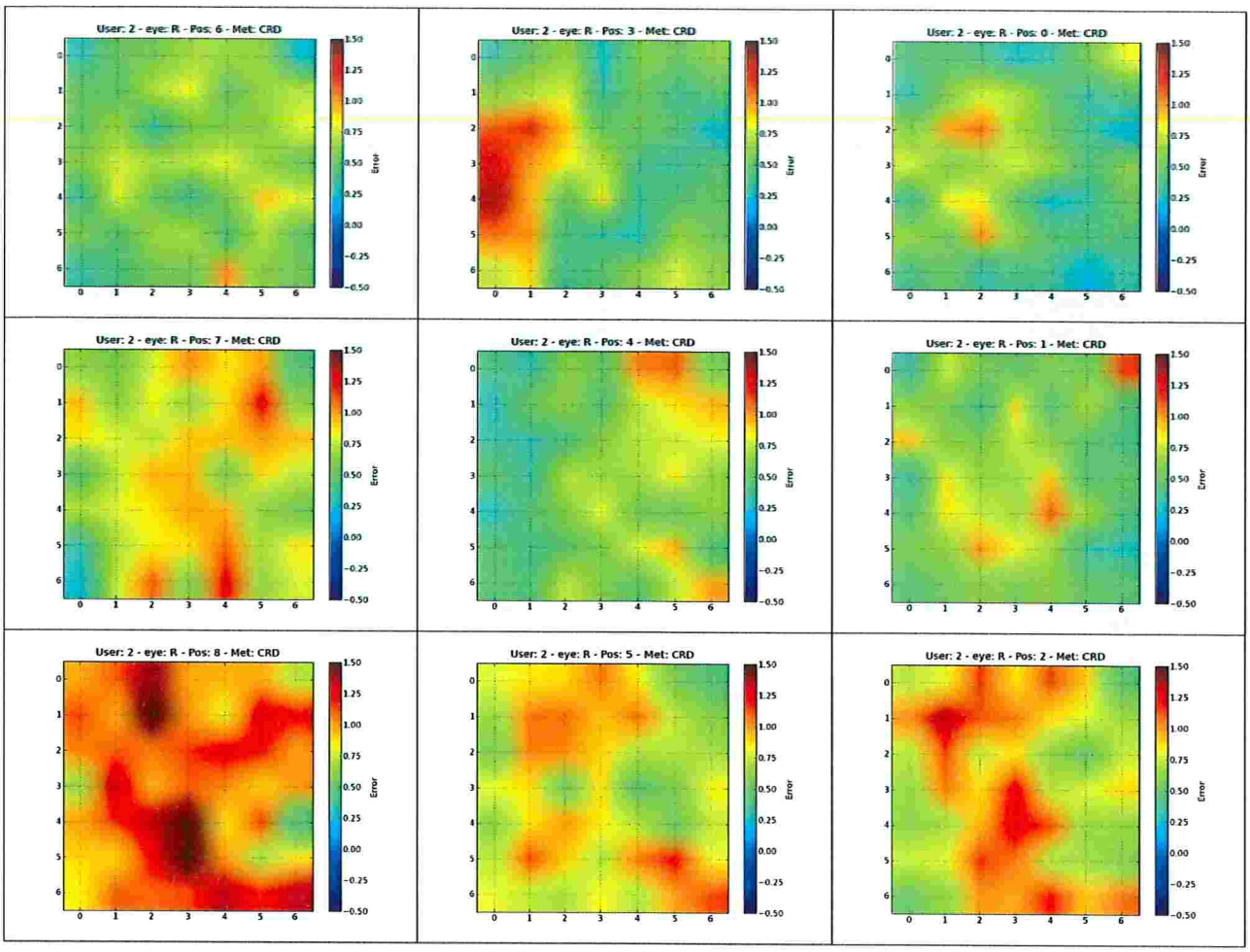

Figure B.45: *Distribution of the gaze error in all positions for the right eye of participant 2 using method* ***CR-D****. Position 0 was used for calibration.*

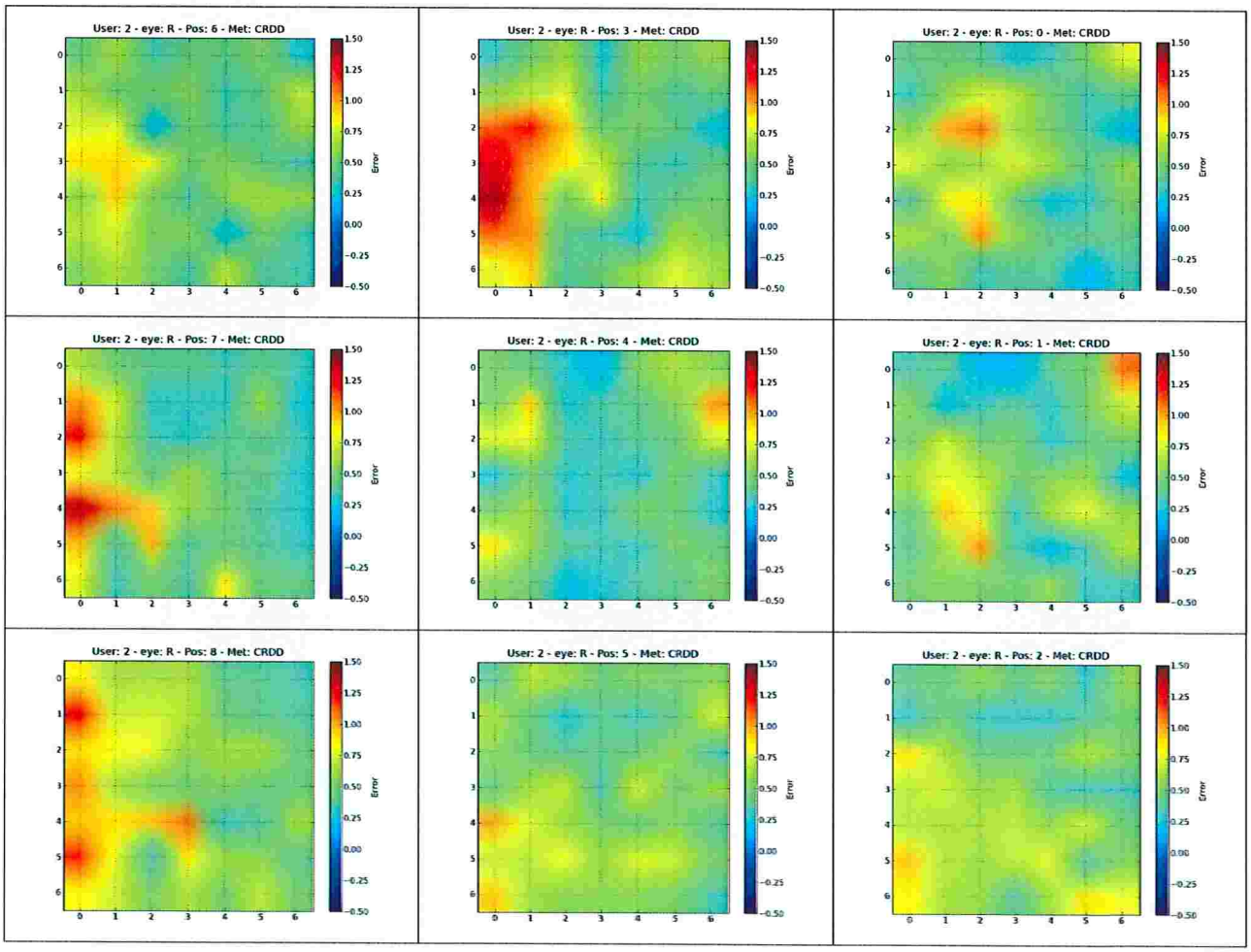

Figure B.46: *Distribution of the gaze error in all positions for the right eye of participant 2 using method* ***CR-DD****. Position 0 was used for calibration.*

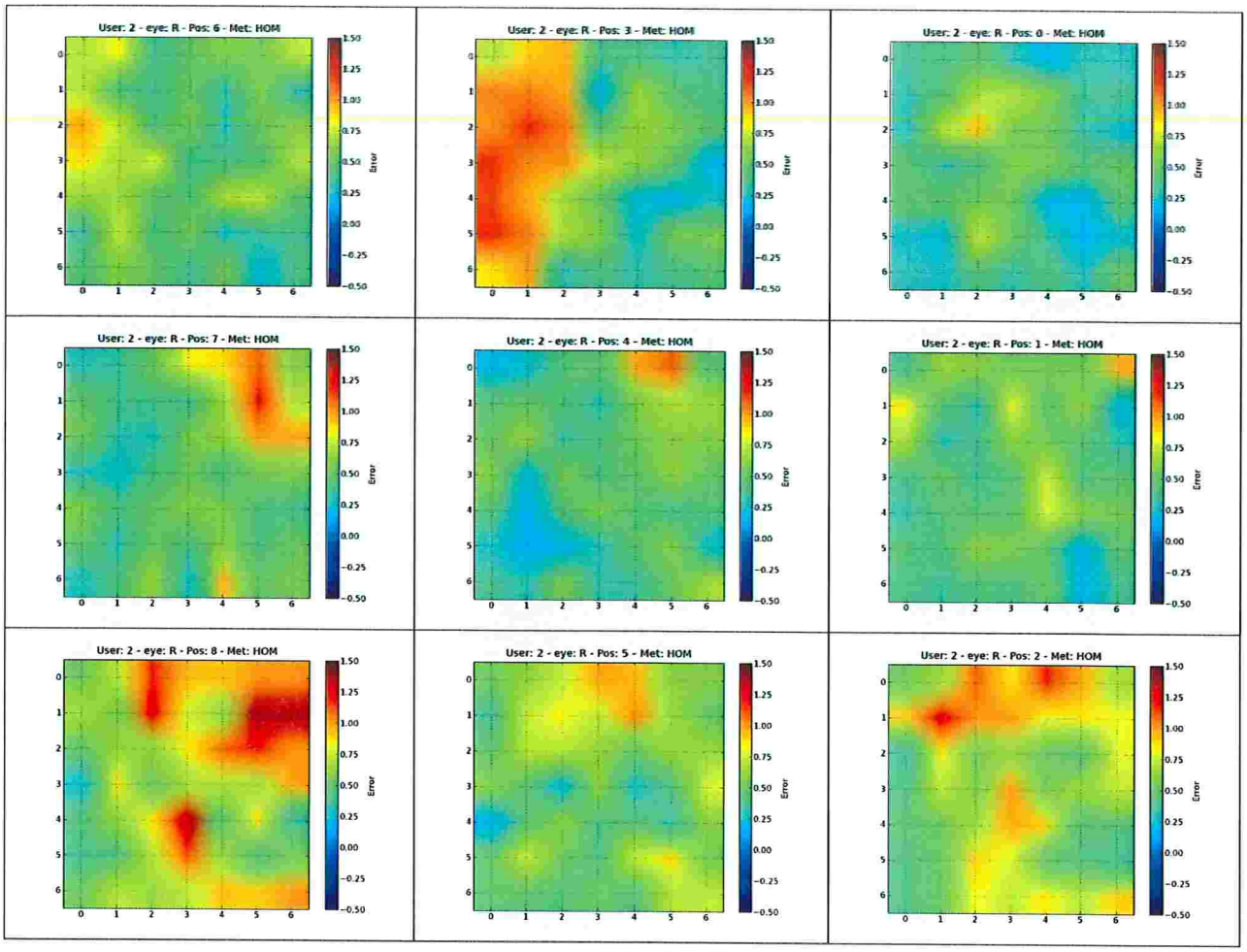

Figure B.47: *Distribution of the gaze error in all positions for the right eye of participant 2 using method* ***HOM****. Position 0 was used for calibration.*

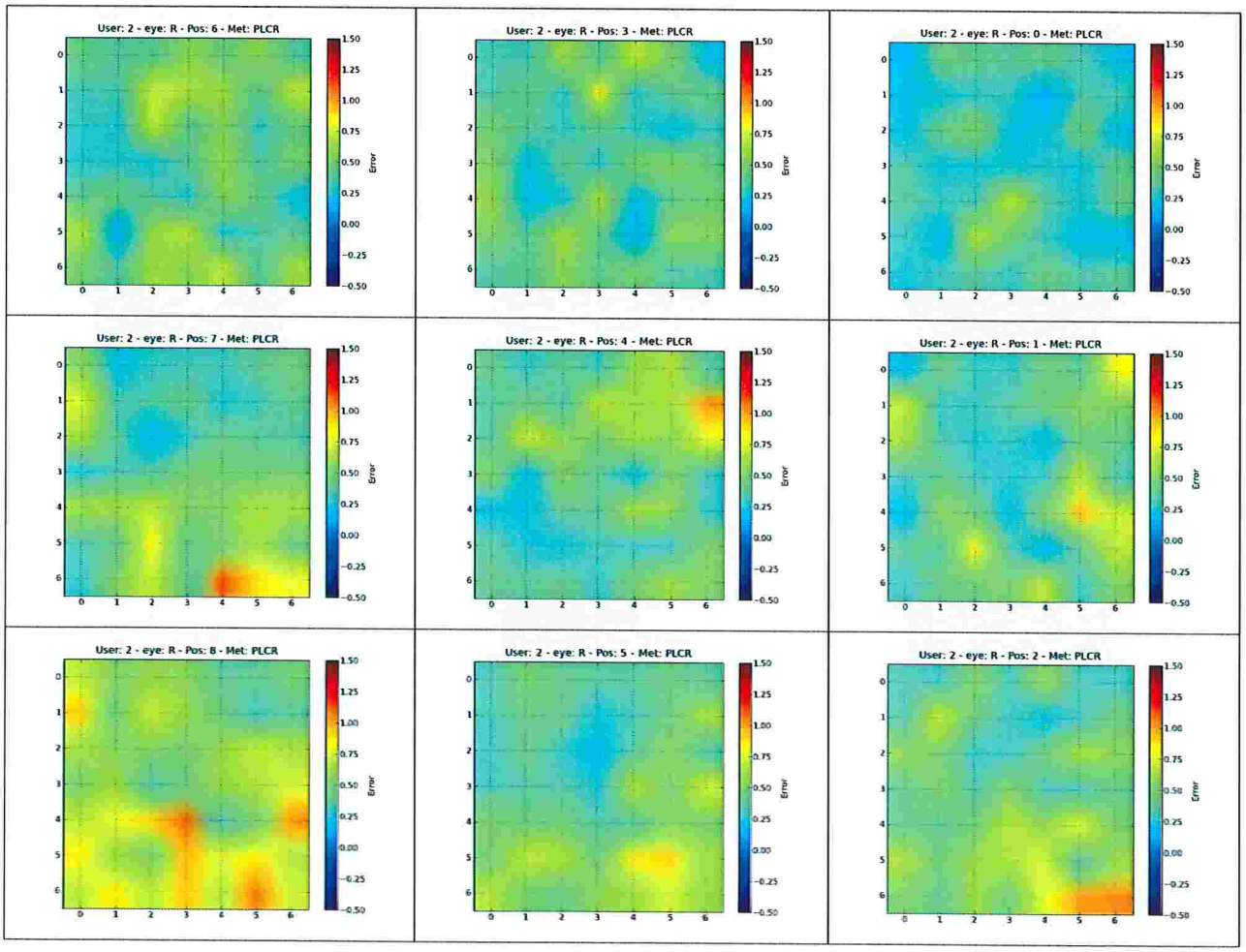

Figure B.48: *Distribution of the gaze error in all positions for the right eye of participant 2 using method* ***PL-CR****. Position 0 was used for calibration.*

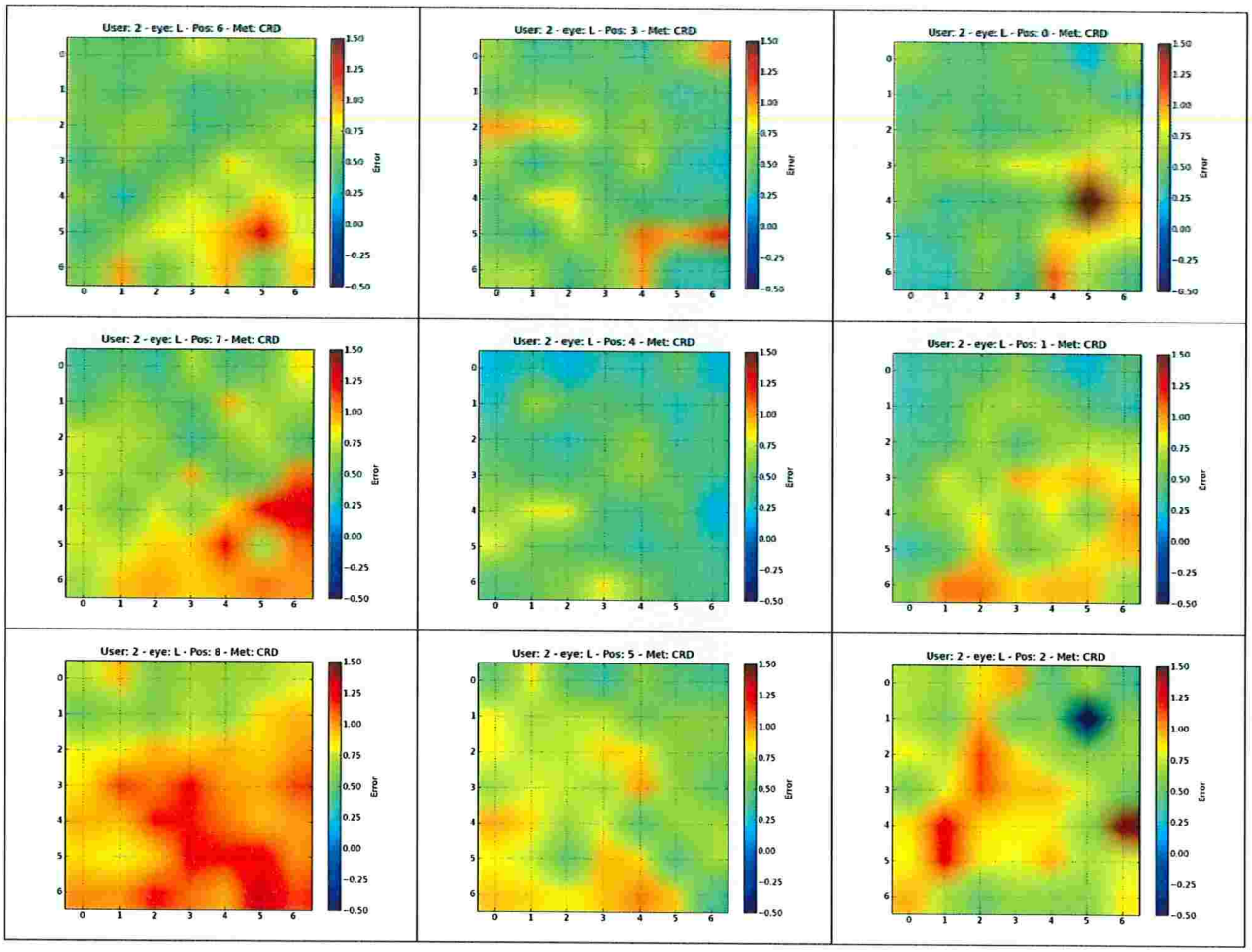

Figure B.49: *Distribution of the gaze error in all positions for the left eye of participant 2 using method* ***CR-D****. Position 0 was used for calibration.*

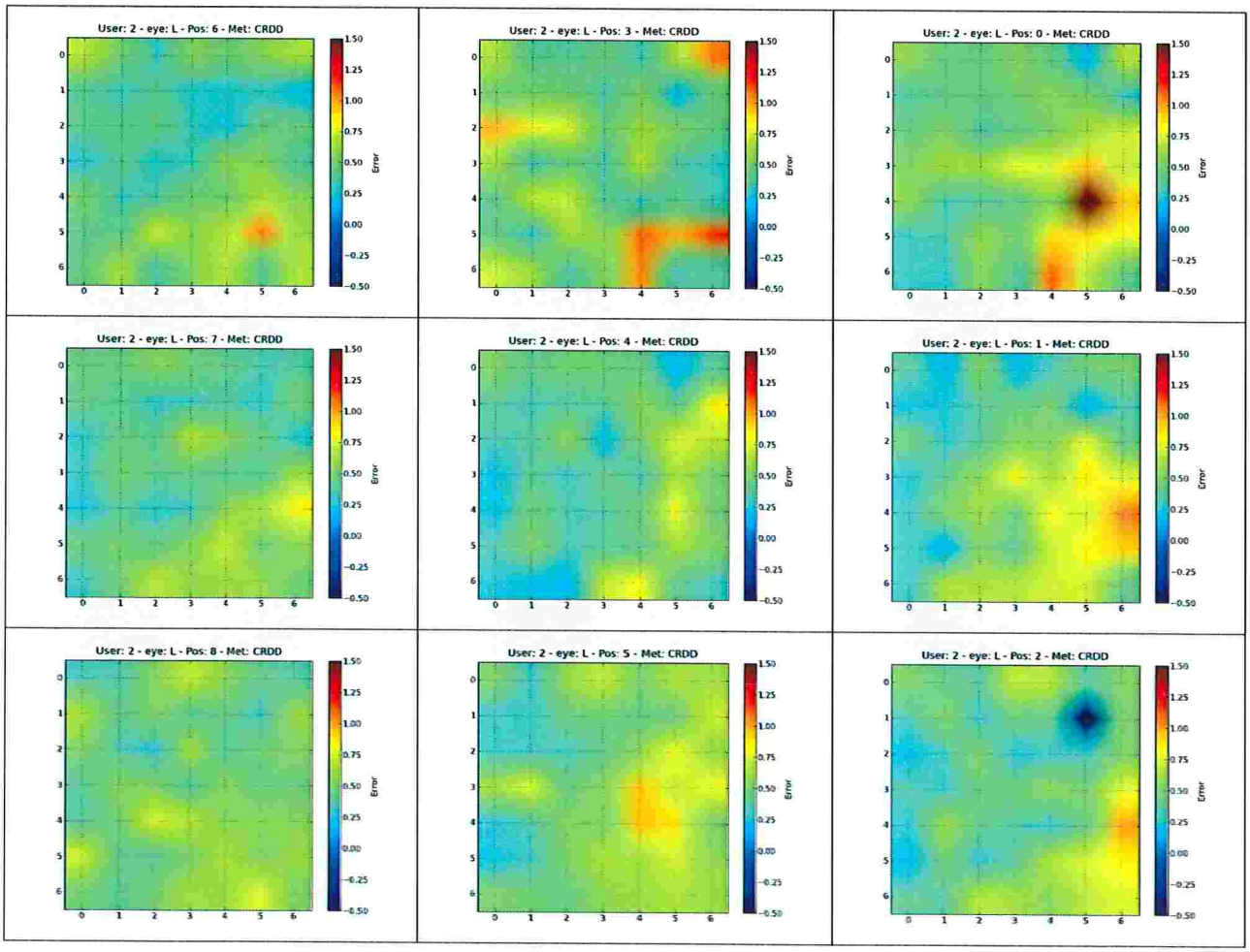

Figure B.50: *Distribution of the gaze error in all positions for the left eye of participant 2 using method* ***CR-DD****. Position 0 was used for calibration.*

