Universidad de Buenos Aires
Facultad de Ciencias Exactas y Naturales
Departamento de Computación

Análisis y predicción de la búsqueda visual humana

Tesis presentada para optar al título de
Licenciada en Ciencias de la Computación

Melanie Sclar

Directores: Dr. Juan Kamienkowski y Dr. Guillermo Solovey

Buenos Aires, Septiembre 2017

Análisis y predicción de la búsqueda visual humana

La búsqueda visual es una tarea clave en la vida diaria. Desde encontrar a un amigo entre una multitud hasta tomar una taza de café requieren de una exploración inteligente del espacio visual. Sin esta capacidad, no podríamos encontrar ningún objeto a menos que aparezca por azar frente a nuestros ojos. Pese a la importancia y aparente simpleza de la búsqueda visual, al día de hoy no existe un modelo capaz de predecir el recorrido de la mirada. En este trabajo nos proponemos estudiar los algoritmos y estrategias de la búsqueda visual humana en escenas naturales. Tendremos en cuenta tres aspectos: (i) la *saliencia* de los objetos en una imagen (indica regiones llamativas de la imagen por su contraste, color, orientación, etc.), estimada a partir del procesamiento de la imagen; (ii) expectativas o *priors* sobre la ubicación de los objetos (por ejemplo, es más probable *a priori* que una taza esté sobre la mesa que en el techo), estimada a partir del análisis visual y lingüístico de un corpus de imágenes; y (iii) las reglas con las que estos mapas de probabilidades de hallar el objeto se actualizan y dirigen la mirada.

Con estos ingredientes se implementaron distintos modelos, algunos que solo tuvieron en cuenta los puntos (i) y (ii) (denominados estáticos) y otros que incorporaron el punto (iii) (denominados dinámicos). Entre ellos, un modelo normativo importante es el modelo de buscador óptimo, en el cual los ojos se mueven hacia la dirección que maximiza la probabilidad de encontrar el objeto buscado. Implementamos este modelo utilizando un mapa de probabilidad que toma en cuenta explícitamente los primeros dos aspectos mencionados, logrando un nivel de predicción hasta 40 % mejor que si se emplea como mapa inicial un modelo de saliencia del estado del arte.

Para comparar los modelos de búsqueda visual se desarrollaron e implementaron distintas métricas con el objetivo de explorar y capturar distintos aspectos del recorrido de la mirada. Asimismo, fue necesario generar un conjunto de datos de búsqueda visual en escenas naturales, anotado con el reporte subjetivo de los observadores respecto de la posición del target y su confianza en la respuesta, y por observadores externos respecto del contenido de las imágenes. A lo largo de este trabajo también desarrollamos predicciones sobre las respuestas más probables del reporte subjetivo humano.

Palabras clave: Búsqueda visual, visión humana, mapa de saliencia, modelos bayesianos, métricas de comparación de scanpaths.

Analysis and prediction of human visual search

Visual search is a vital task in everyday life. From finding a friend among a crowd to having a cup of coffee, many tasks require a smart exploration of visual space. Without this ability, we would not find any object unless it appeared by chance before our eyes. Despite the importance and apparent simplicity of visual search, to date there is no model capable of predicting the path of the human gaze. In this work, we propose to study the algorithms and strategies of human visual search in natural scenes. We will take into account three aspects: (i) the *salience* of objects in an image (indicates conspicuous regions of the image by contrast, color, orientation, etc.), estimated from the image processing, (ii) expectations or priors on the location of objects (e.g., it is more likely that a cup is on the table than on the ceiling), estimated from the visual and linguistic analysis of a corpus of images, and (iii) the rules with which these maps of probabilities of finding the object are updated and direct the gaze.

Different models were implemented combining these components, some of which only took into account points (i) and (ii) (called static) and others that incorporated point (iii) (called dynamic). Among them, an important normative model is the model of the ideal bayesian observer, in which the eyes move towards the direction that maximizes the probability of finding the object searched. We implemented this model using a probability map which explicitly takes into account the first two aspects mentioned, achieving a prediction level up to 40 % better than if a state-of-art salience model is used as the initial map.

To compare visual search models, different metrics were developed and implemented with the goal of exploring and capturing different aspects of the gaze's path. It was also necessary to produce a set of visual search data in natural scenes, annotated with the subjective report of the observers regarding the position of the target and the confidence in their answer, and by external observers regarding the content of the images. Throughout this paper we also develop predictions about the most likely responses of human subjective reporting.

Keywords: Visual search, human vision, saliency map, bayesian models, scanpath comparison metrics.

Agradecimientos

A Carlos Diuk y Alejo Salles, por aceptar ser jurados y por tomarse el tiempo y el trabajo de leer esta tesis.

A mis directores, Juan Kamienkowski y Guillermo Solovey, por introducirme al área de la Inteligencia Artificial y guiarme durante el proceso de este trabajo.

A mi mamá y a mi papá, por apoyarme siempre en todas mis decisiones y proyectos. Gracias por ser mi ejemplo de amor, de generosidad y de saber enfrentarse a los problemas con templanza.