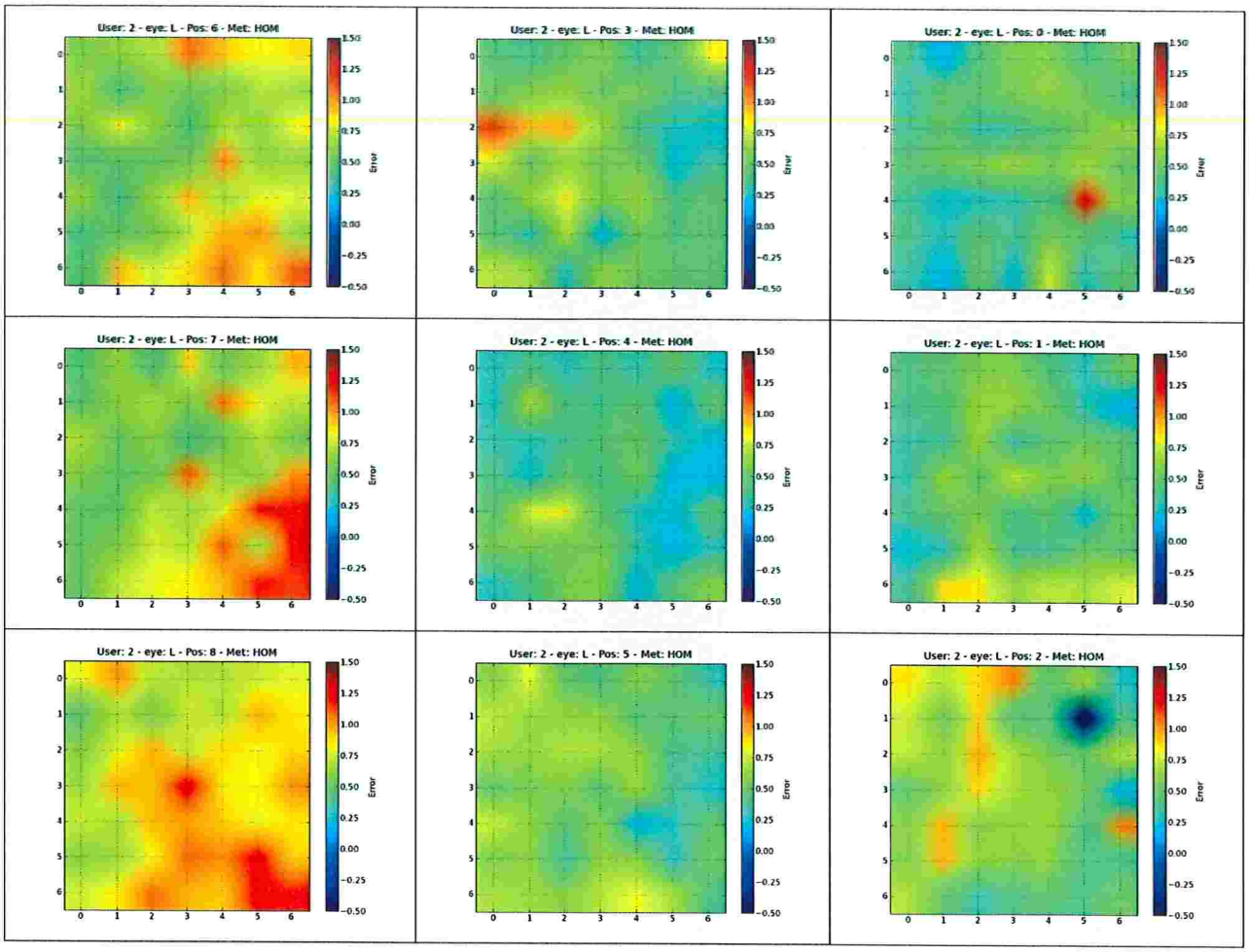

Figure B.51: *Distribution of the gaze error in all positions for the left eye of participant 2 using method* ***HOM****. Position 0 was used for calibration.*

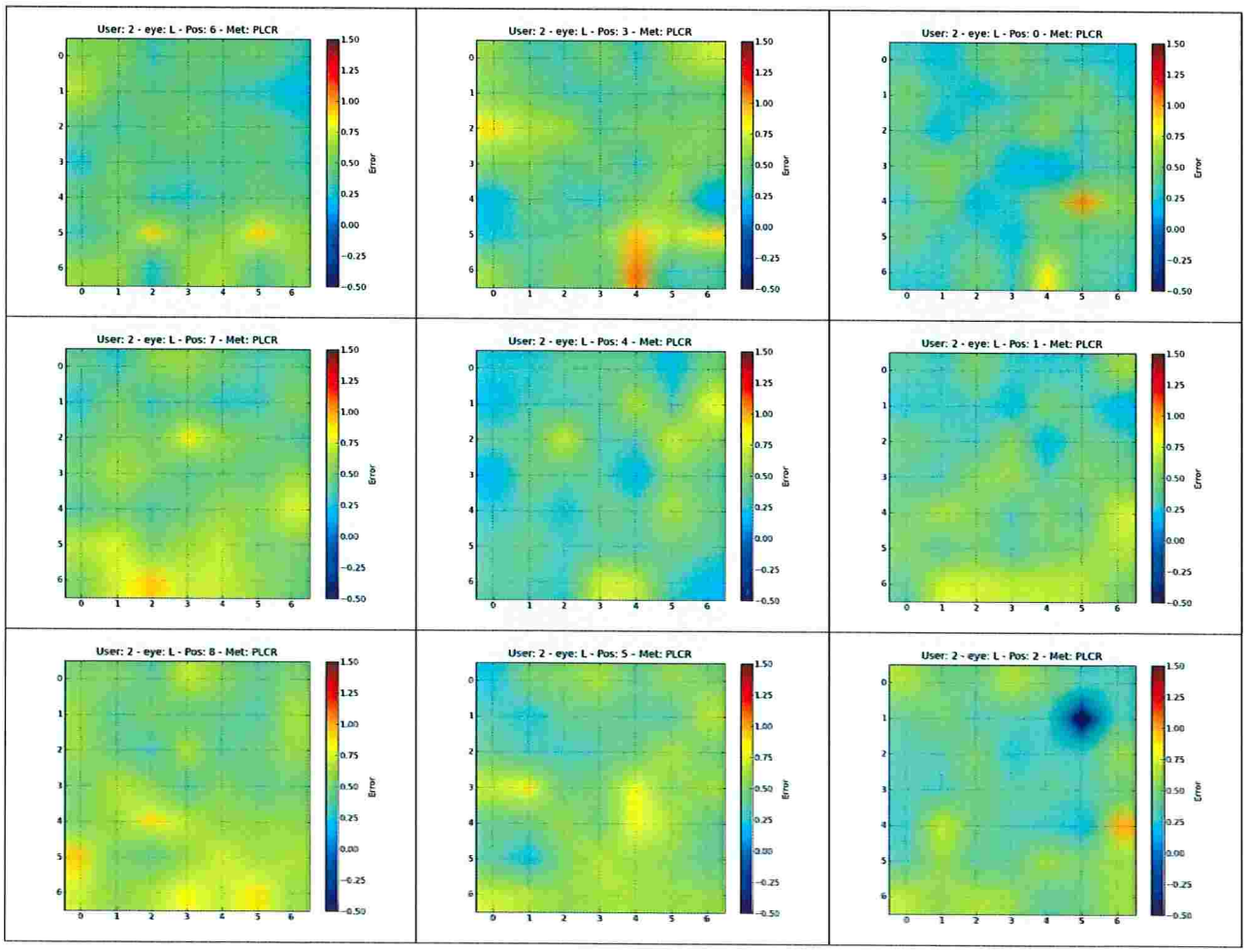

Figure B.52: *Distribution of the gaze error in all positions for the left eye of participant 2 using method* ***PL-CR****. Position 0 was used for calibration.*

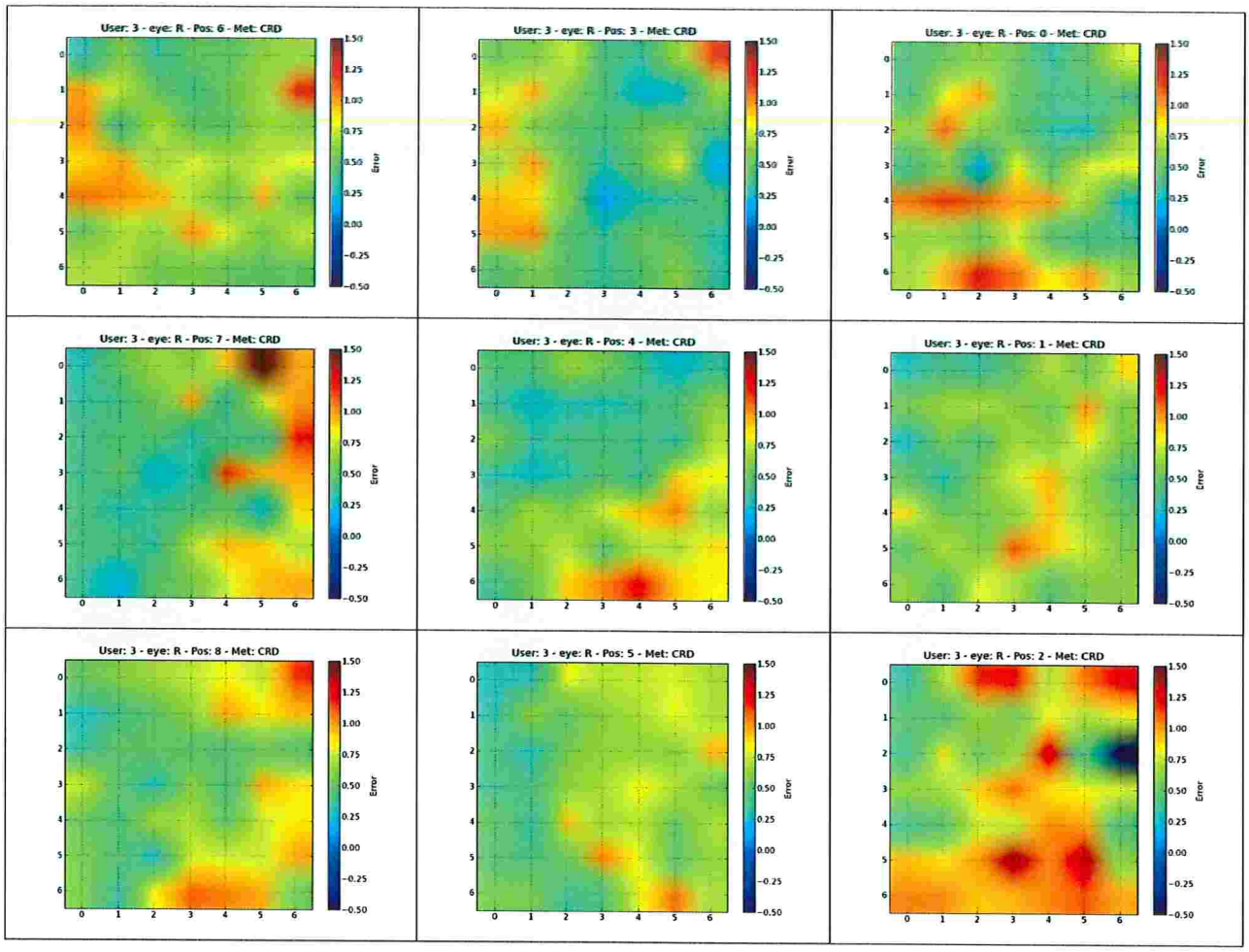

Figure B.53: *Distribution of the gaze error in all positions for the right eye of participant 3 using method* ***CR-D****. Position 0 was used for calibration.*

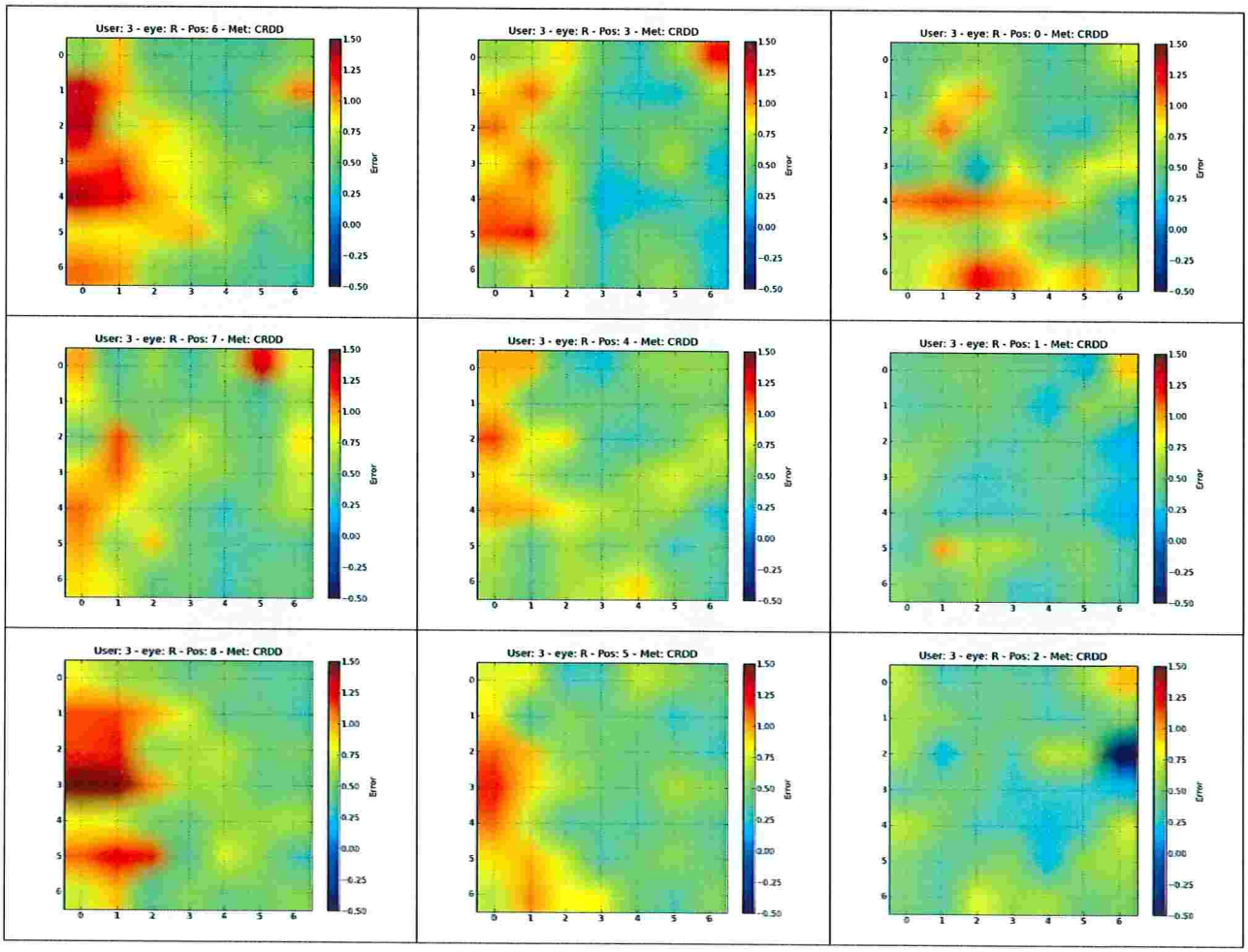

Figure B.54: *Distribution of the gaze error in all positions for the right eye of participant 3 using method* ***CR-DD****. Position 0 was used for calibration.*

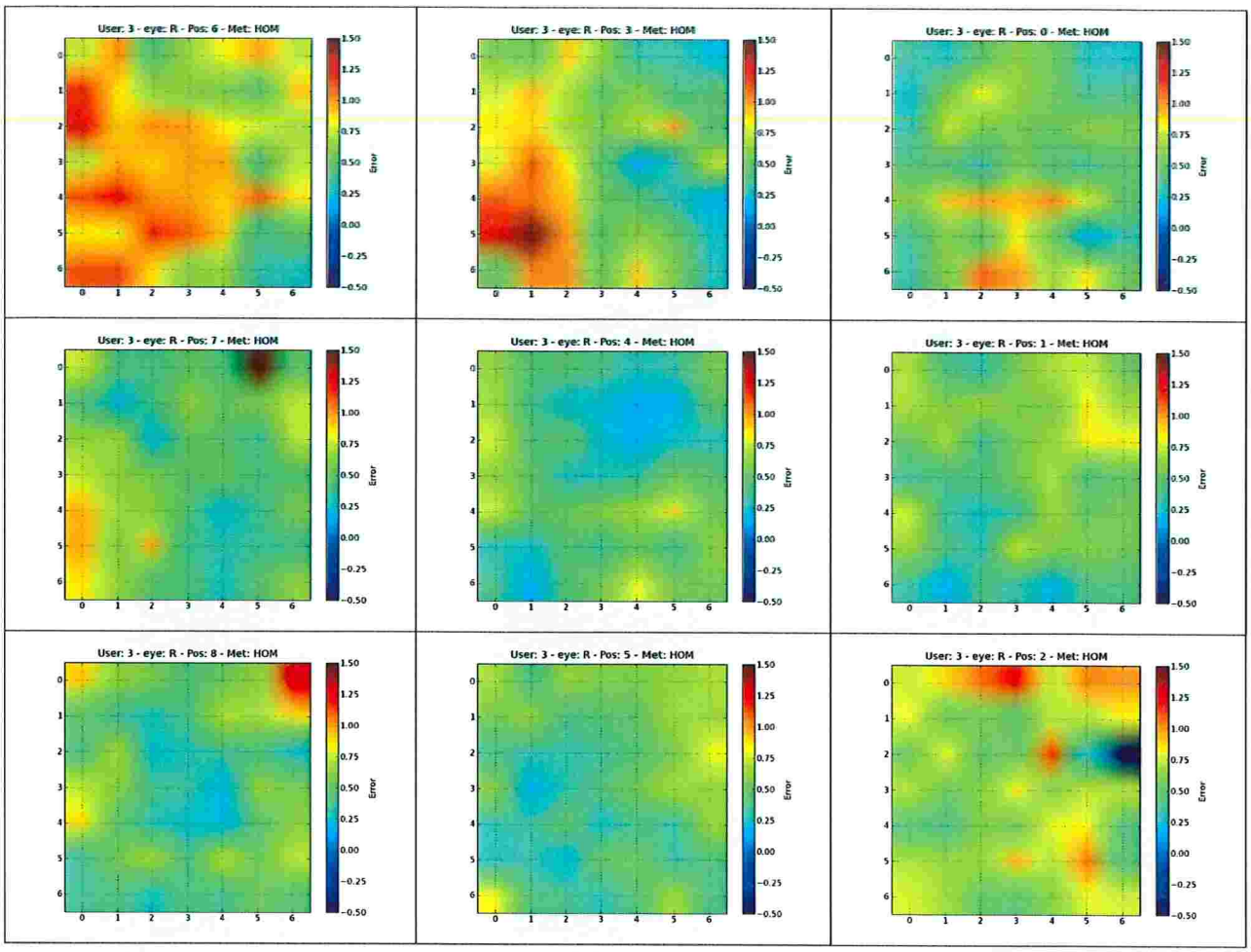

Figure B.55: *Distribution of the gaze error in all positions for the right eye of participant 3 using method* ***HOM****. Position 0 was used for calibration.*

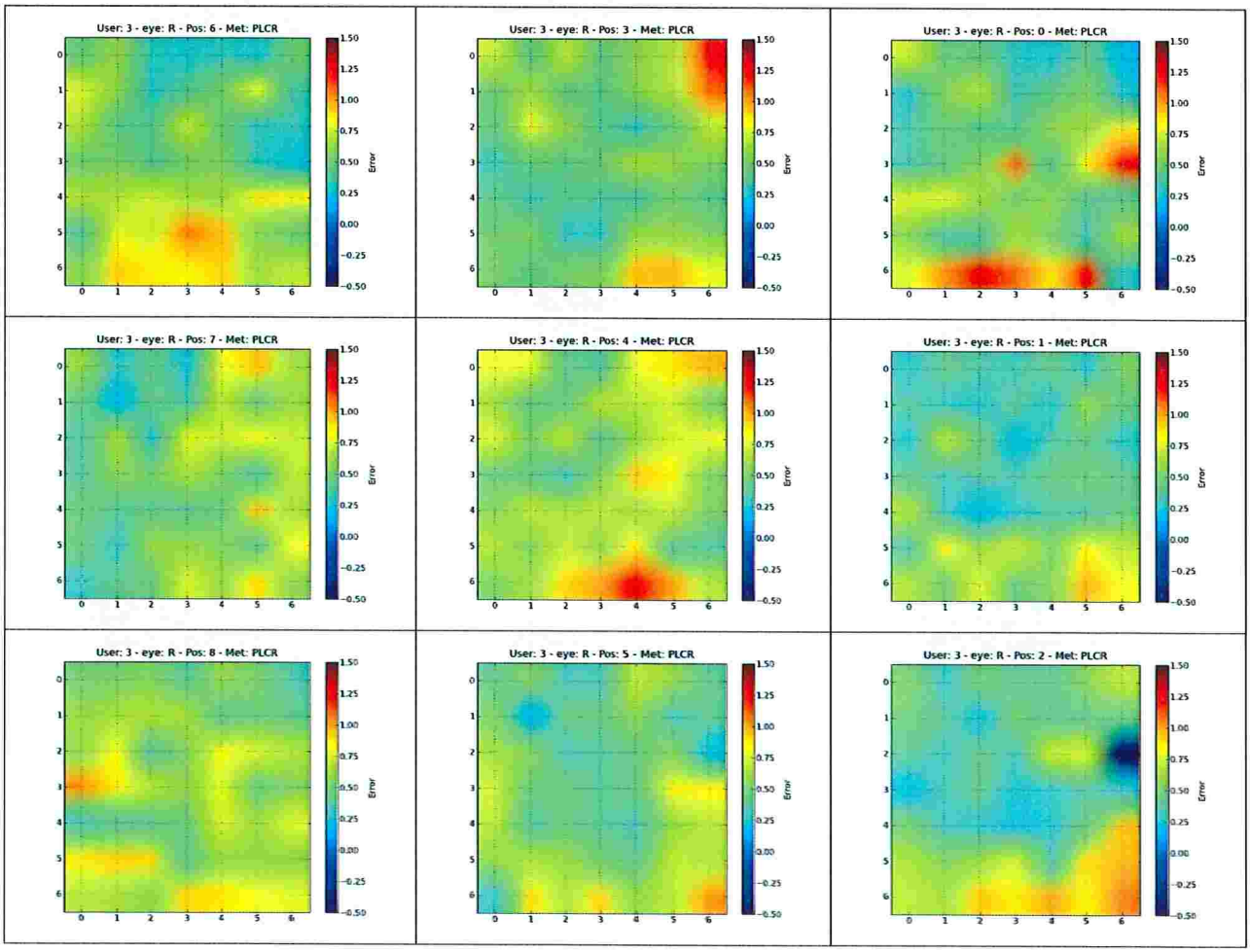

Figure B.56: *Distribution of the gaze error in all positions for the right eye of participant 3 using method* ***PL-CR****. Position 0 was used for calibration.*

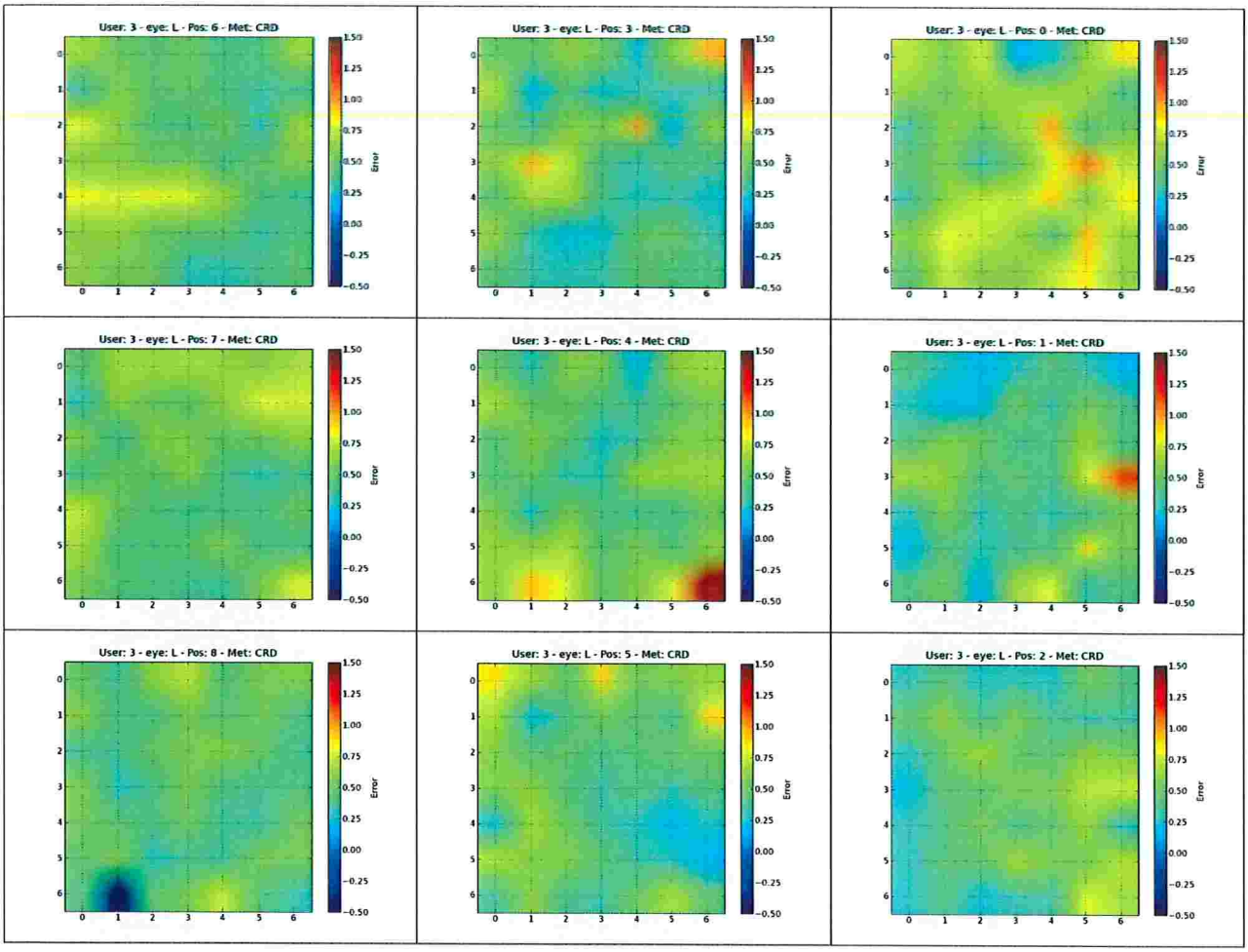

Figure B.57: *Distribution of the gaze error in all positions for the left eye of participant 3 using method* ***CR-D****. Position 0 was used for calibration.*

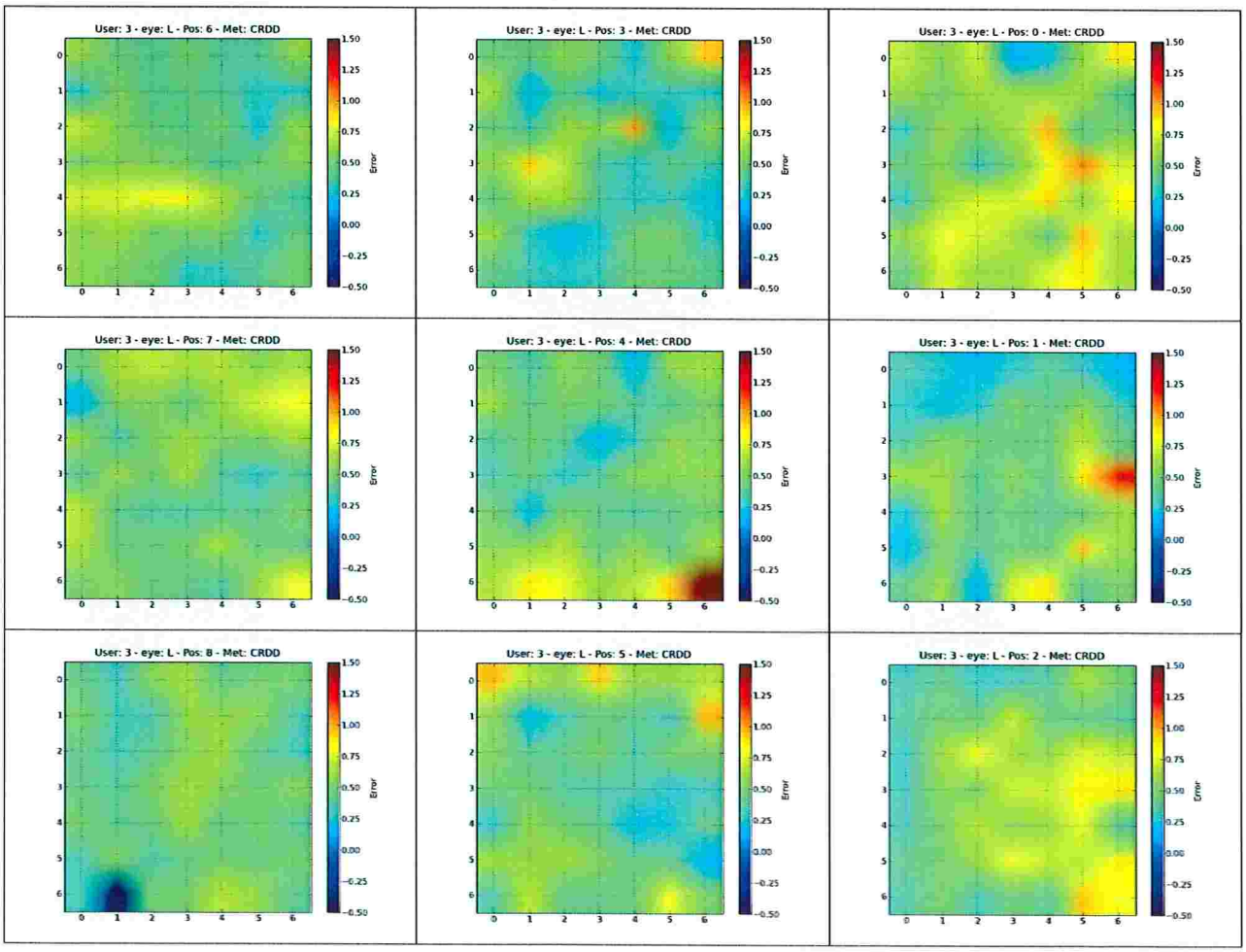

Figure B.58: *Distribution of the gaze error in all positions for the left eye of participant 3 using method* ***CR-DD****. Position 0 was used for calibration.*

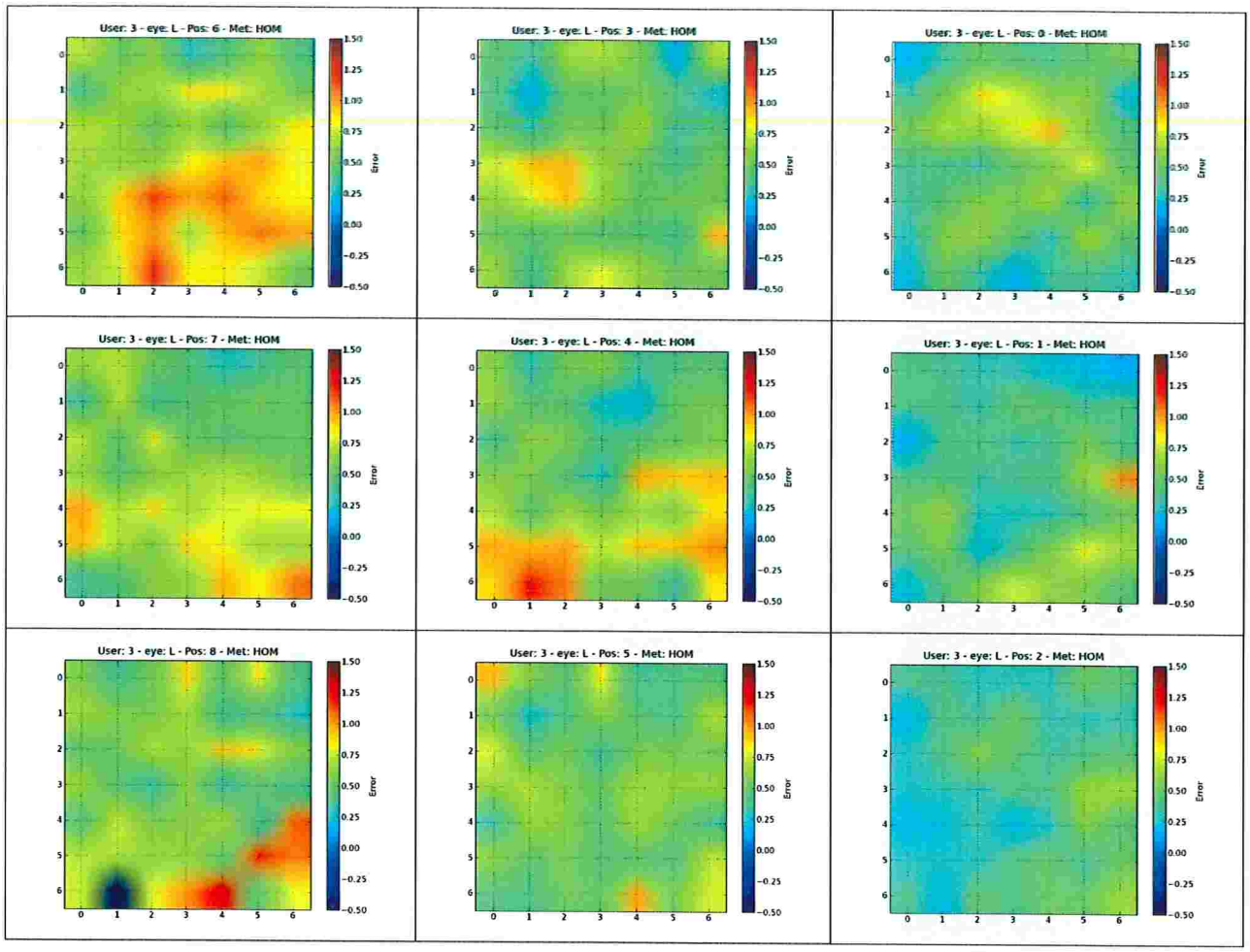

Figure B.59: *Distribution of the gaze error in all positions for the left eye of participant 3 using method* ***HOM****. Position 0 was used for calibration.*

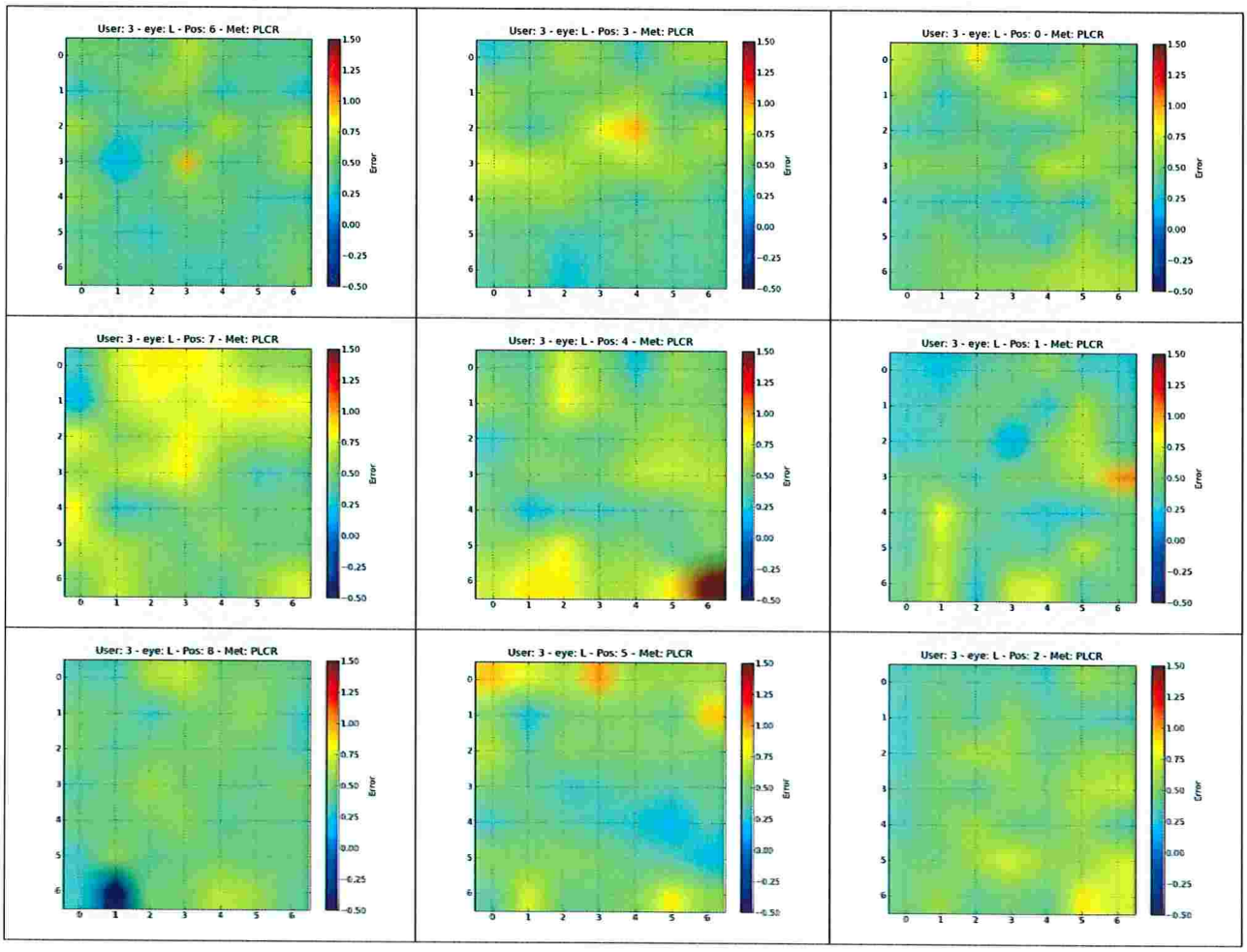

Figure B.60: *Distribution of the gaze error in all positions for the left eye of participant 3 using method* ***PL-CR****. Position 0 was used for calibration.*

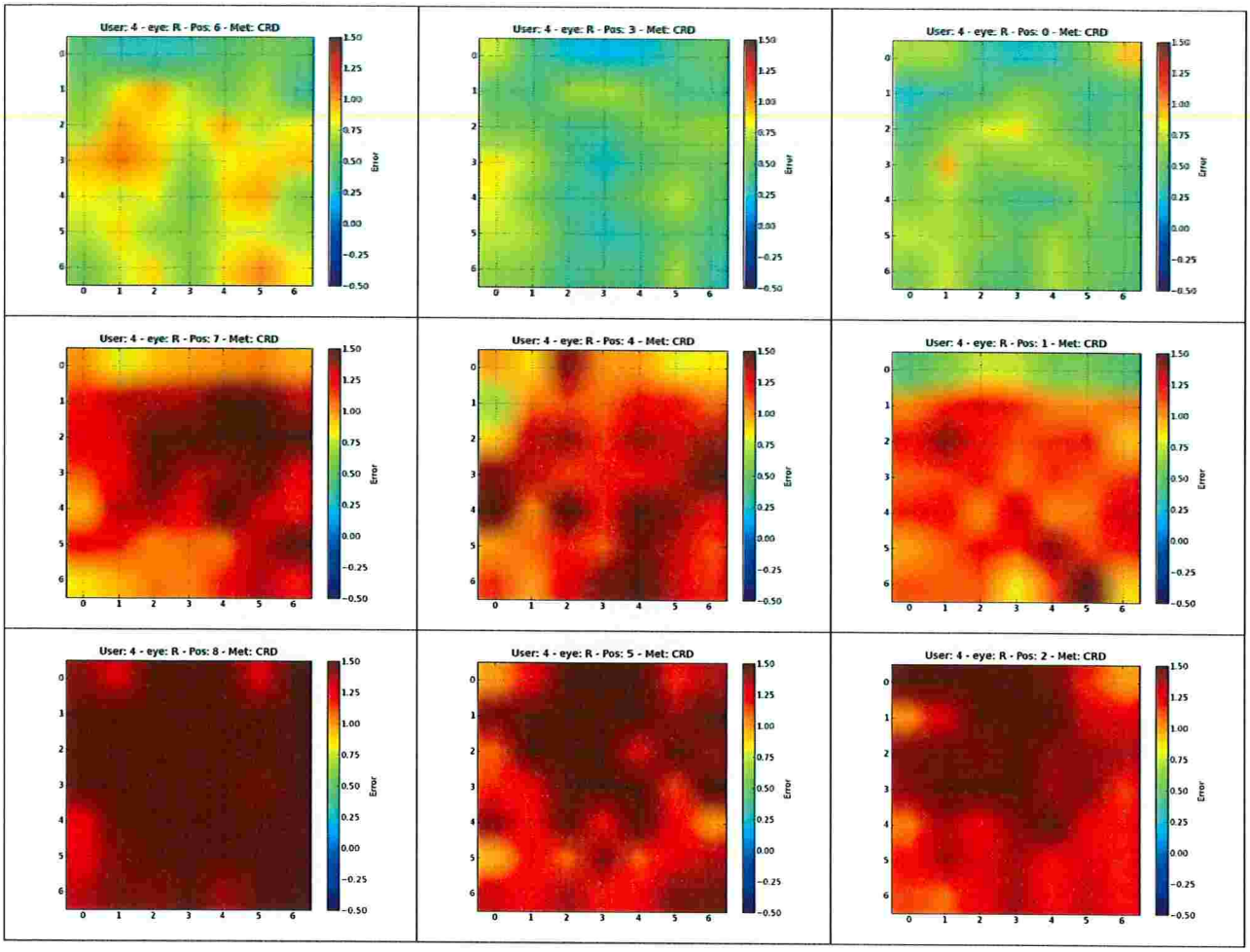

Figure B.61: *Distribution of the gaze error in all positions for the right eye of participant 4 using method* ***CR-D****. Position 0 was used for calibration.*

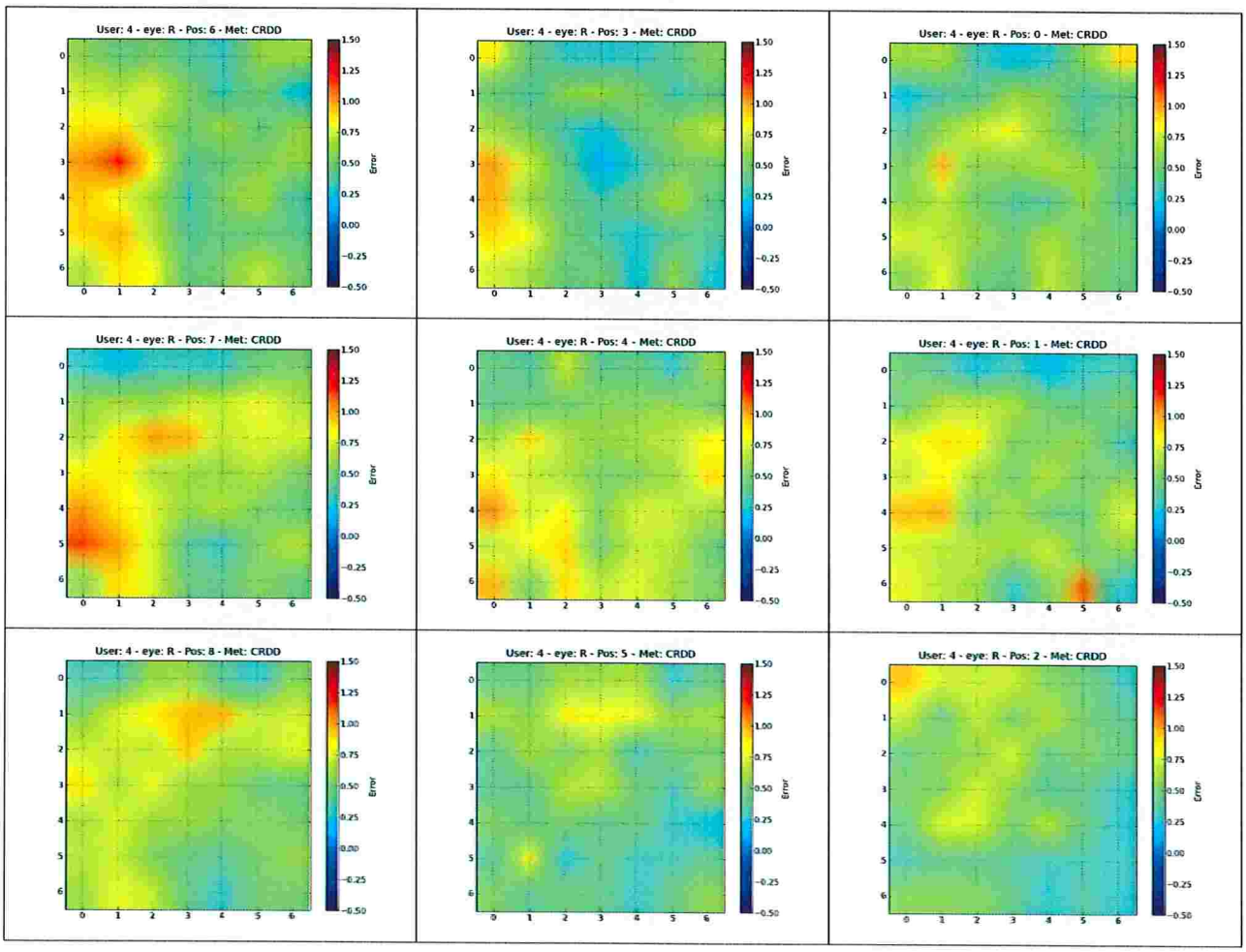

Figure B.62: *Distribution of the gaze error in all positions for the right eye of participant 4 using method* ***CR-DD****. Position 0 was used for calibration.*

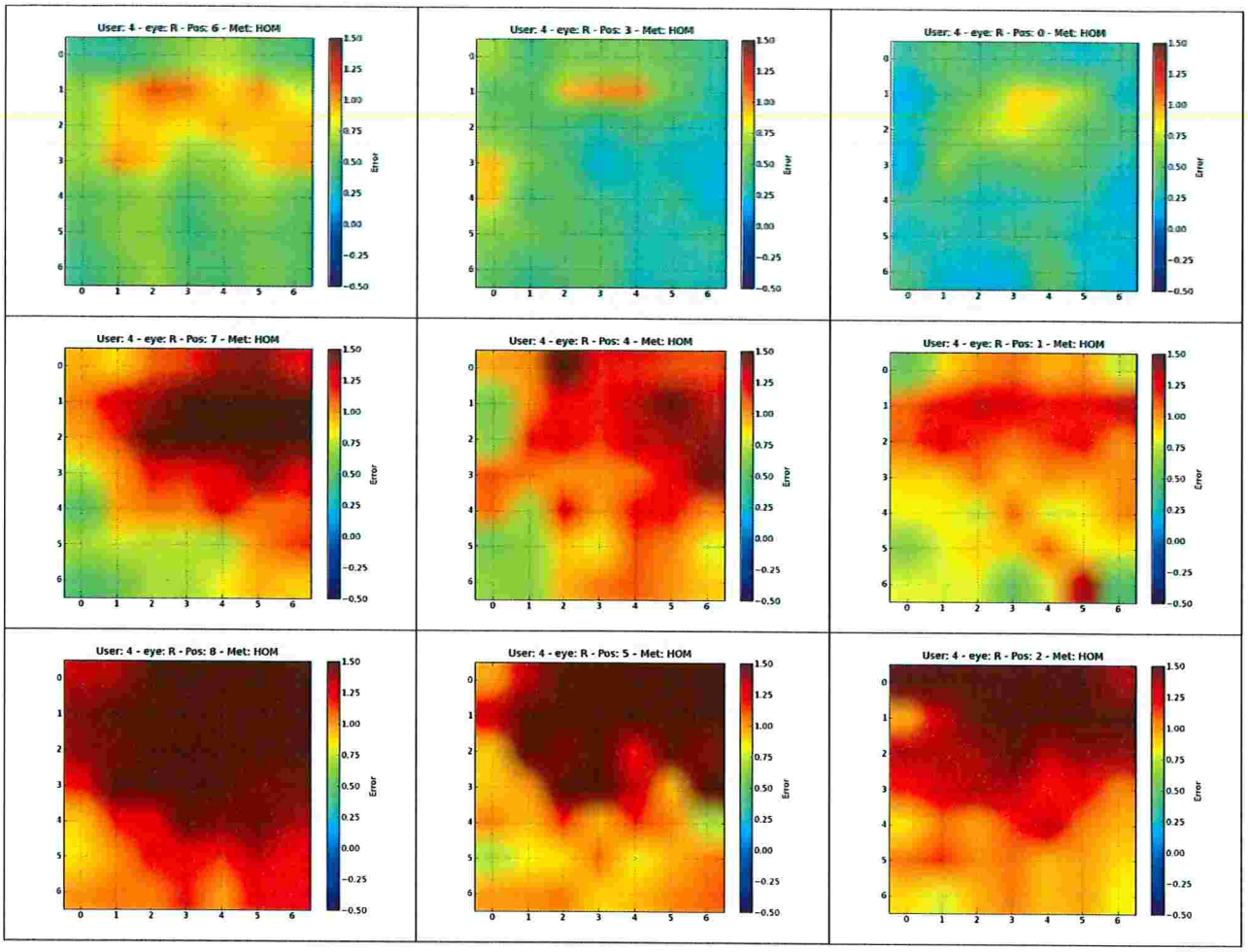

Figure B.63: *Distribution of the gaze error in all positions for the right eye of participant 4 using method* ***HOM****. Position 0 was used for calibration.*

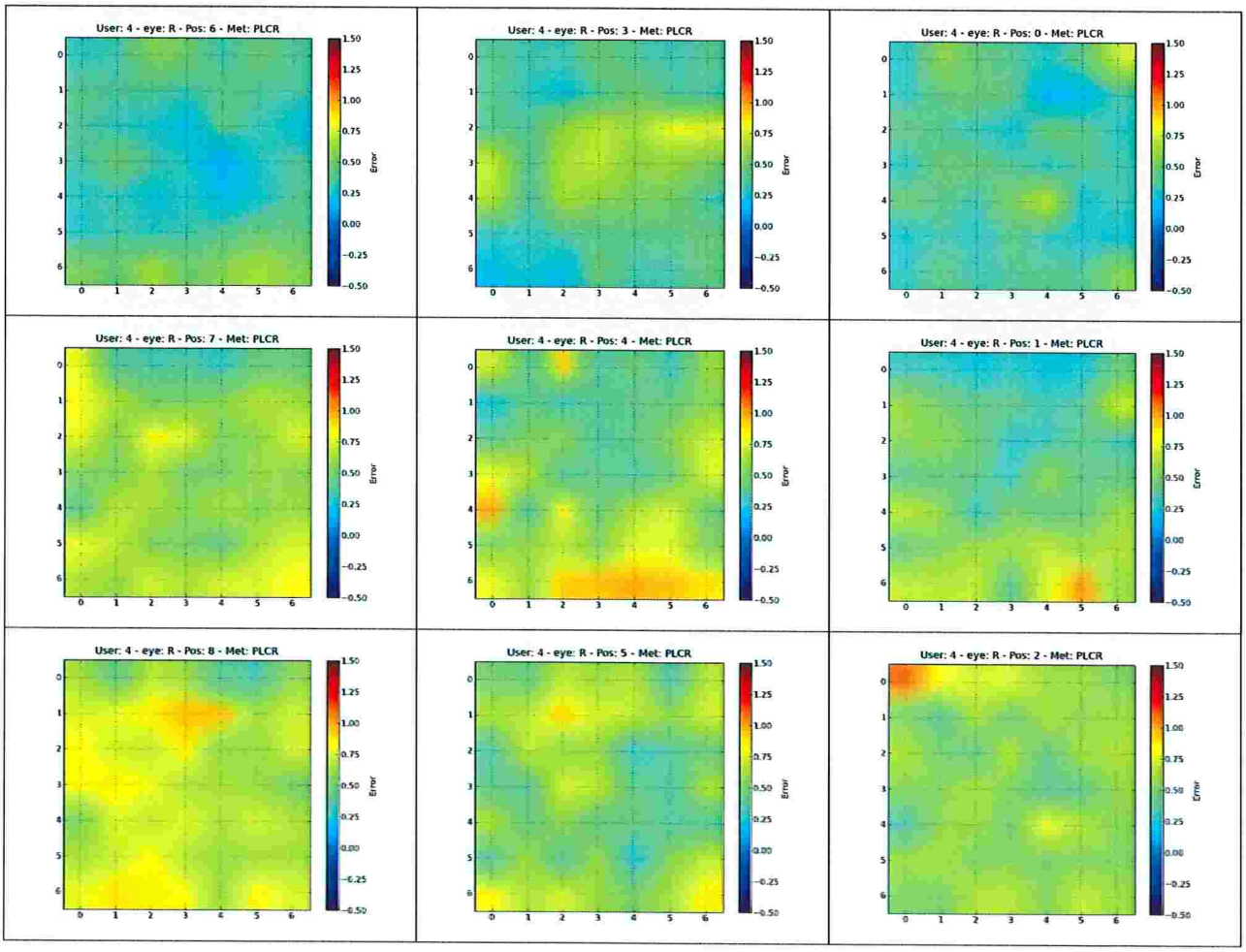

Figure B.64: *Distribution of the gaze error in all positions for the right eye of participant 4 using method* ***PL-CR****. Position 0 was used for calibration.*

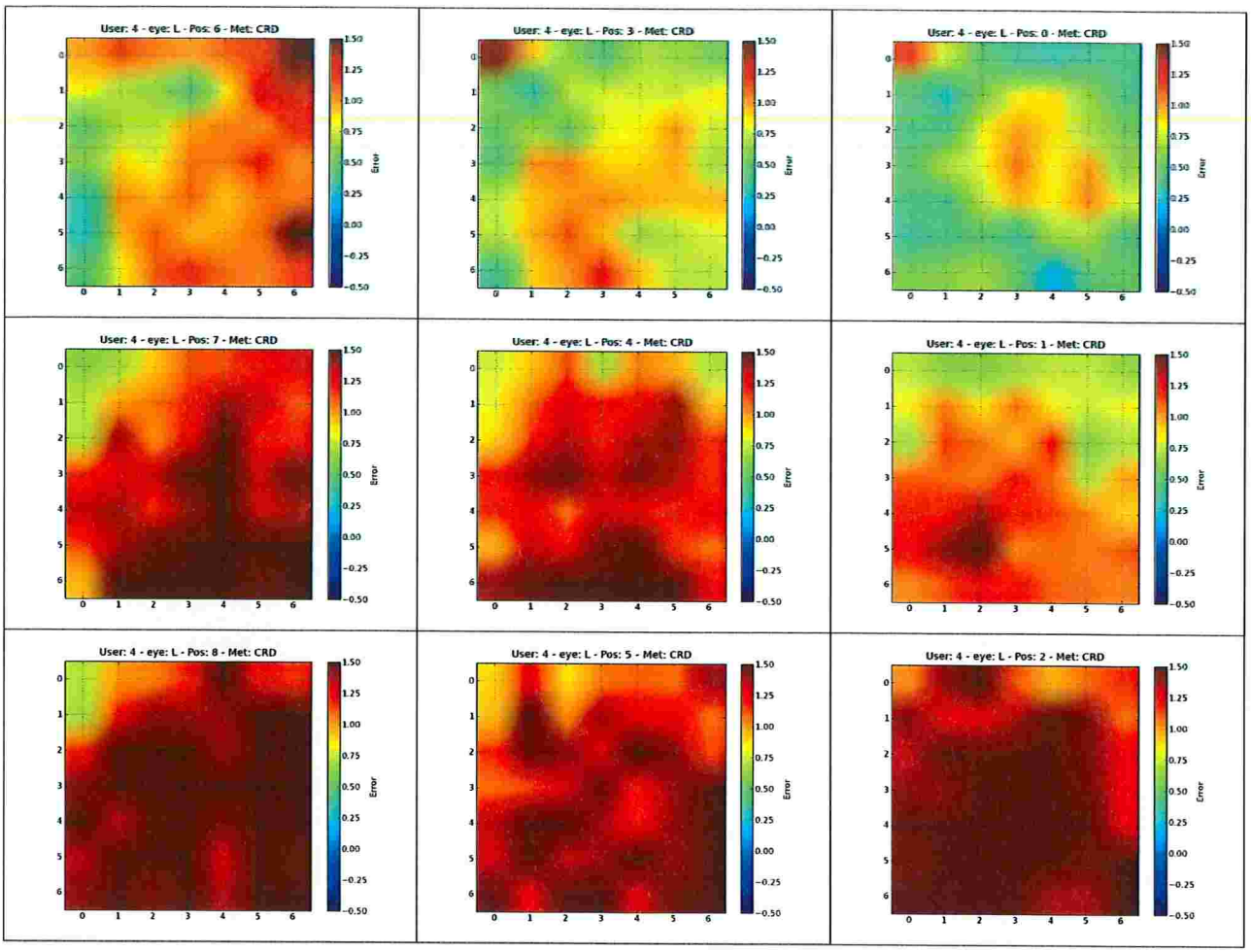

Figure B.65: *Distribution of the gaze error in all positions for the left eye of participant 4 using method* ***CR-D****. Position 0 was used for calibration.*

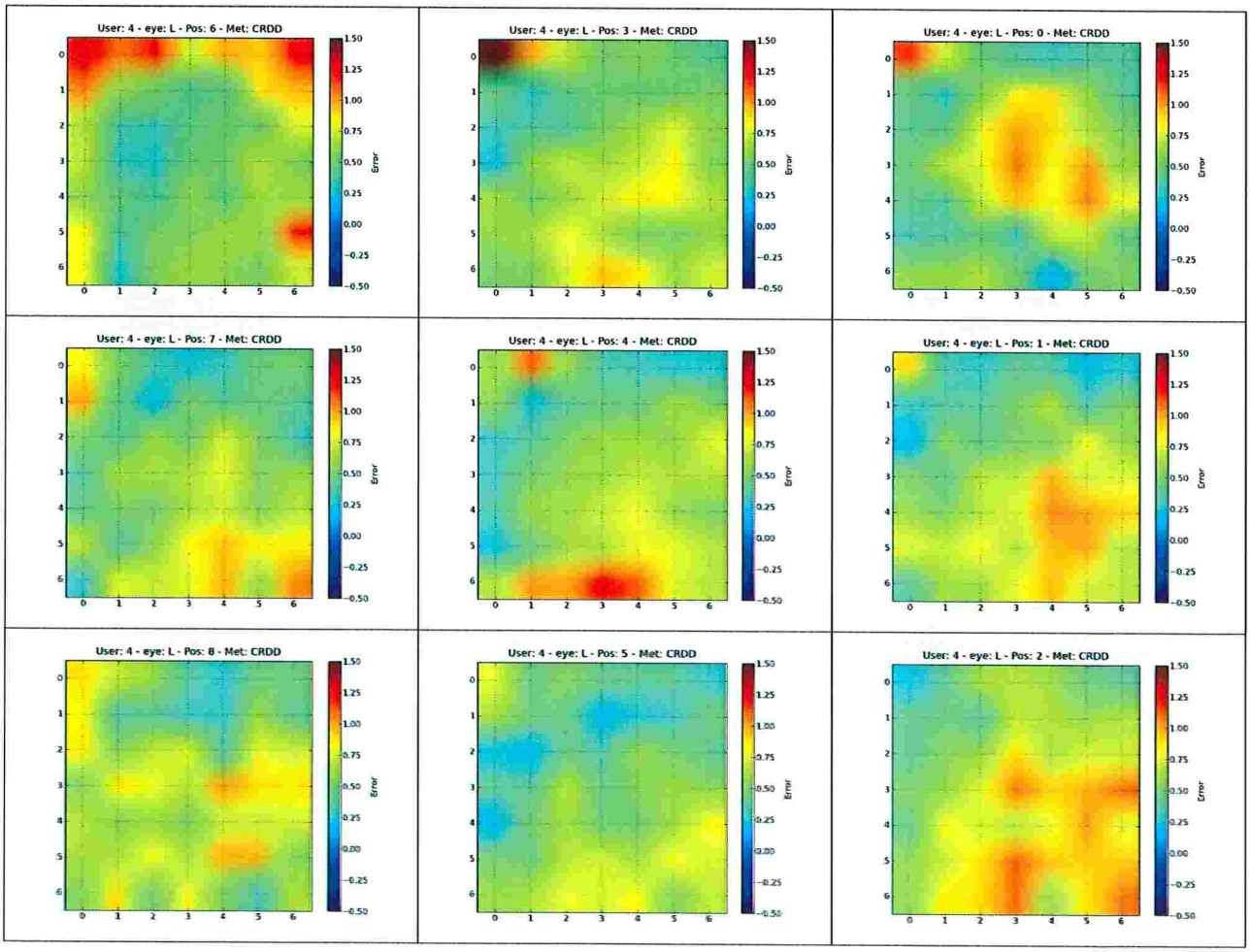

Figure B.66: *Distribution of the gaze error in all positions for the left eye of participant 4 using method* ***CR-DD****. Position 0 was used for calibration.*

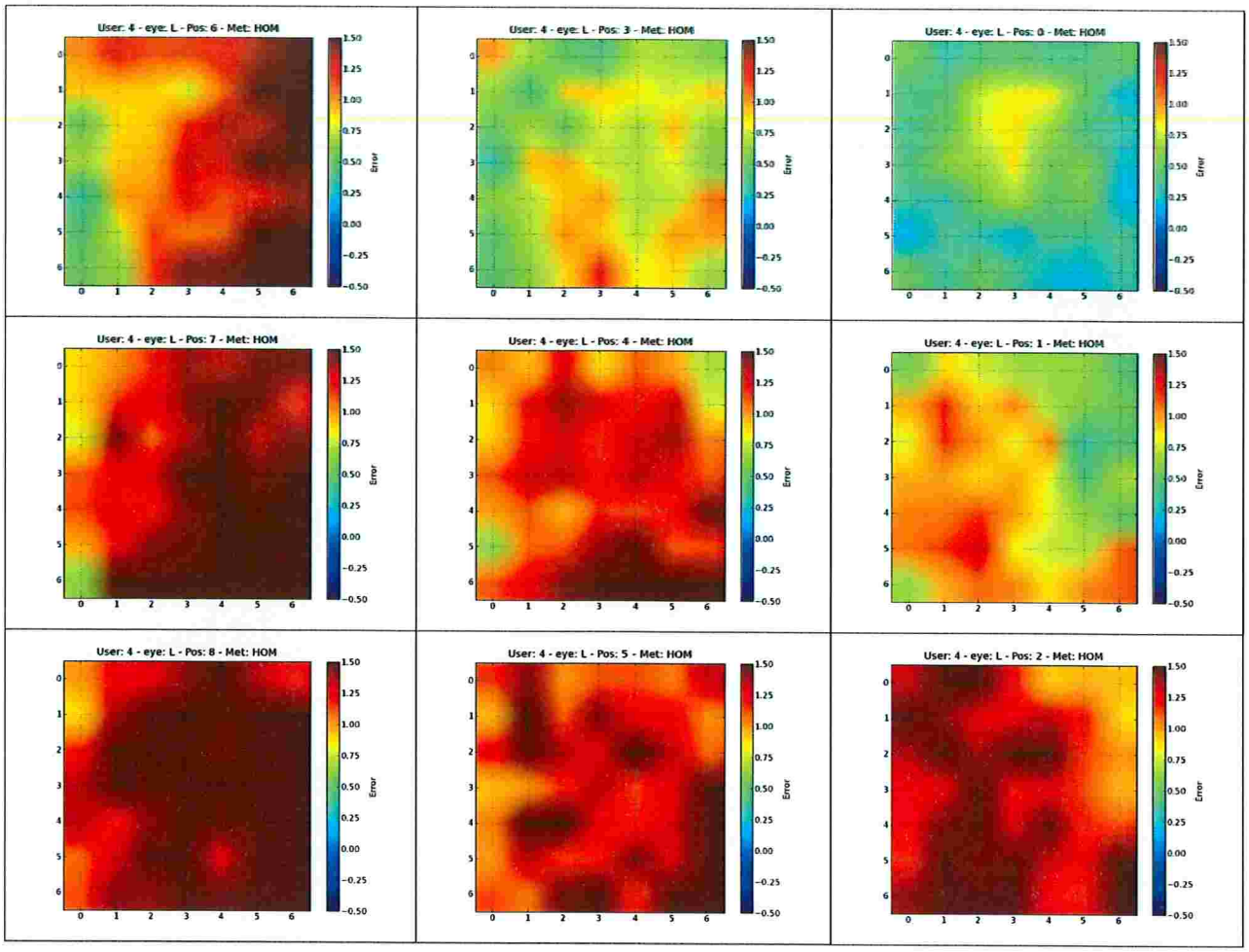

Figure B.67: *Distribution of the gaze error in all positions for the left eye of participant 4 using method* ***HOM****. Position 0 was used for calibration.*

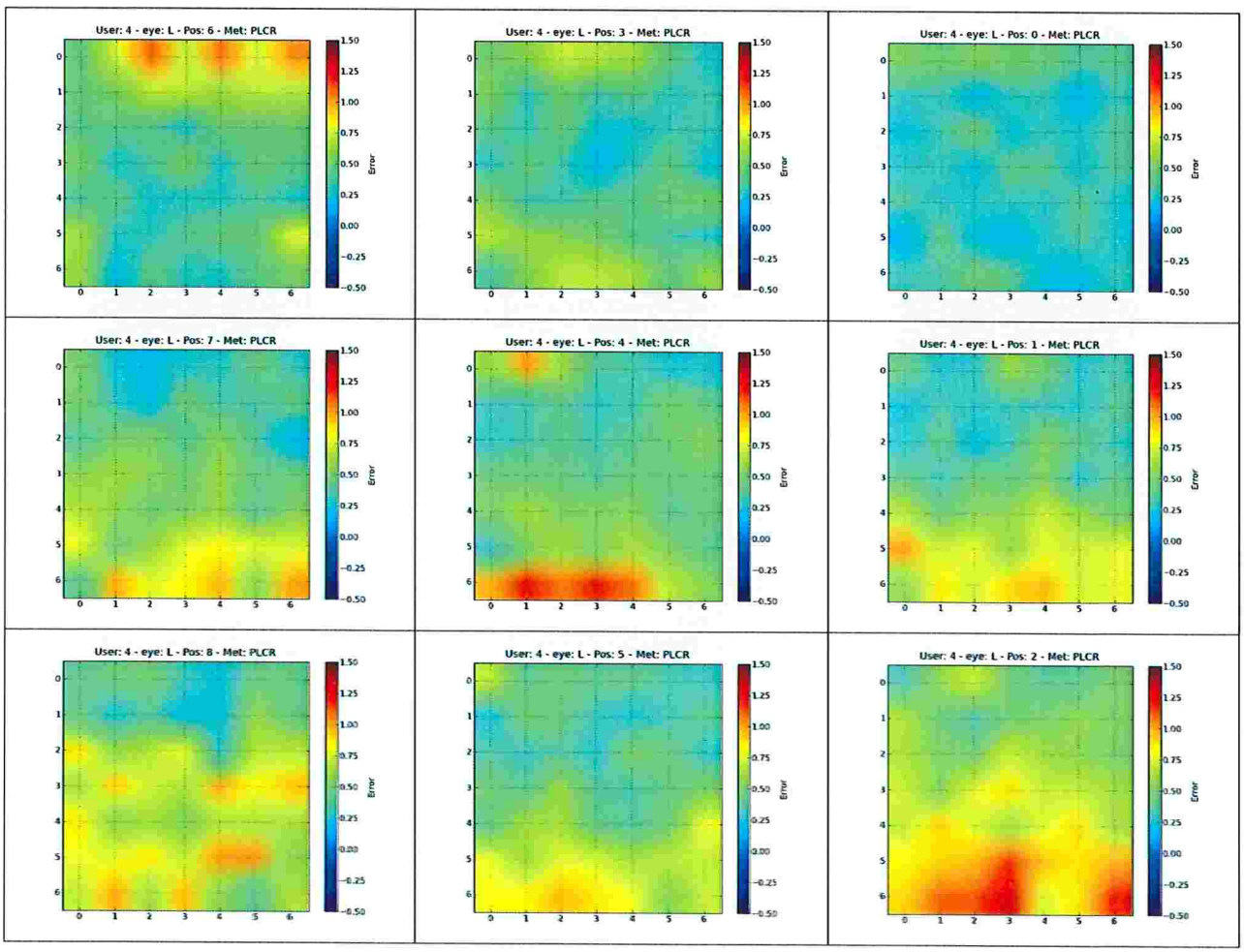

Figure B.68: *Distribution of the gaze error in all positions for the left eye of participant 4 using method* ***PL-CR****. Position 0 was used for calibration.*

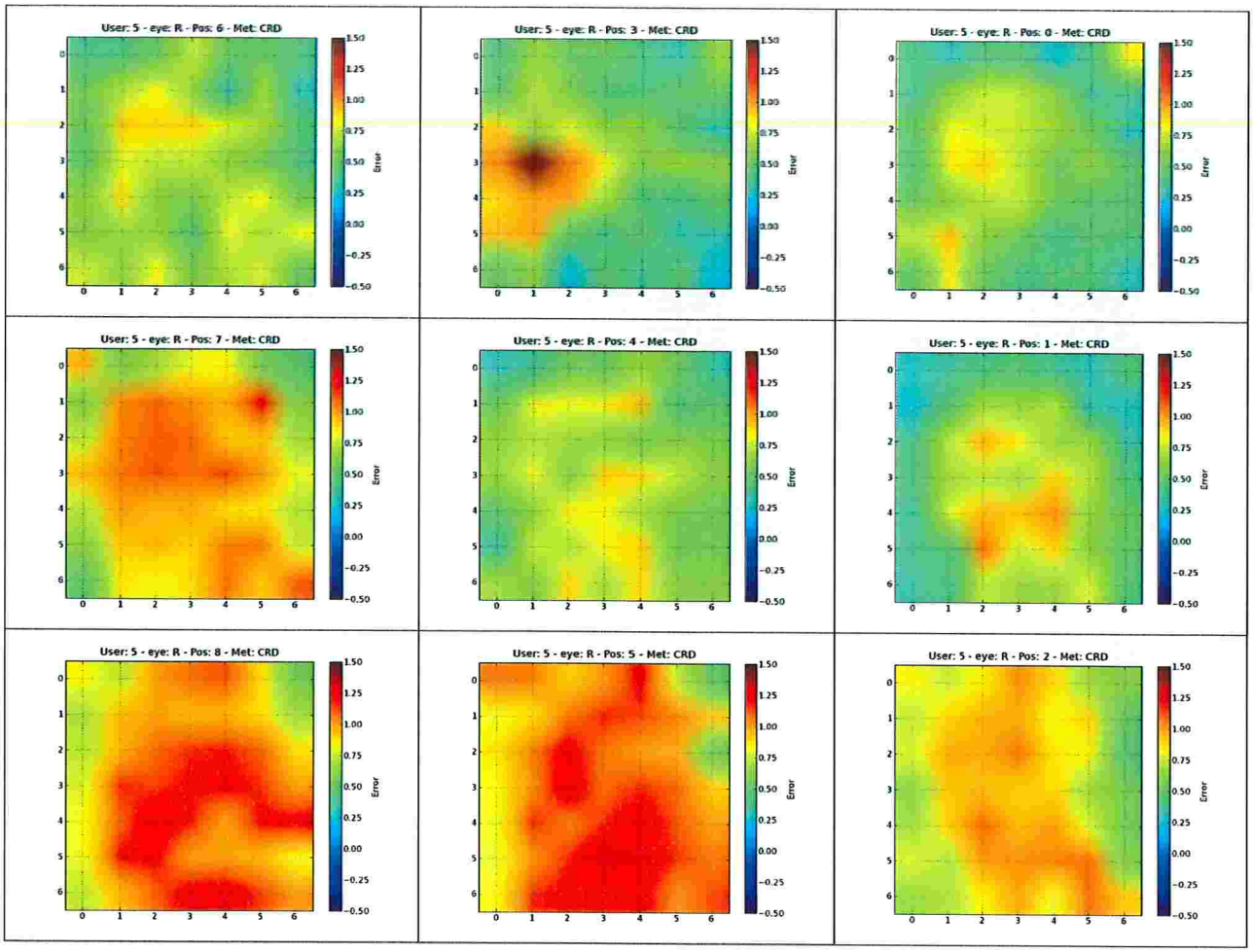

Figure B.69: *Distribution of the gaze error in all positions for the right eye of participant 5 using method* ***CR-D****. Position 0 was used for calibration.*

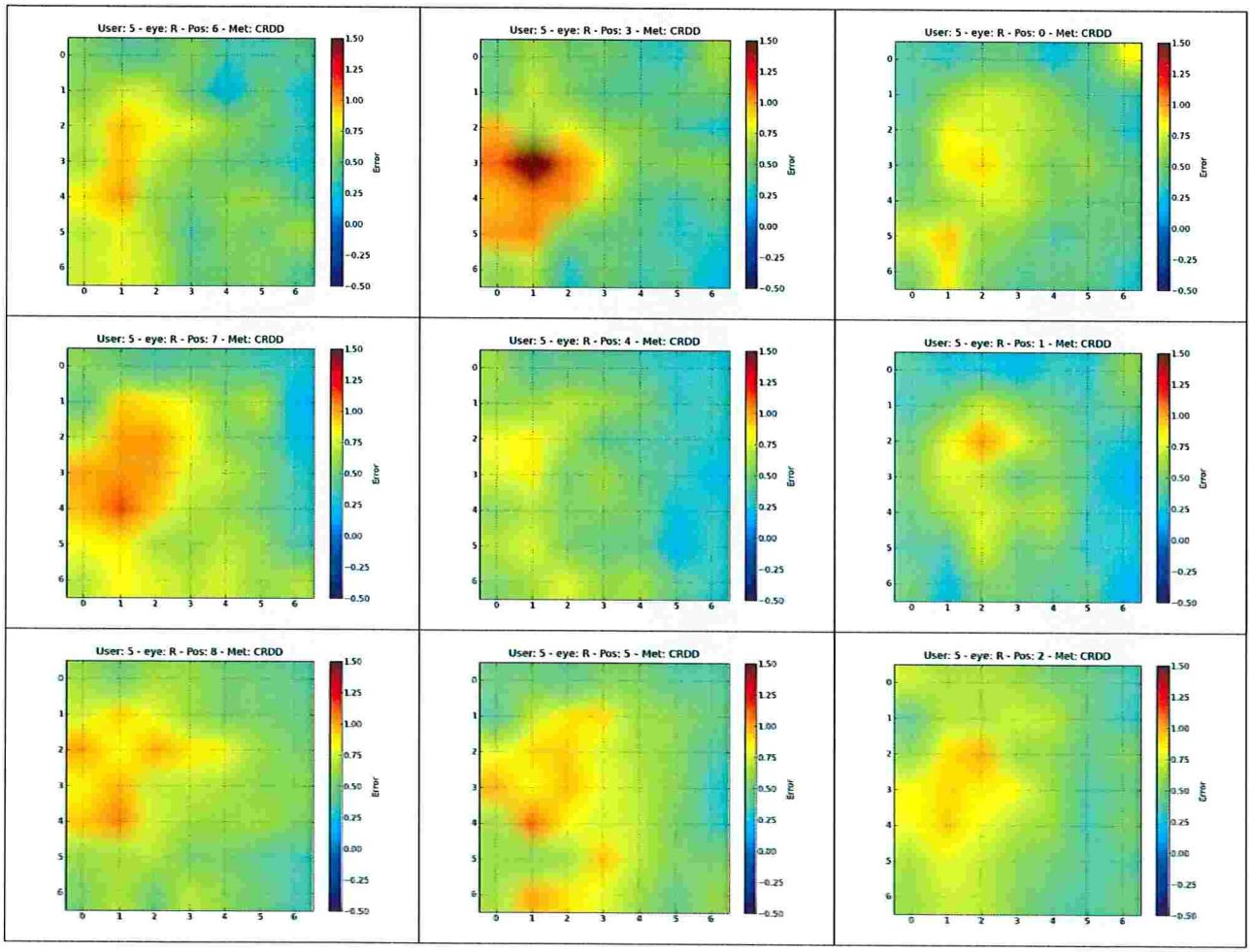

Figure B.70: *Distribution of the gaze error in all positions for the right eye of participant 5 using method* ***CR-DD****. Position 0 was used for calibration.*

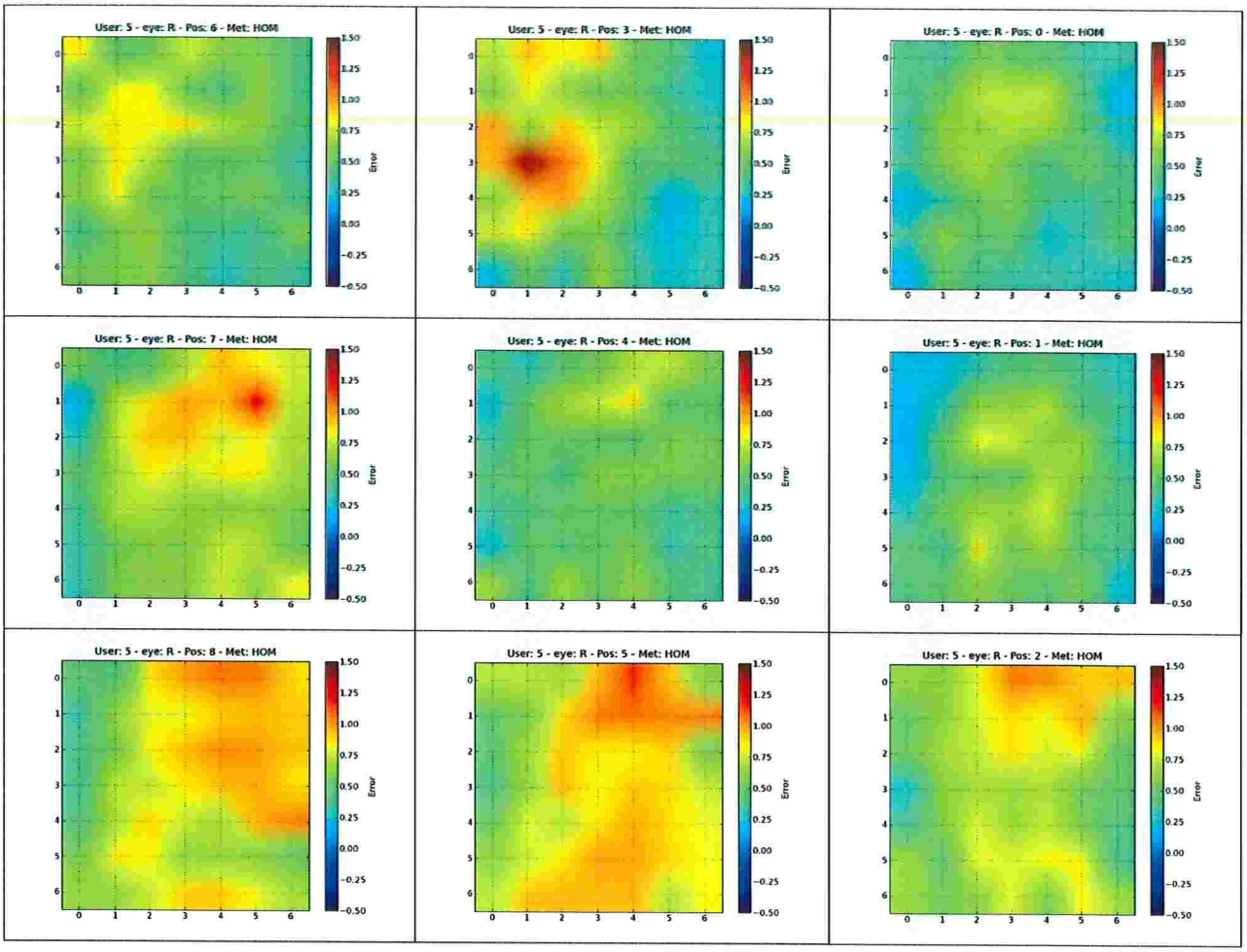

Figure B.71: *Distribution of the gaze error in all positions for the right eye of participant 5 using method* ***HOM****. Position 0 was used for calibration.*

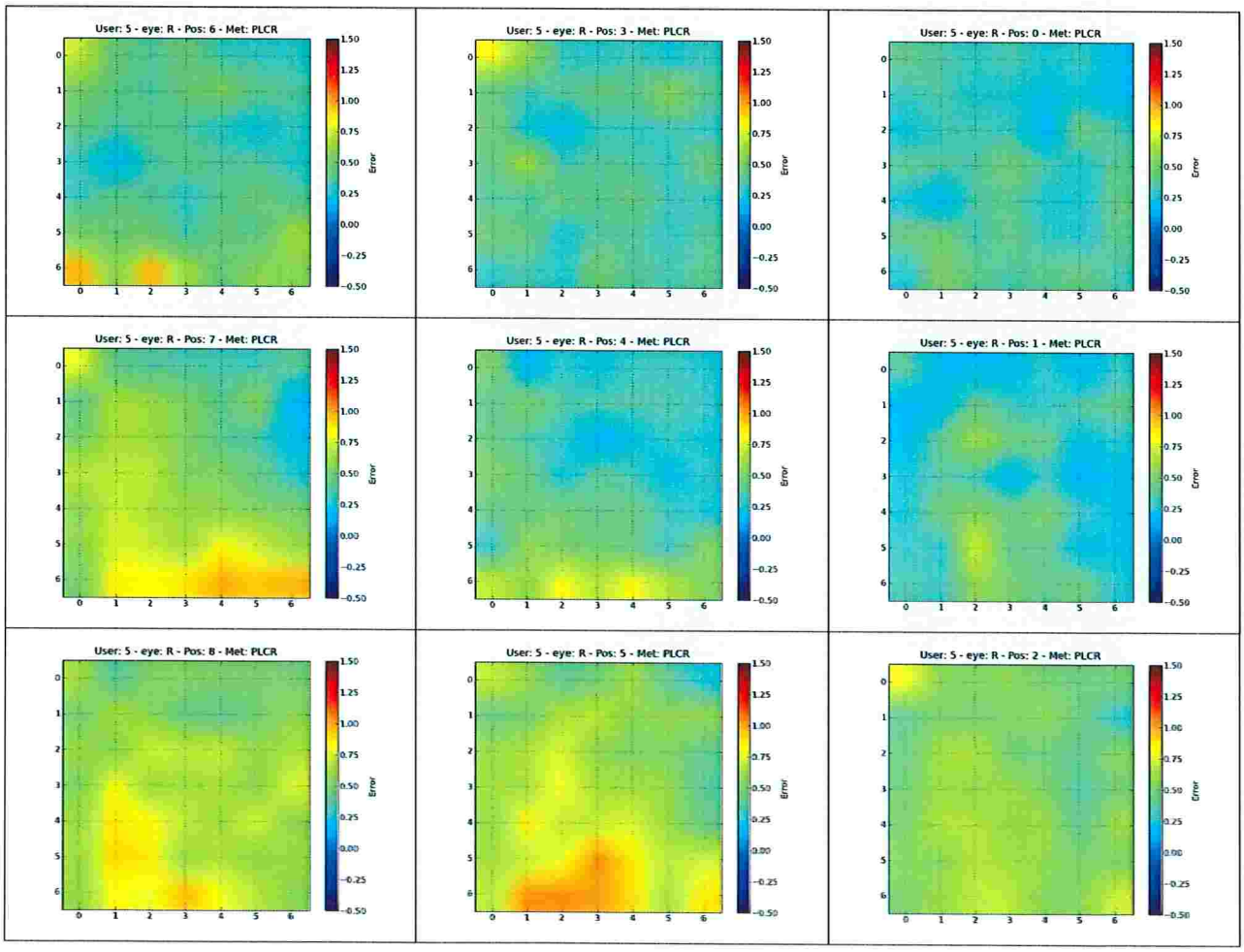

Figure B.72: *Distribution of the gaze error in all positions for the right eye of participant 5 using method* ***PL-CR****. Position 0 was used for calibration.*

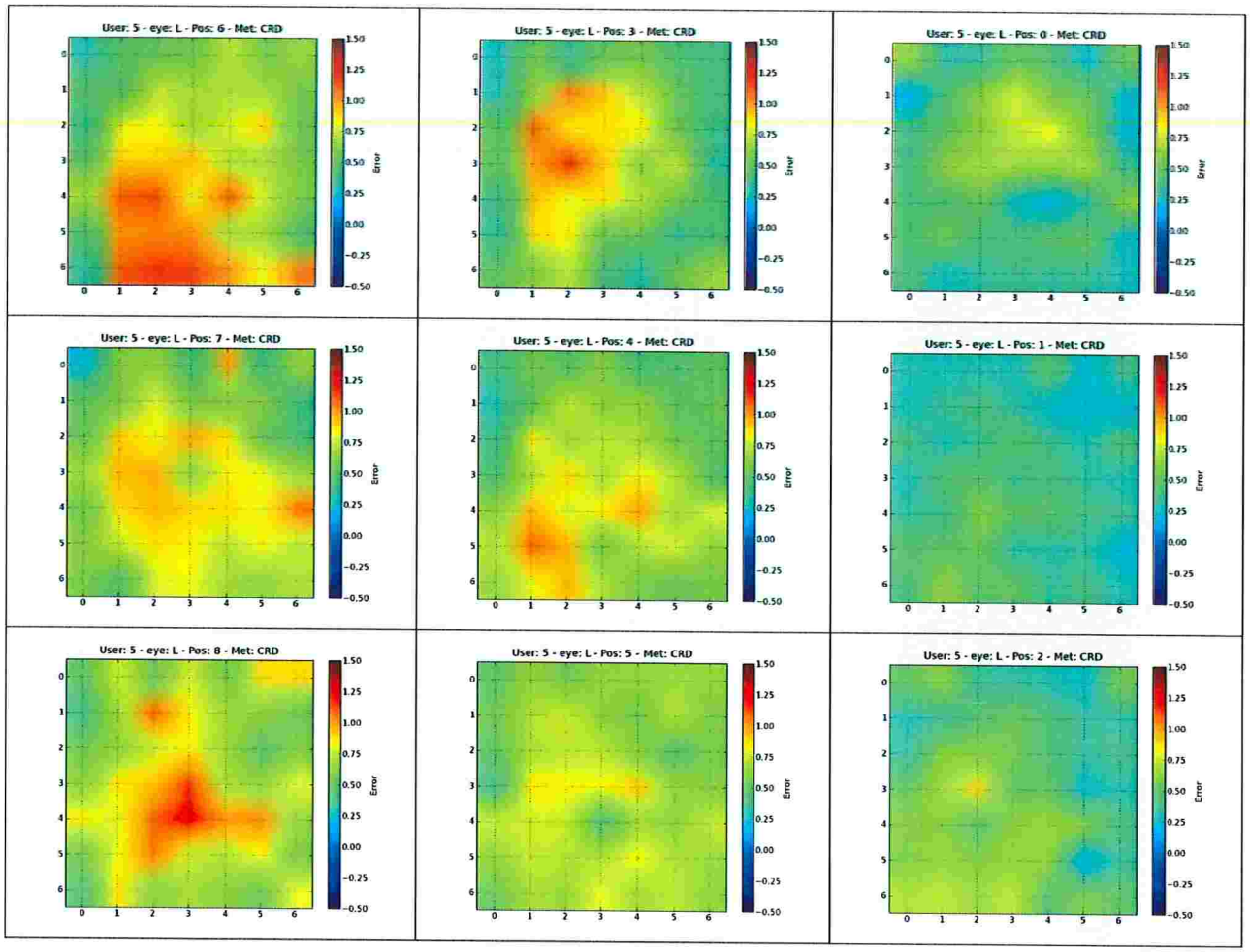

Figure B.73: *Distribution of the gaze error in all positions for the left eye of participant 5 using method* ***CR-D****. Position 0 was used for calibration.*

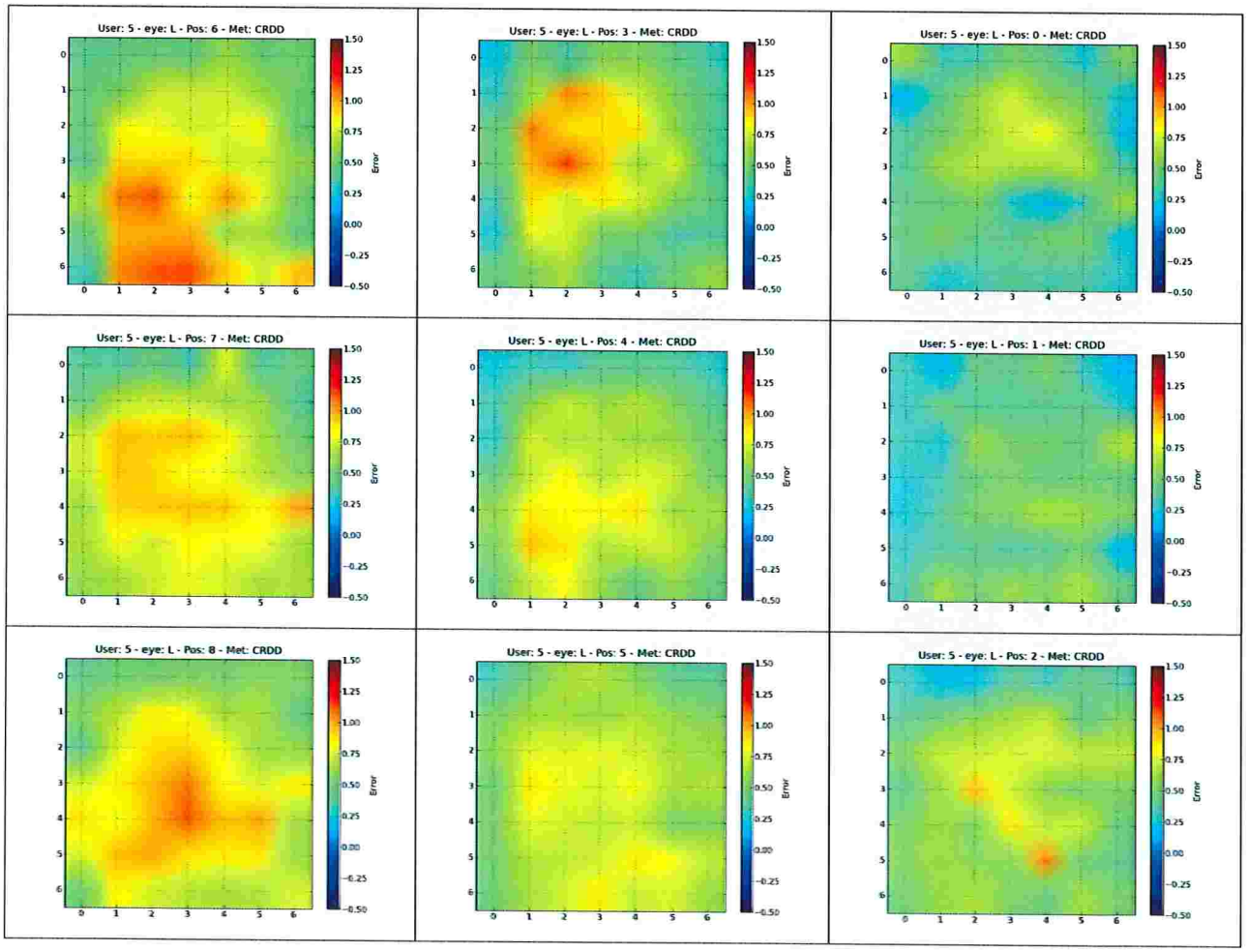

Figure B.74: *Distribution of the gaze error in all positions for the left eye of participant 5 using method* ***CR-DD****. Position 0 was used for calibration.*

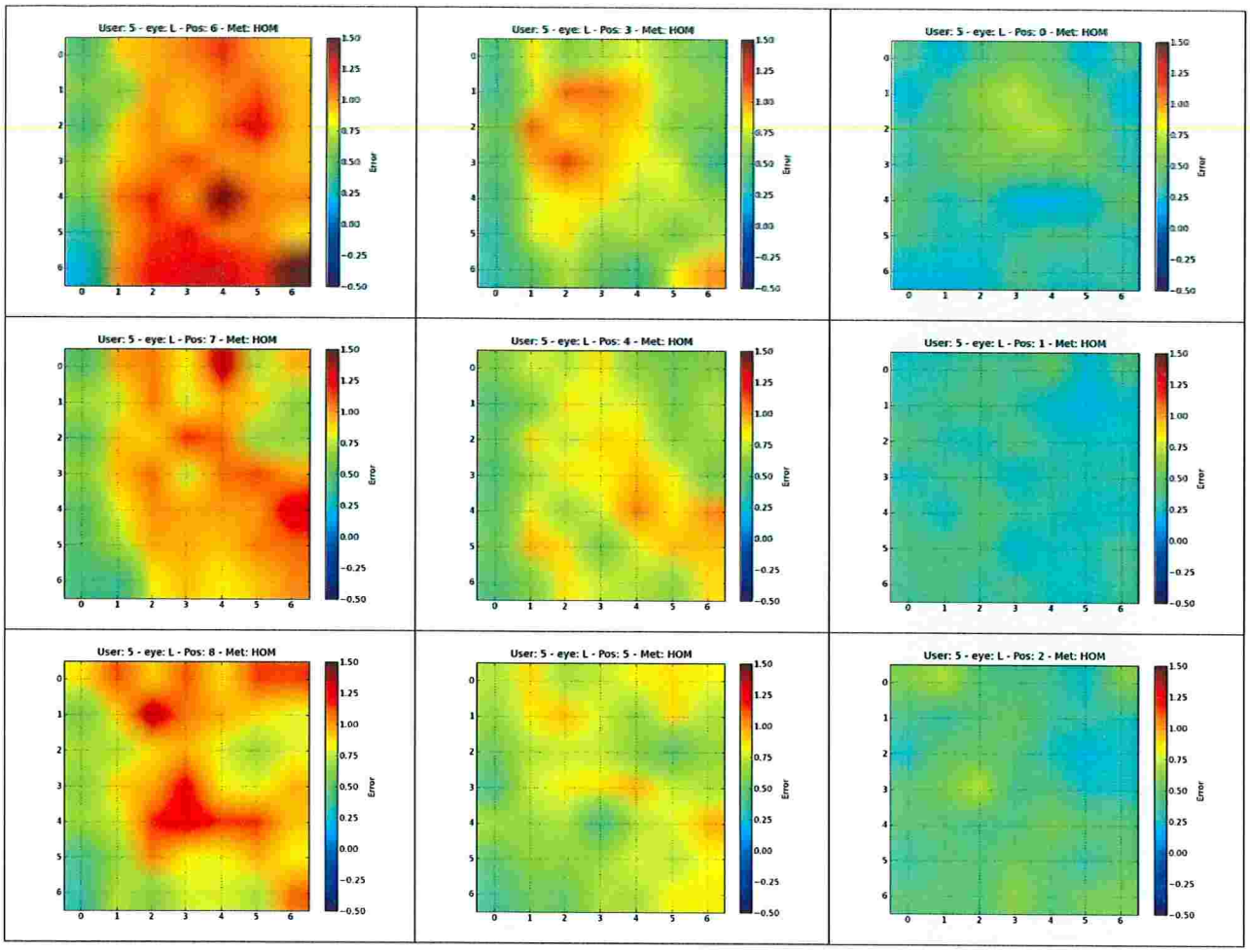

Figure B.75: *Distribution of the gaze error in all positions for the left eye of participant 5 using method **HOM**. Position 0 was used for calibration.*

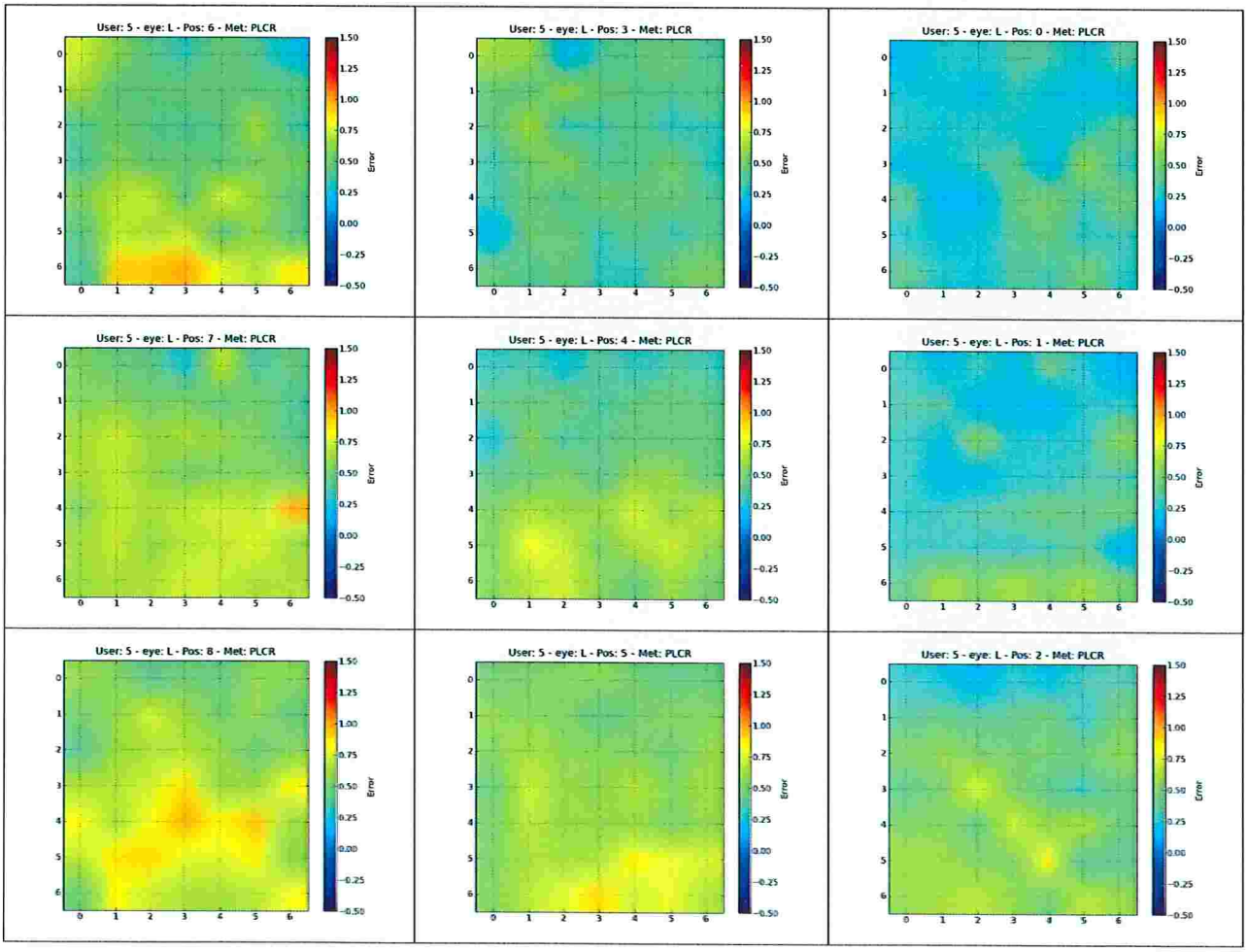

Figure B.76: *Distribution of the gaze error in all positions for the left eye of participant 5 using method **PL-CR**. Position 0 was used for calibration.*

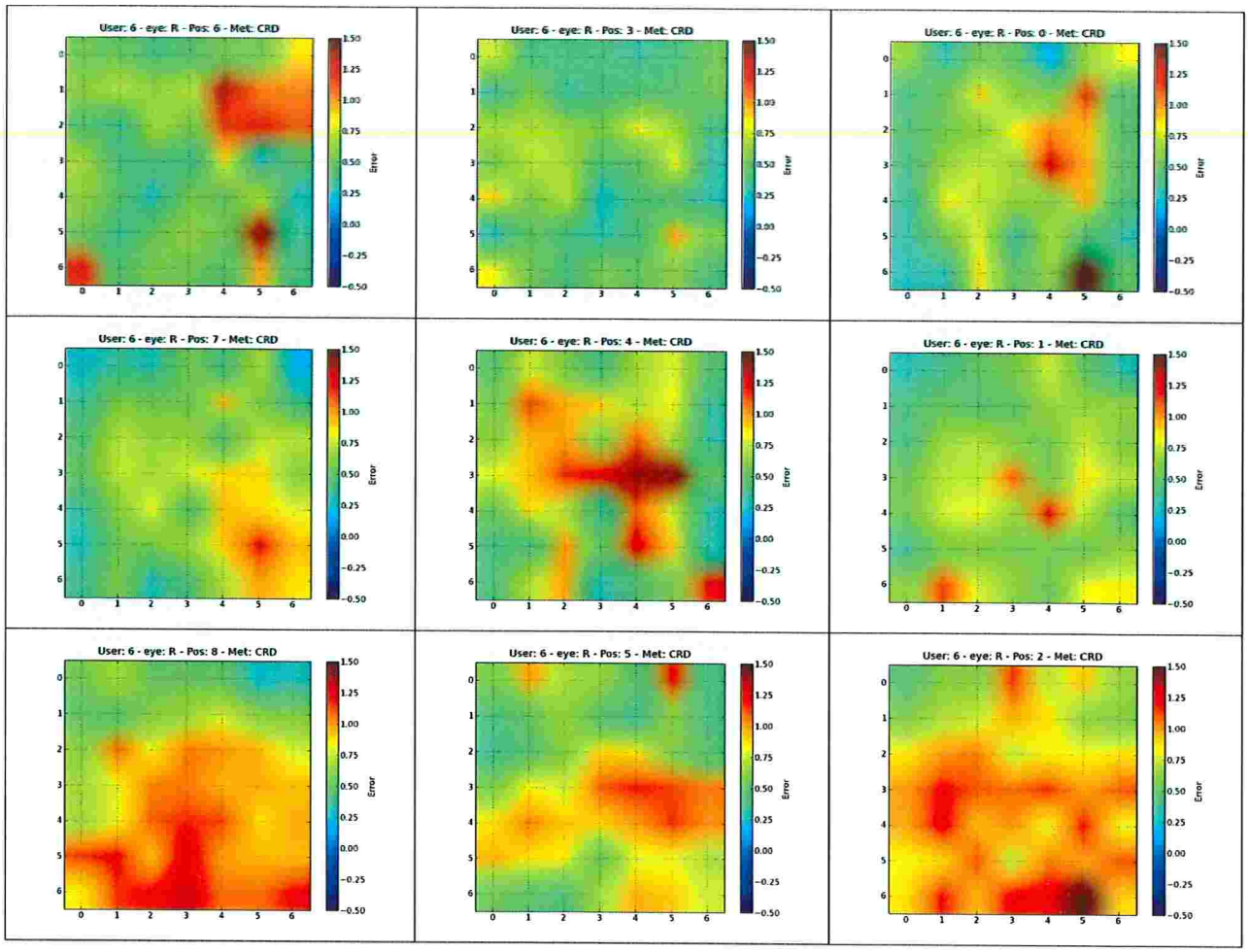

Figure B.77: *Distribution of the gaze error in all positions for the right eye of participant 6 using method* ***CR-D****. Position 0 was used for calibration.*

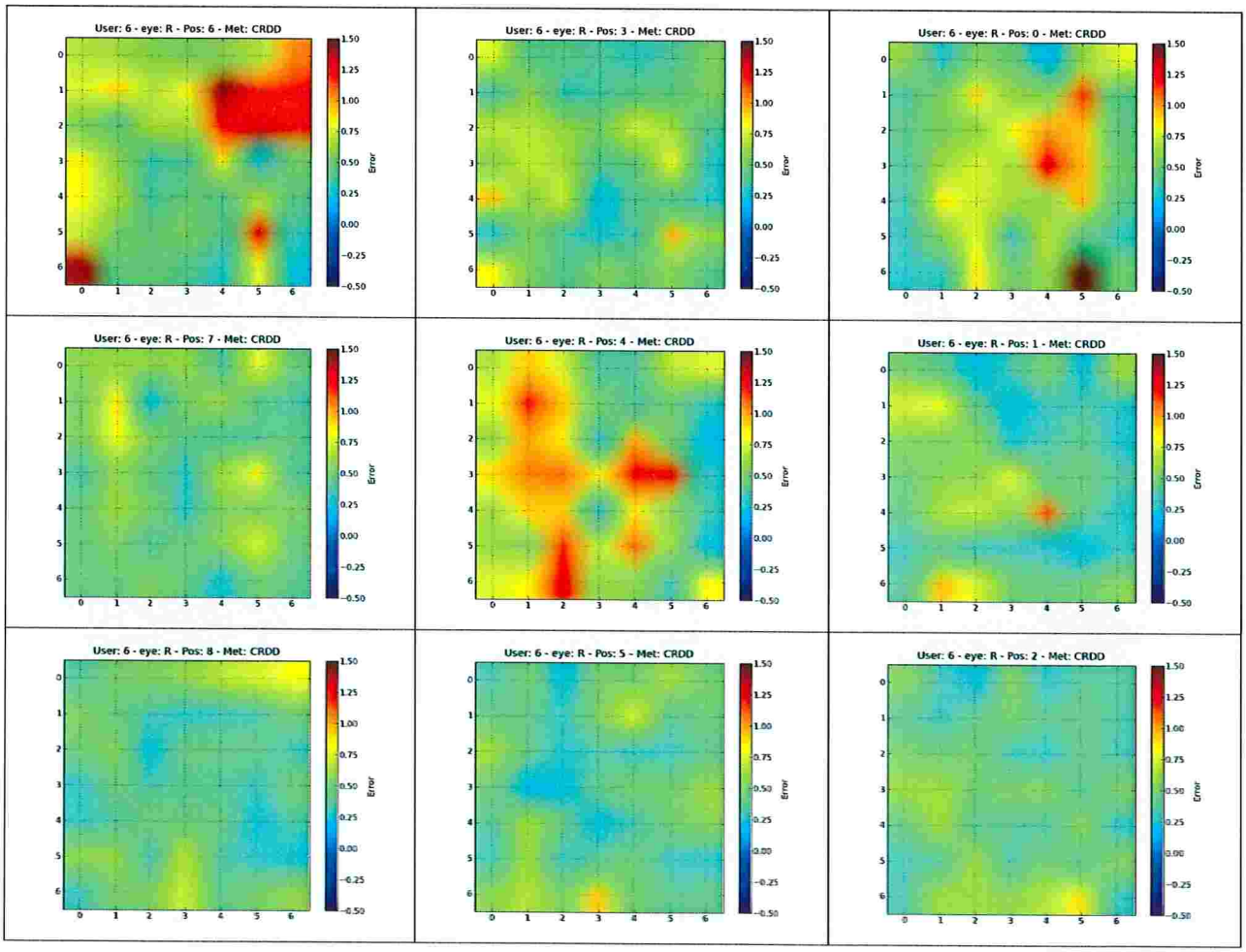

Figure B.78: *Distribution of the gaze error in all positions for the right eye of participant 6 using method* ***CR-DD****. Position 0 was used for calibration.*

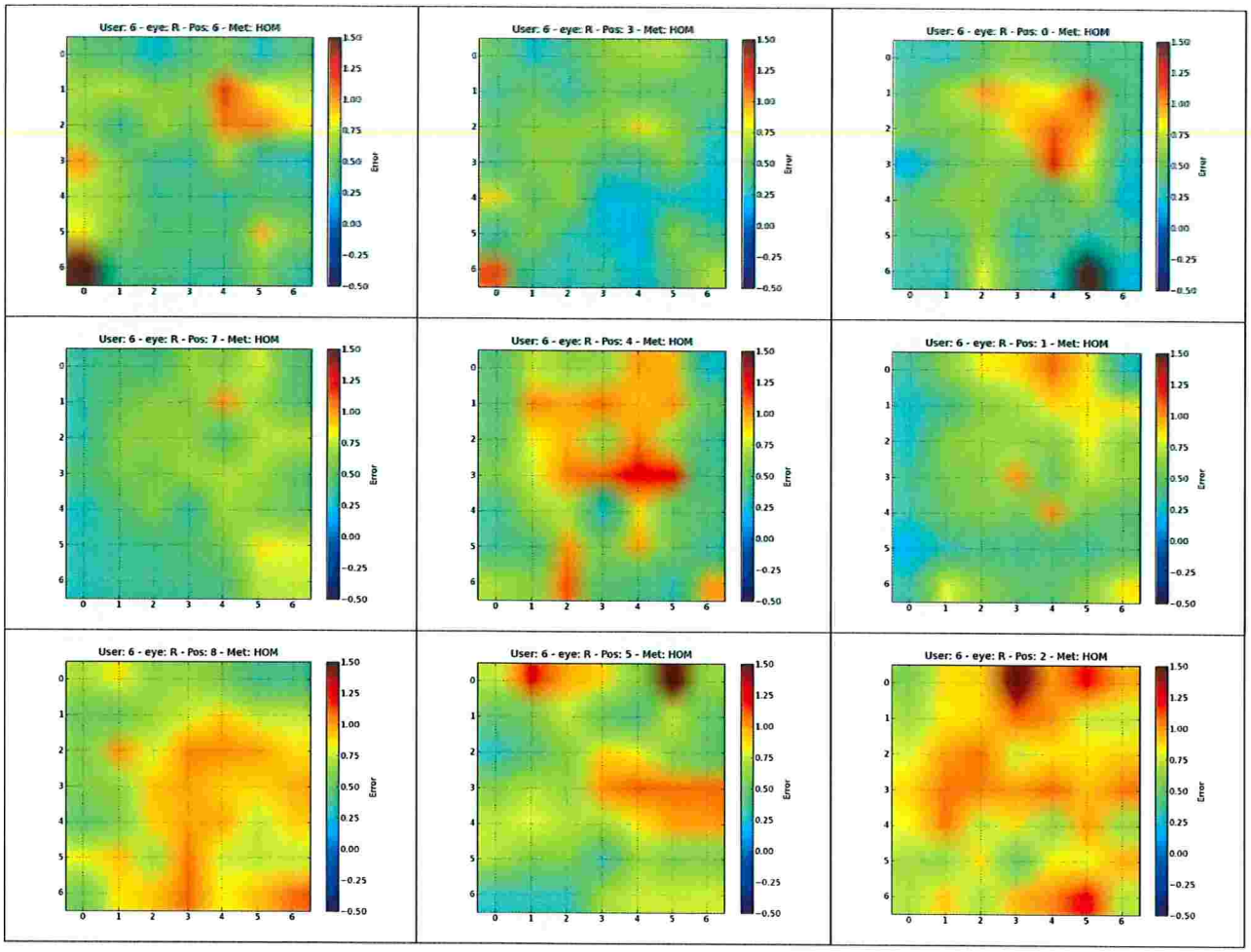

Figure B.79: *Distribution of the gaze error in all positions for the right eye of participant 6 using method* ***HOM****. Position 0 was used for calibration.*

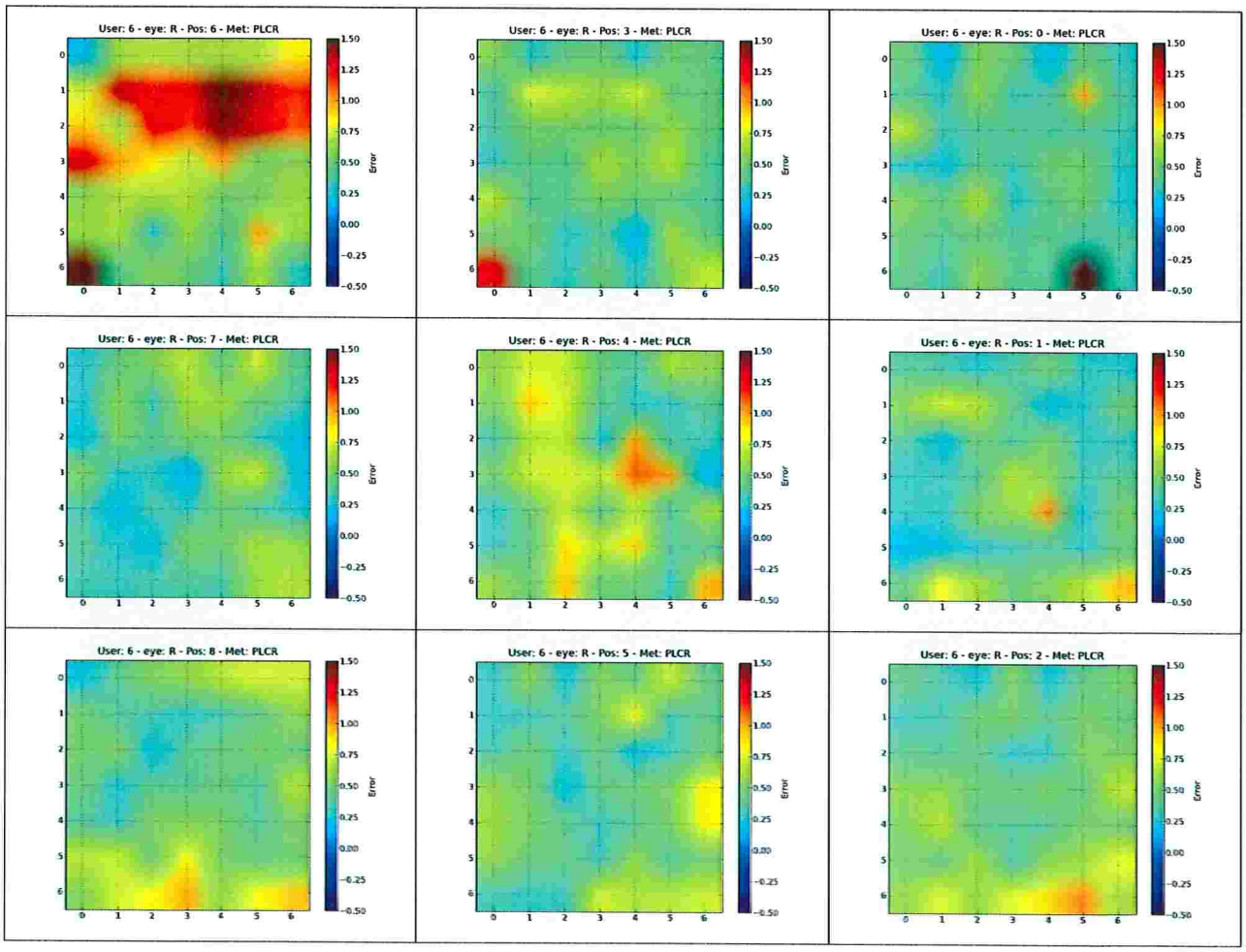

Figure B.80: *Distribution of the gaze error in all positions for the right eye of participant 6 using method* ***PL-CR****. Position 0 was used for calibration.*

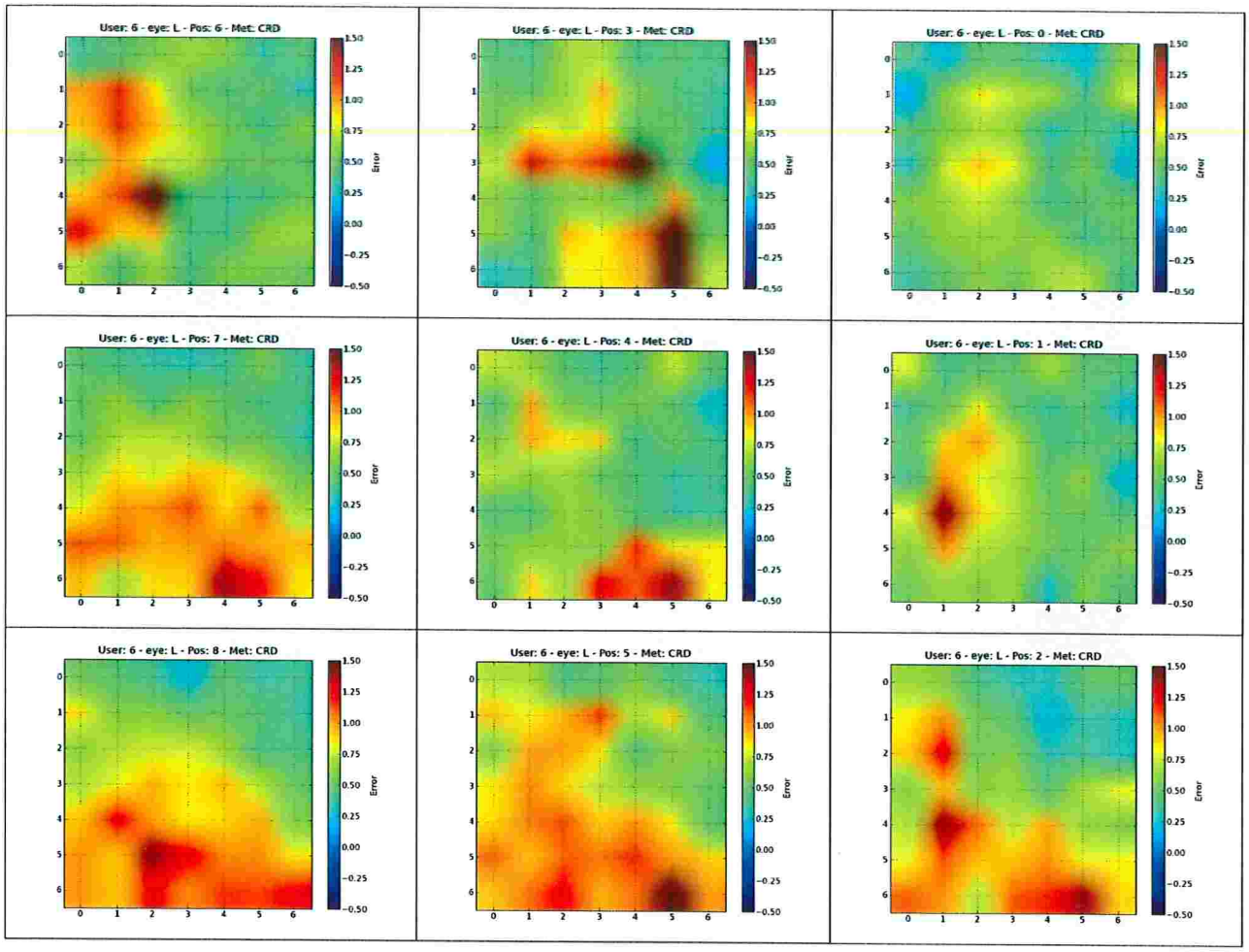

Figure B.81: *Distribution of the gaze error in all positions for the left eye of participant 6 using method* ***CR-D****. Position 0 was used for calibration.*

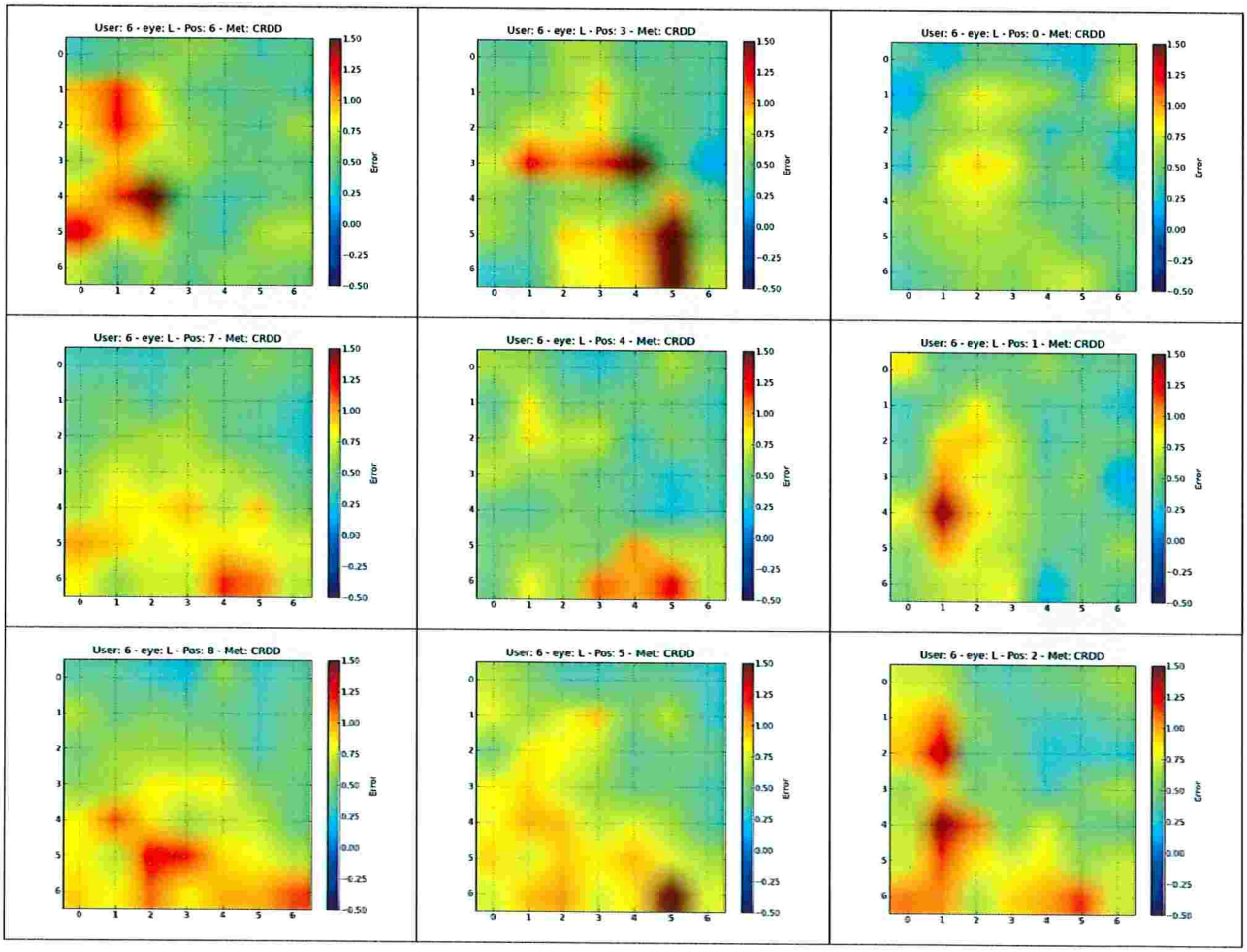

Figure B.82: *Distribution of the gaze error in all positions for the left eye of participant 6 using method* ***CR-DD****. Position 0 was used for calibration.*

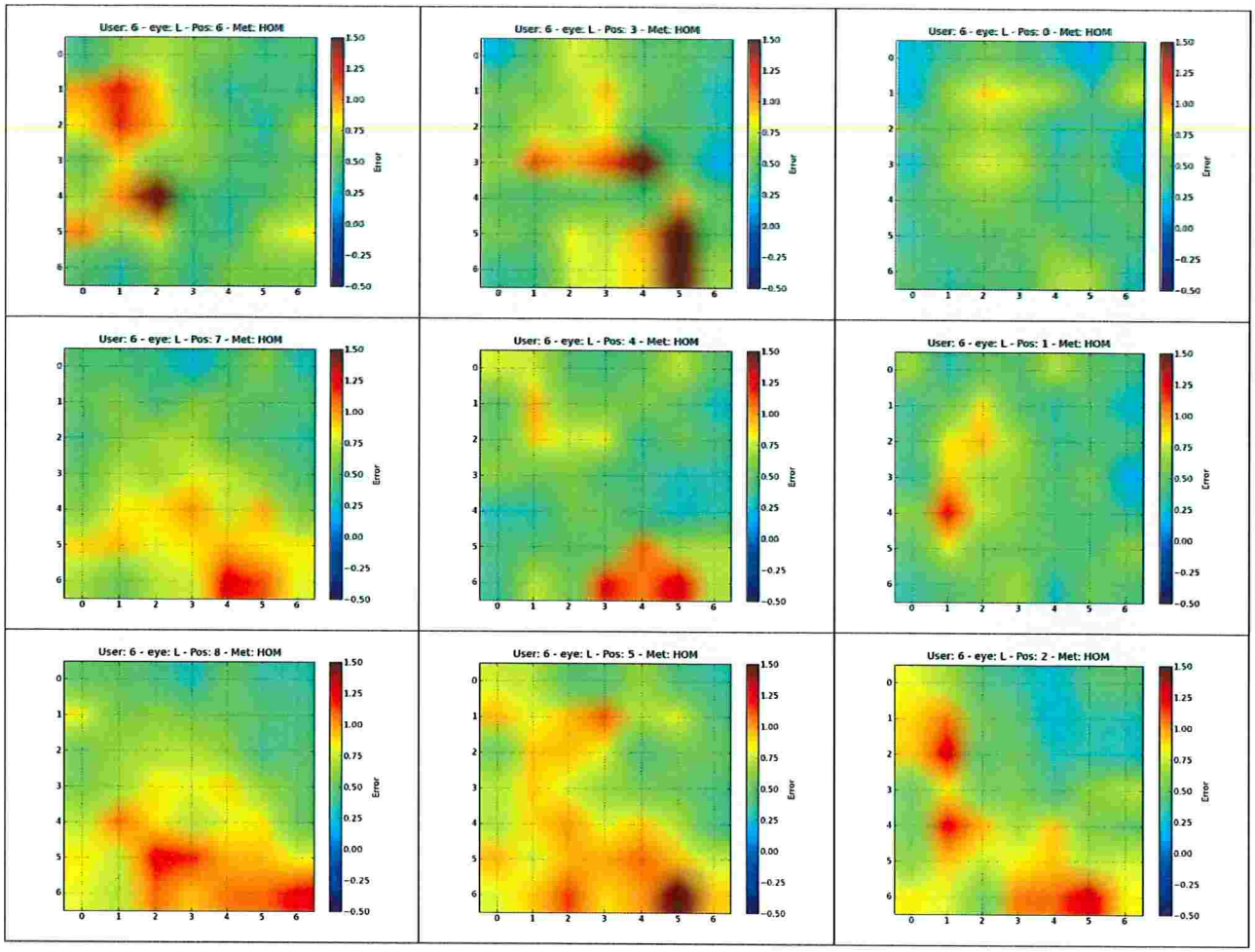

Figure B.83: *Distribution of the gaze error in all positions for the left eye of participant 6 using method* ***HOM****. Position 0 was used for calibration.*

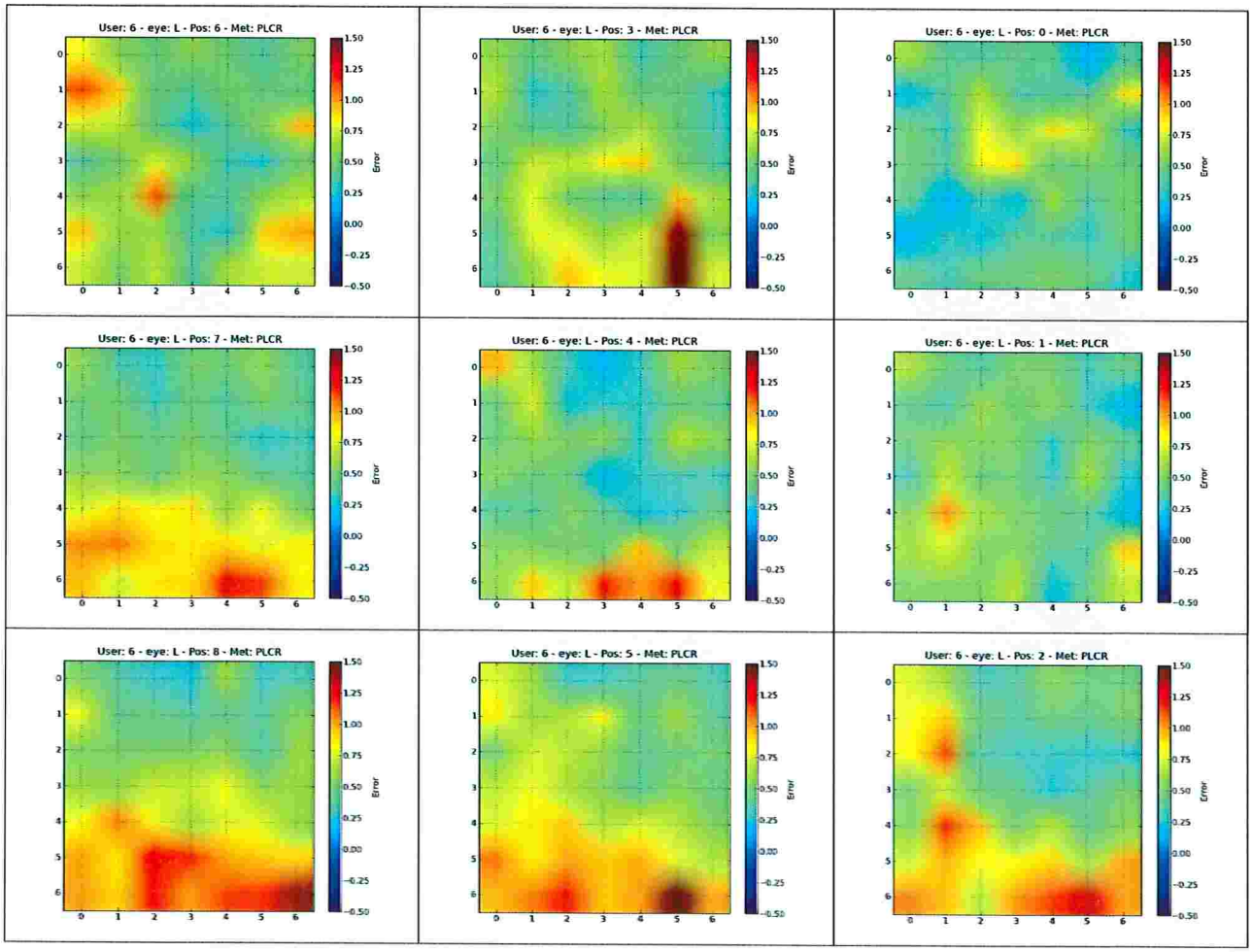

Figure B.84: *Distribution of the gaze error in all positions for the left eye of participant 6 using method* ***PL-CR****. Position 0 was used for calibration.*

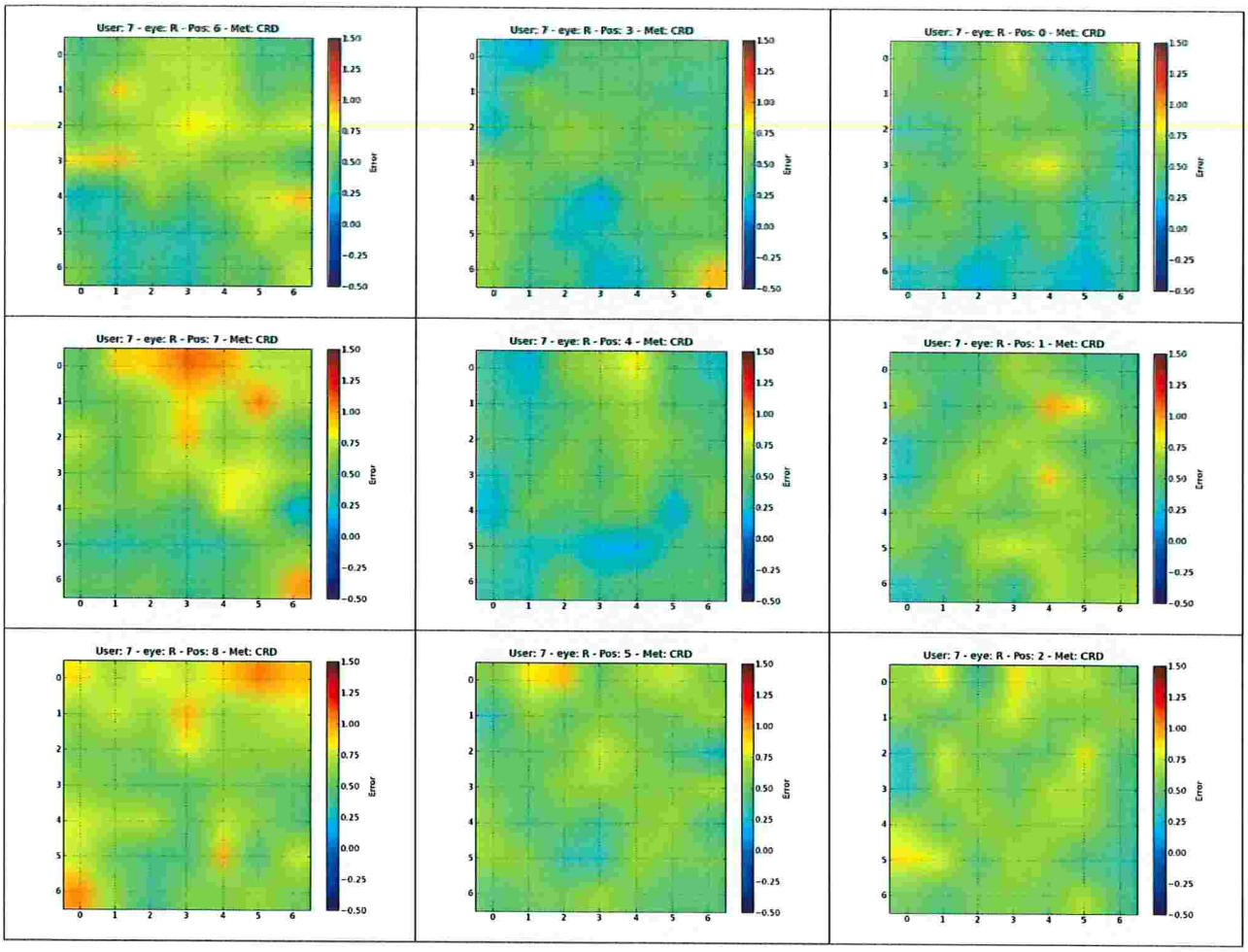

Figure B.85: *Distribution of the gaze error in all positions for the right eye of participant 7 using method* ***CR-D****. Position 0 was used for calibration.*

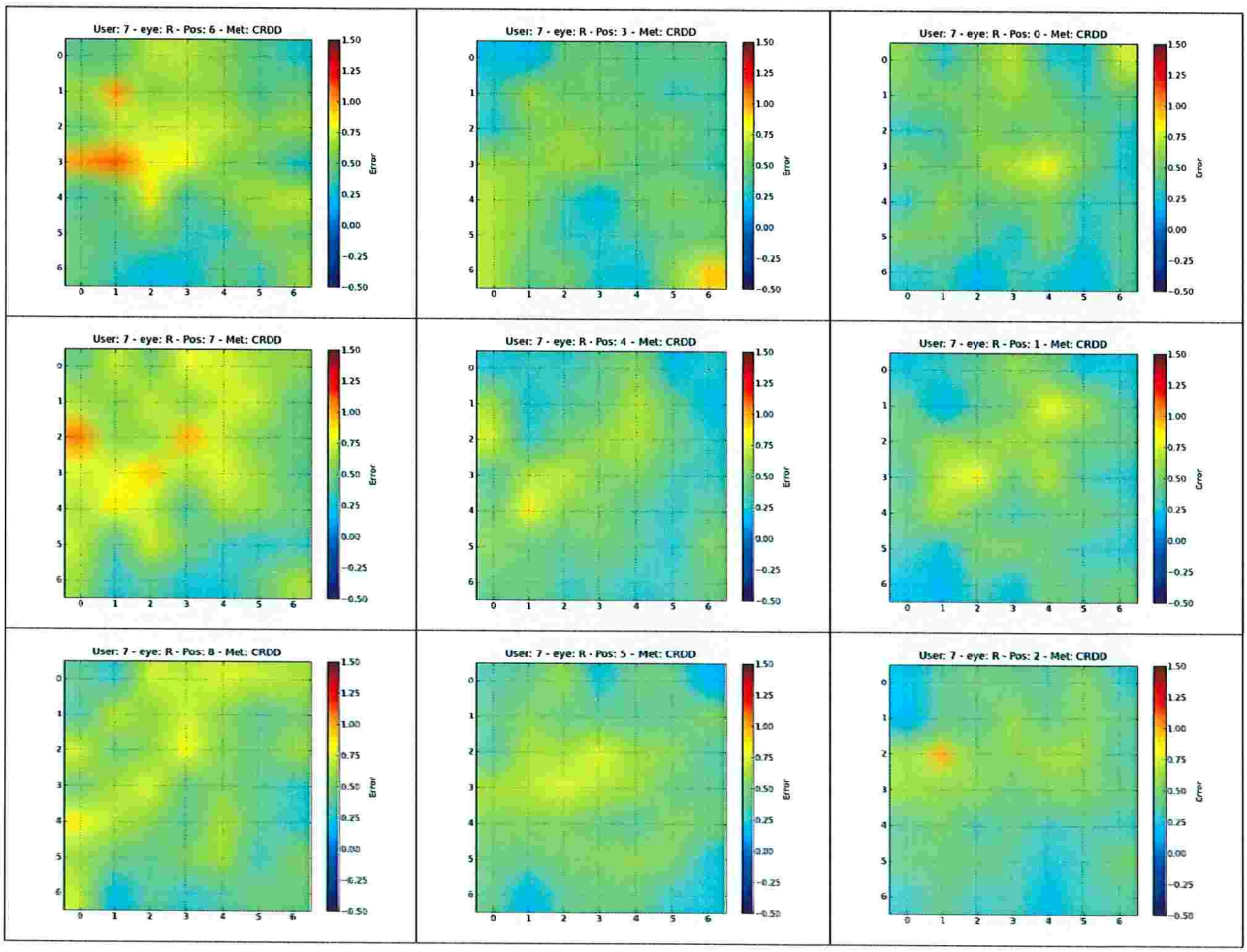

Figure B.86: *Distribution of the gaze error in all positions for the right eye of participant 7 using method* ***CR-DD****. Position 0 was used for calibration.*

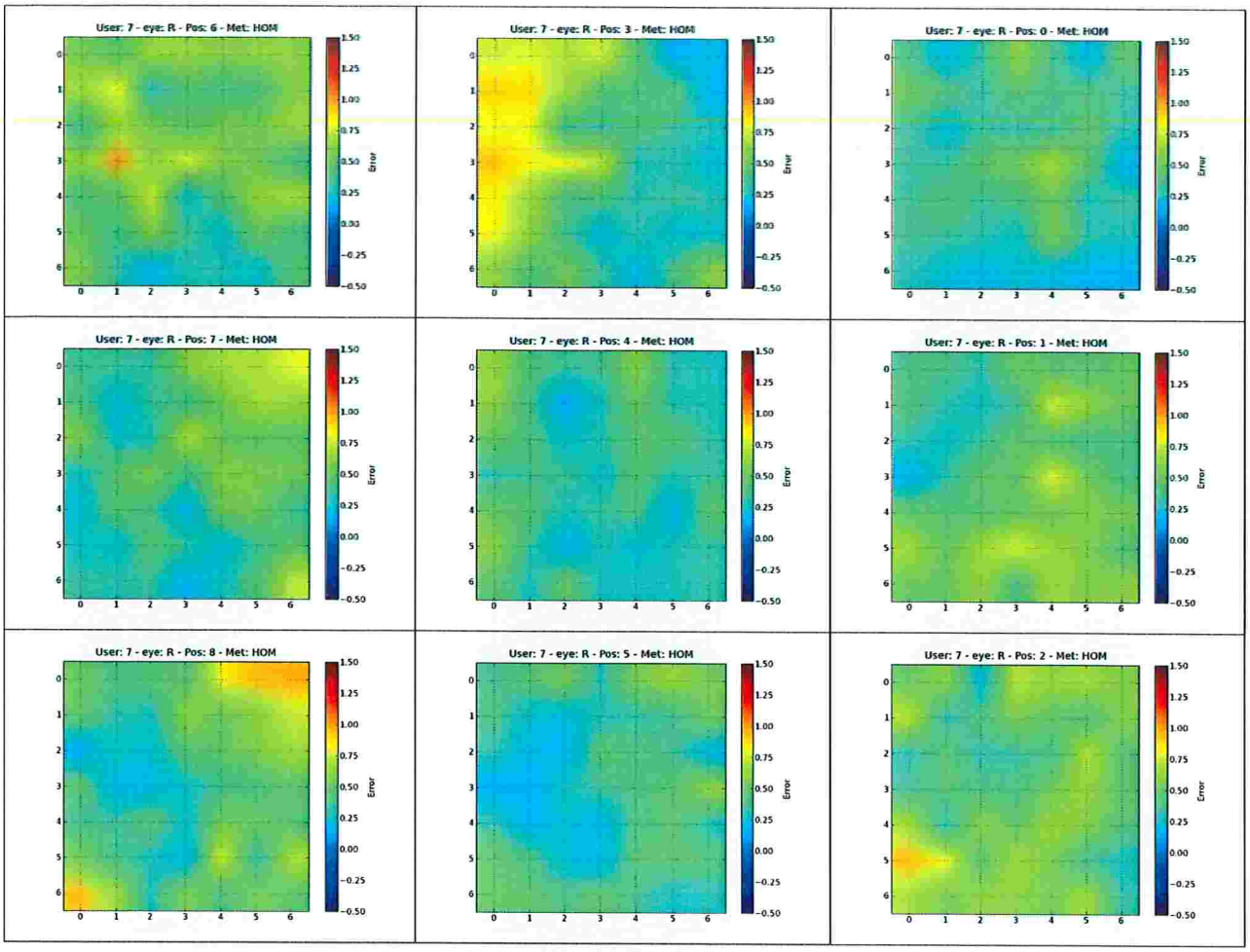

Figure B.87: *Distribution of the gaze error in all positions for the right eye of participant 7 using method* ***HOM****. Position 0 was used for calibration.*

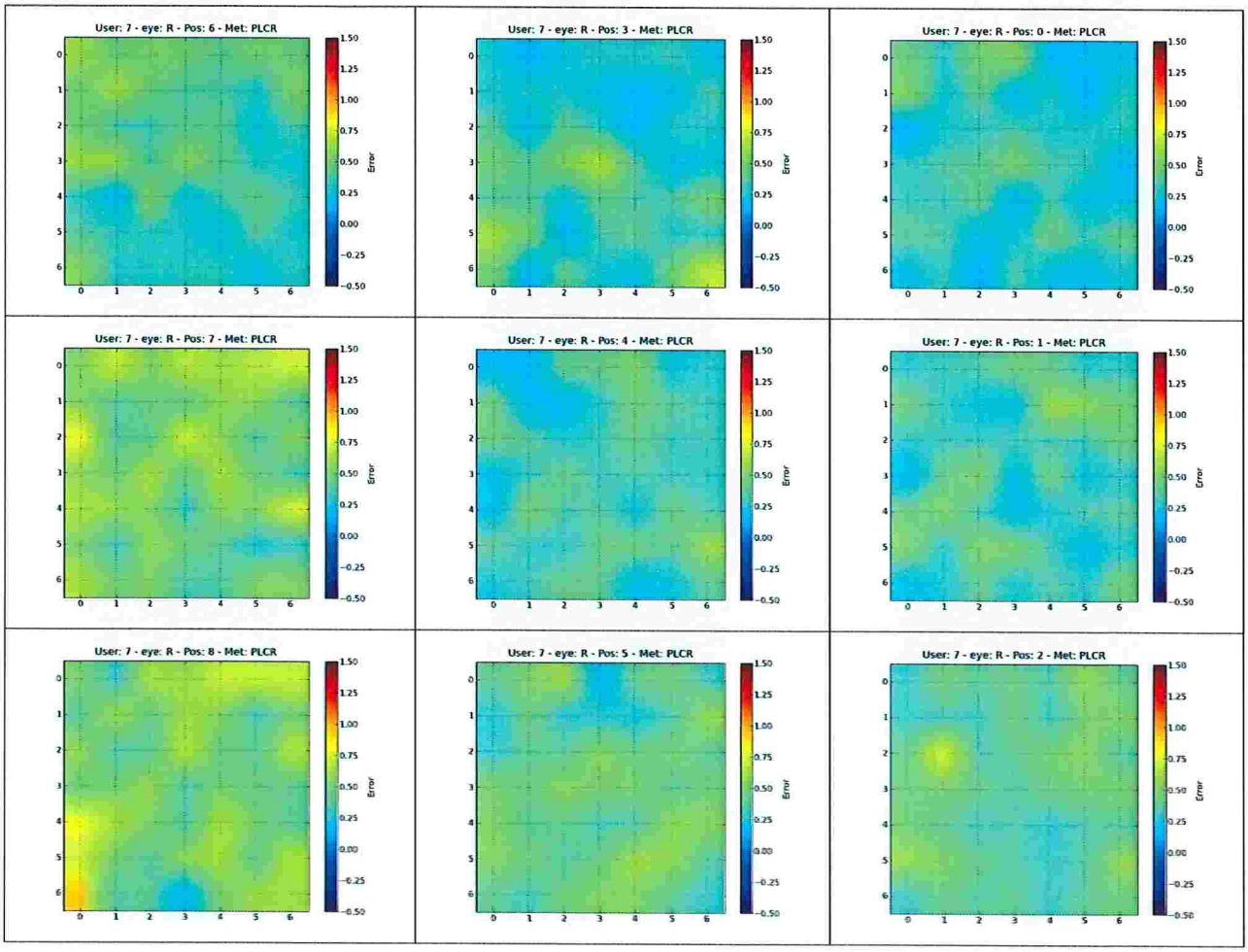

Figure B.88: *Distribution of the gaze error in all positions for the right eye of participant 7 using method* ***PL-CR****. Position 0 was used for calibration.*

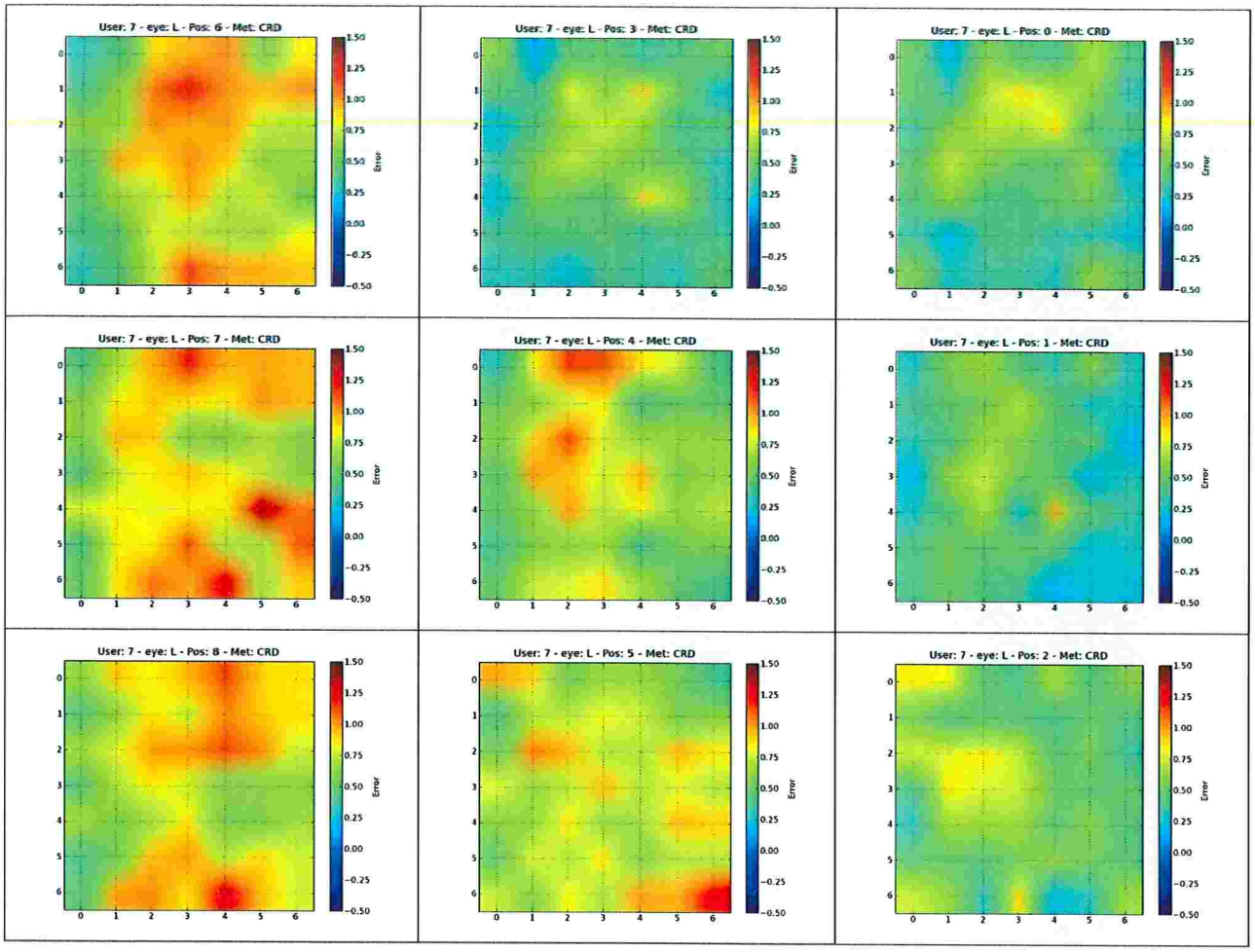

Figure B.89: *Distribution of the gaze error in all positions for the left eye of participant 7 using method* ***CR-D****. Position 0 was used for calibration.*

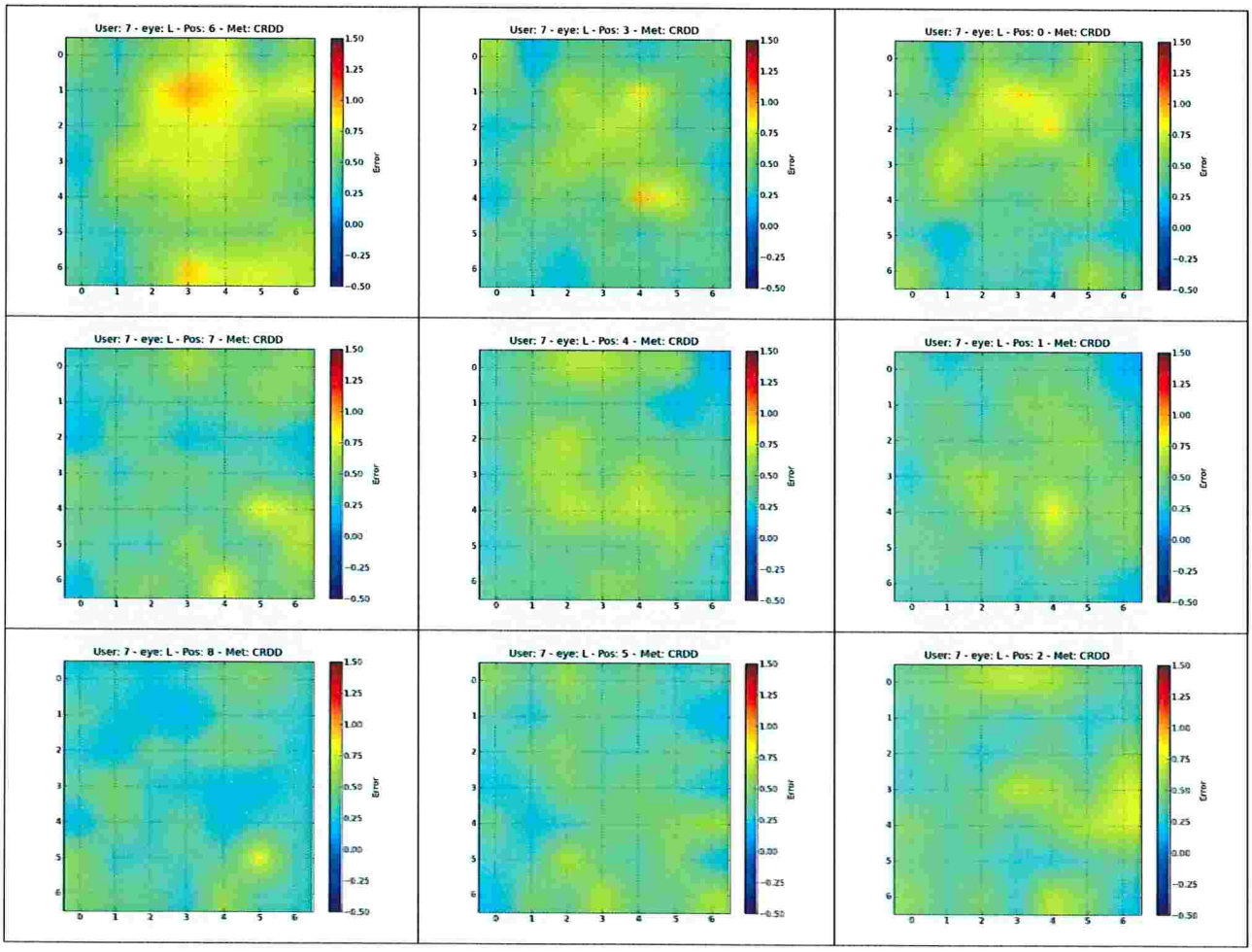

Figure B.90: *Distribution of the gaze error in all positions for the left eye of participant 7 using method* ***CR-DD****. Position 0 was used for calibration.*

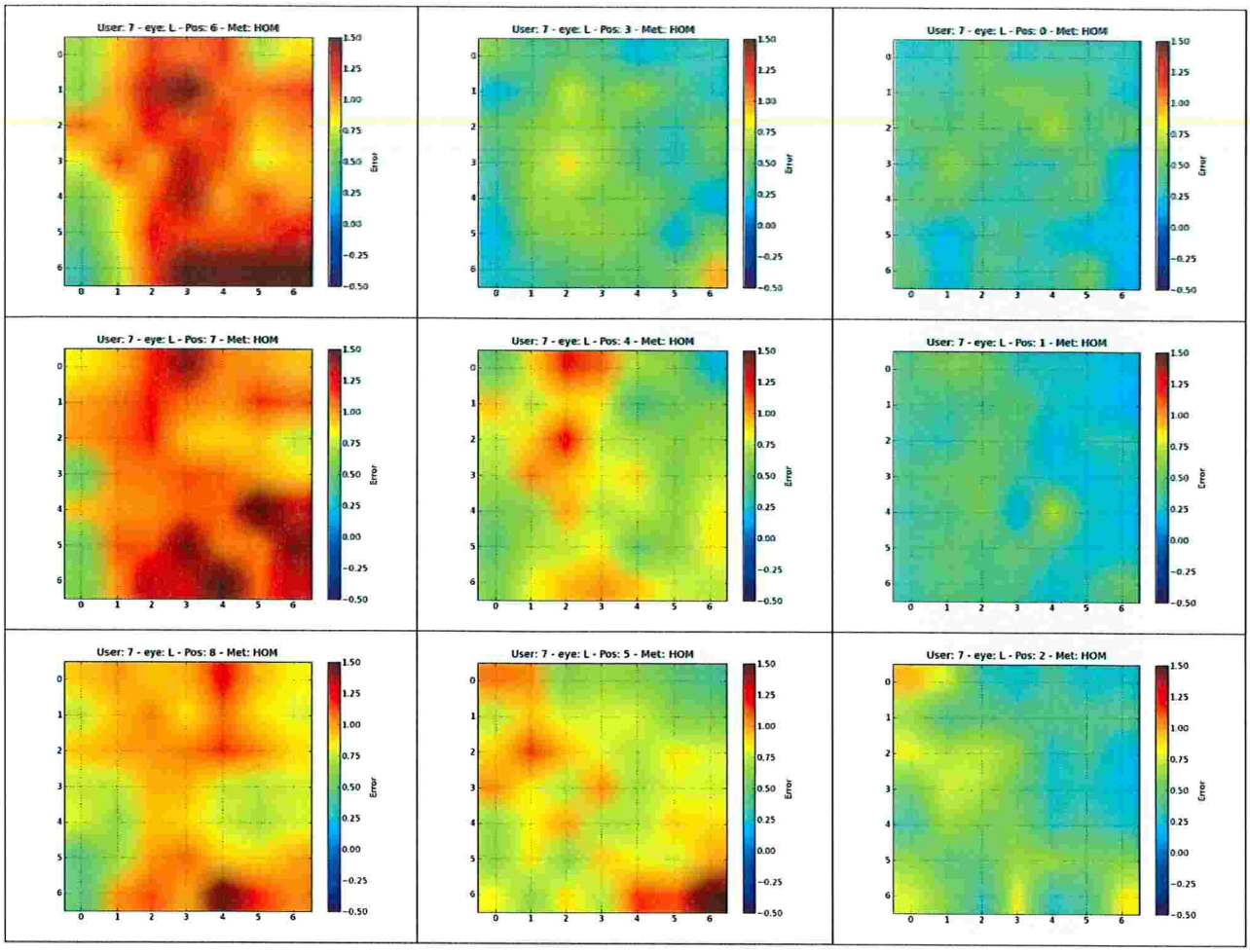

Figure B.91: *Distribution of the gaze error in all positions for the left eye of participant 7 using method* ***HOM****. Position 0 was used for calibration.*

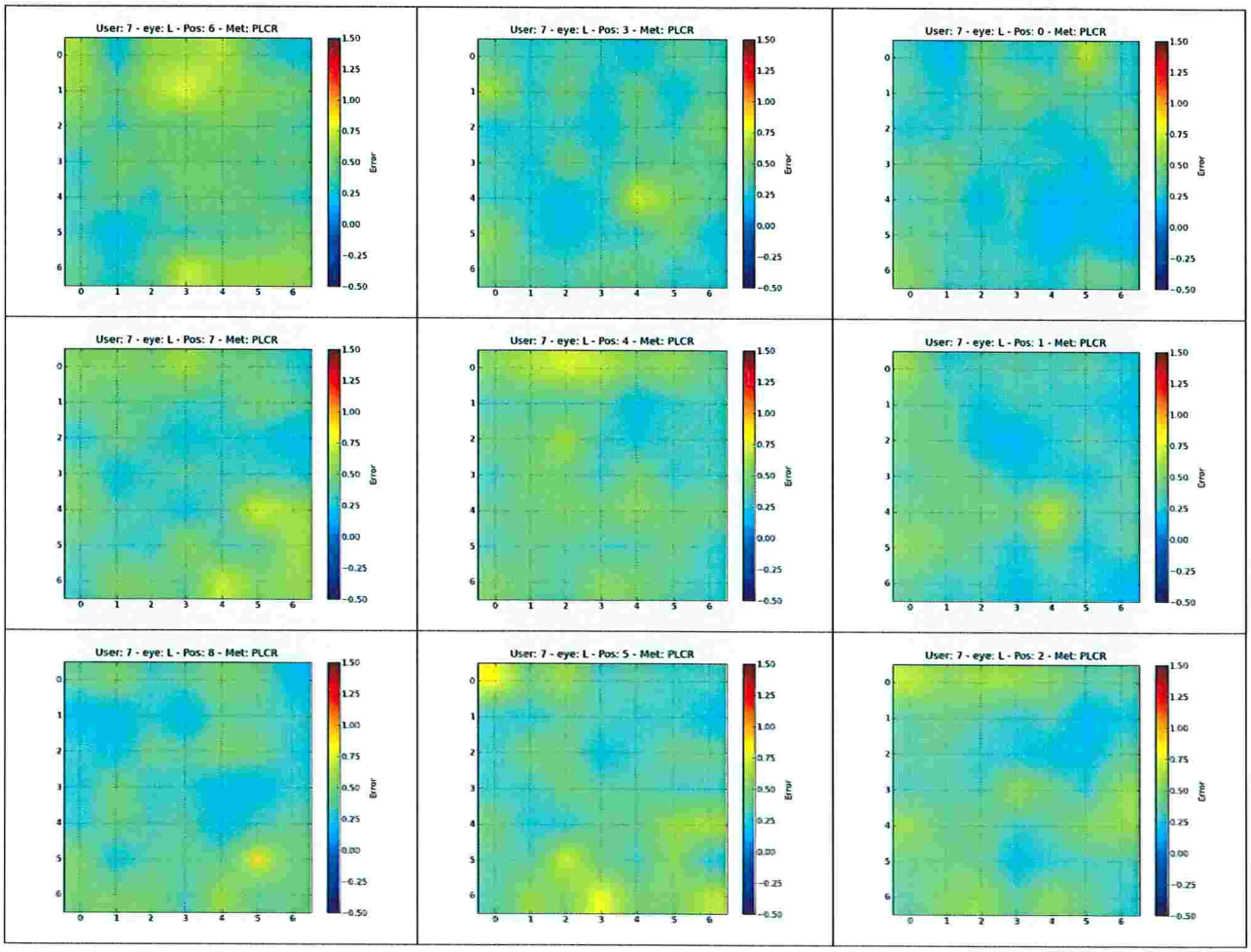

Figure B.92: *Distribution of the gaze error in all positions for the left eye of participant 7 using method* ***PL-CR****. Position 0 was used for calibration.*

Bibliography

[AKS93] A. Bandopadhay A. Kaufman e B. Shaviv. An eye tracking computer user interface. Em *Proc. of the research frontier in virtual reallity workshop*, páginas 78–84, 1993. 3, 10

[Bar05] T.S. Barcelos. Interfaces prestativas baseadas em visão computacional e informação de contexto. Dissertação de Mestrado, Departamento de Ciência da Computação do Instituto de Matemática e Estatística, Universidade de São Paulo, Junho 2005. 3

[CM06] F.L. Coutinho e C.H. Morimoto. Free head motion eye gaze tracking using a single camera and multiple light sources. Em *Computer Graphics and Image Processing, 2006. SIBGRAPI '06. 19th Brazilian Symposium on*, páginas 171 –178, Outubro 2006. 5, 7, 18, 22, 25, 26, 27, 30

[CTGJ08] Jixu Chen, Yan Tong, Wayne Gray e Qiang Ji. A robust 3d eye gaze tracking system using noise reduction. Em *ETRA '08: Proceedings of the 2008 symposium on Eye tracking research & applications*, páginas 189–196, New York, NY, USA, 2008. ACM. 12, 13

[CVC08] Juan J. Cerrolaza, Arantxa Villanueva e Rafael Cabeza. Taxonomic study of polynomial regressions applied to the calibration of video-oculographic systems. Em *Proceedings of the 2008 symposium on Eye tracking research & applications*, ETRA '08, páginas 259–266, New York, NY, USA, 2008. ACM. 11

[Duc02] Andrew T. Duchowski. A breadth-first survey of eye tracking application. *ehavior Research Methods, Instruments, and Computers (BRMIC)*, 34:455–470, 2002. 2

[Duc03] Andrew T. Duchowski. *Eye Tracking Methodology: Theory and Practice*. Springer Verlag, January 2003. 1, 10, 12

[Ebi95] Yoshinobu Ebisawa. Unconstrained pupil detection technique using two light sources and the image difference method. *Visualization and Intelligent Design in engineering and architecture*, páginas 79–89, 1995. 77

[ET98] Alessandro Verri Emanuele Trucco. *Introductory Techniques for 3-D Computer Vision*. Prentice Hall, March 1998. 33

[GE06] E.D. Guestrin e M. Eizenman. General theory of remote gaze estimation using the pupil center and corneal reflections. *Biomedical Engineering, IEEE Transactions on*, 53(6):1124 –1133, june 2006. 12, 13, 45

[GE08] Elias Daniel Guestrin e Moshe Eizenman. Remote point-of-gaze estimation requiring a single-point calibration for applications with infants. Em *ETRA '08: Proceedings of the 2008 symposium on Eye tracking research & applications*, páginas 267–274, New York, NY, USA, 2008. ACM. 5, 12, 13

[GE10] Elias D. Guestrin e Moshe Eizenman. Listing's and donders' laws and the estimation of the point-of-gaze. Em *Proceedings of the 2010 Symposium on Eye-Tracking Research & Applications*, ETRA '10, páginas 199–202, New York, NY, USA, 2010. ACM. 75

[GEKE08] Elias D. Guestrin, Moshe Eizenman, Jeffrey J. Kang e Erez Eizenman. Analysis of subject-dependent point-of-gaze estimation bias in the cross-ratios method. Em *Proceedings of the 2008 symposium on Eye tracking research & applications*, ETRA '08, páginas 237–244, New York, NY, USA, 2008. ACM. 18, 25, 30, 80

[HAV10] Dan Witzner Hansen, Javier San Agustin e Arantxa Villanueva. Homography normalization for robust gaze estimation in uncalibrated setups. Em *Proceedings of the 2010 Symposium on Eye-Tracking Research & Applications*, ETRA '10, páginas 13–20, New York, NY, USA, 2010. ACM. 5, 7, 19, 23, 24, 25, 26, 44

[HJ10] Dan Witzner Hansen e Qiang Ji. In the eye of the beholder: A survey of models for eyes and gaze. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32:478–500, 2010. 10, 11

[HNL06] C. Hennessey, B. Noureddin e P. Lawrence. A single camera eye-gaze tracking system with free head motion. Em *Proc. of the ETRA 2006*, páginas 87–94, San Diego, CA, March 2006. 12, 13

[HZ00] R. Hartley e A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, ISBN: 0521623049, 2000. 16

[JC80] Marcel A. Just e Patricia A. Carpenter. A theory of reading: From eye fixations to comprehension. *Psychological Review*, 87(4):329 – 354, 1980. 1

[KGME07] J. J. KANG, E. D. GUESTRIN, W. J. MACLEAN e M EIZENMAN. Simplifying the cross-ratios method of point-of-gaze estimation. Em *30th Canadian Medical and Biological Engineering Conference (CMBEC30)*, 2007. 5, 7, 23, 25, 26

[ME10] Dmitri Model e Moshe Eizenman. User-calibration-free remote gaze estimation system. Em *Proceedings of the 2010 Symposium on Eye-Tracking Research & Applications*, ETRA '10, páginas 29–36, New York, NY, USA, 2010. ACM. 12, 14

[MKAF99] C. H. Morimoto, D. Koons, A. Amir e M. Flickner. Frame-rate pupil detector and gaze tracker. Em *Proceedings of the IEEE ICCV'99 Frame-Rate Computer Vision Applications Workshop*, 1999. 11

[MM05] Carlos H. Morimoto e Marcio R. M. Mimica. Eye gaze tracking techniques for interactive applications. *Comput. Vis. Image Underst.*, 98(1):4–24, 2005. 10, 12

[NIK+10] Takashi Nagamatsu, Yukina Iwamoto, Junzo Kamahara, Naoki Tanaka e Michiya Yamamoto. Gaze estimation method based on an aspherical model of

the cornea: surface of revolution about the optical axis of the eye. Em *Proceedings of the 2010 Symposium on Eye-Tracking Research & Applications*, ETRA '10, páginas 255–258, New York, NY, USA, 2010. ACM. 75

[NKIT08] Takashi Nagamatsu, Junzo Kamahara, Takumi Iko e Naoki Tanaka. One-point calibration gaze tracking based on eyeball kinematics using stereo cameras. Em *ETRA '08*, páginas 95–98, 2008. 12, 13

[NSI+10] Takashi Nagamatsu, Ryuichi Sugano, Yukina Iwamoto, Junzo Kamahara e Naoki Tanaka. User-calibration-free gaze tracking with estimation of the horizontal angles between the visual and the optical axes of both eyes. Em *Proceedings of the 2010 Symposium on Eye-Tracking Research & Applications*, ETRA '10, páginas 251–254, New York, NY, USA, 2010. ACM. 12, 14

[Rob63] D. A. Robinson. A method of measuring eye movements using a scleral search coil in a magnetic field. *IEEE Trans. Bio-Med. Electronics*, 10:137–145, 1963. 3, 10

[Sel04] T. Selker. Visual attentive interfaces. *BT Technology Journal*, 22(4):146–150, 2004. 3

[SL03] S.W. Shih e J. Liu. A novel approach to 3d gaze tracking using stereo cameras. *IEEE Transactions on systems, man, and cybernetics - PART B*, páginas 1–12, Jan 2003. 12, 13

[SVS03] Jeffrey S. Shell, Roel Vertegaal e Alexander W. Skaburskis. Eyepliances: attention-seeking devices that respond to visual attention. Em *CHI Extended Abstracts*, páginas 770–771, 2003. 3

[VCP+08] A. Villanueva, R. Cabeza, S. Porta, M. Böhme, D. Droege e F. Mulvey. Report on new approaches to eye tracking. summary of new algorithms. *Communication by Gaze Interaction (COGAIN), IST-2003-511598: Deliverable 5.6*, 2008. 4, 10

[WS82] G. Wyszecki e W. S. Stiles. *Color Science: Concepts and Methods, Quantitative Data and Formulae.* John Wiley & Sons, 1982. 5

[WS02] Jian-Gang Wang e E. Sung. Study on eye gaze estimation. *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, 32(3):332 –350, jun 2002. 38

[YC05] Dong Hyun Yoo e Myung Jin Chung. A novel non-intrusive eye gaze estimation using cross-ratio under large head motion. *Computer Vision and Image Understanding*, 98(1):25 – 51, 2005. Special Issue on Eye Detection and Tracking. 5, 7, 21, 22, 25

[YKLC02] Dong Hyun Yoo, Jae Heon Kim, Bang Rae Lee e Myoung Jin Chung. Non-contact eye gaze tracking system by mapping of corneal reflections. Em *Automatic Face and Gesture Recognition, 2002. Proceedings. Fifth IEEE International Conference on*, páginas 94 –99, Maio 2002. 4, 5, 7, 15, 25

[YS75] L. Young e D. Sheena. Methods & designs: Survey of eye movement recording methods. *Behavioral Research Methods & Instrumentation*, 7(5):397–429, 1975. 10

[ZMI99] Shumin Zhai, Carlos Morimoto e Steven Ihde. Manual and gaze input cascaded (magic) pointing. Em *CHI*, páginas 246–253, 1999. 3