A Ariel Zylber y Agustín Gutiérrez, sin quienes probablemente no hubiera estudiado esta carrera, y sin quienes todos estos años no hubieran sido igual de buenos. Gracias por mostrarme el mundo de la programación y de la programación competitiva.

A Matías Saucedo y Sebastián Prillo, que cada uno a su manera se dedican con pasión a mejorar el espacio que nos vio crecer y donde surgió nuestra amistad inquebrantable.

A Miguel Maurizio, cuya amistad y compañerismo ya pasó a ser no solo entre nosotros sino entre nuestras familias.

A todos los amigos que me dejó la olimpíada y con los que seguí compartiendo momentos hasta hoy. A algunos de ellos llegué a conocerlos más durante la carrera: a Franco, Margarita, Mariano, Iván, Caro González, Caro Lang, Leopoldo, Alan, entre muchos otros.

A Nuria Madrid de Susmel, por mostrarme que la pasión por la matemática y las ganas de transmitirla no tienen límites. Además, por enseñarme que aún más importante que transmitir pasión por la matemática es dar amor y cariño hacia quienes nos rodean.

A la Olimpíada de Matemática, a la ACM-ICPC y a todas las personas que hacen posible que año a año se lleven a cabo. Gracias por haber tocado las vidas de tantos chicos y en especial, gracias por haber cambiado la mía. Sin ellas, hoy no sería la persona que soy.

Índice general

Capítulo 1

Motivación

Uno de los desafíos centrales de la ciencia cognitiva y de la neurociencia de la visión es entender cómo percibimos una escena visual. En los años sesenta, los trabajos pioneros de Yarbus mostraron que cuando exploramos una escena nuestros ojos realizan una secuencia sistemática de movimientos rápidos [Yar67]. Más de 50 años después de las primeras exploraciones cuantitativas de los movimientos oculares, aún no hay una teoría formal de la interacción entre la percepción y los movimientos oculares [Rol15, TWK+10]. Tampoco existe un algoritmo que indique, *a priori*, si una tarea de búsqueda visual será fácil o difícil [Rol15, TWK+10].

En materia de modelos han habido enormes avances en el procesamiento de imágenes y sobre todo en la comprensión automática del contenido de la imagen [JDT12, BI15, BCJL15, VTBE15]. Estos modelos han crecido en gran parte por una fuerte interacción entre las ciencias cognitivas, neurociencias y ciencias de la computación.

Una idea que se ha fortalecido en la última década a partir de esta interacción es la aplicación de un marco bayesiano para explicar cómo el cerebro es capaz de generalizar y realizar inferencias sobre entornos ruidosos y saturados de información. En otras palabras, de qué forma los humanos -y muchos otros animales- son capaces de ir más allá de los datos para construir modelos completos y abstractos del entorno [TGK06]. Los modelos bayesianos hoy cubren un amplio espectro de aspectos cognitivos como la toma de decisiones y la confianza, distintos tipos de aprendizaje, percepción multisensorial, entre otros [TGK06, GKT08, MSM15].

El presente proyecto se propone dar un paso adelante en este camino combinando estos modelos para comprender a nivel algorítmico el proceso de búsqueda visual en imágenes naturales.

Capítulo 2

Introducción

2.1. Movimientos oculares y búsqueda visual

La retina es una capa fina (∼0.25 mm) de tejido neural que recubre la parte posterior del ojo. Tiene células fotorreceptoras sensibles a la luz - los bastones y los conos. En su superficie se pueden observar varias estructuras, y una de ellas es la *fóvea*. La fóvea es el área de la retina donde se enfocan los rayos luminosos del centro del campo visual y posee muchos conos, responsables de la percepción de colores. La fóvea es responsable de la visión central aguda (también llamada visión foveal), que es necesaria en los humanos para las actividades donde los detalles visuales son de importancia. La fóvea está rodeada por la parafóvea y la región perifóvea externa [Dow07].

Si bien tenemos la impresión de que podemos procesar todo el campo visual fijando la vista en un único lugar, en realidad no seríamos capaces de procesar completamente la información fuera de la visión foveal si no pudiéramos mover nuestros ojos [RC07].

Debido a las limitaciones de agudeza en la retina, los movimientos oculares son necesarios para procesar detalles. Nuestra capacidad de discriminar los detalles finos cae marcadamente fuera de la fóvea en la parafóvea (que se extiende a unos 5 grados a cada lado) y en la periferia (más allá de la parafóvea). En la figura 2.1 se puede ver un gráfico de agudeza visual en visión normal.

Existen dos categorías básicas de movimientos oculares: las sacadas y las persecuciones suaves. Las movimientos sacádicos son movimientos rápidos que cambian la mirada de un punto a otro. Varían en amplitud según la tarea, yendo desde los pequeños movimientos hechos durante la lectura hasta los movimientos amplios hechos durante la observación de una habitación [PAF01]. Los movimientos

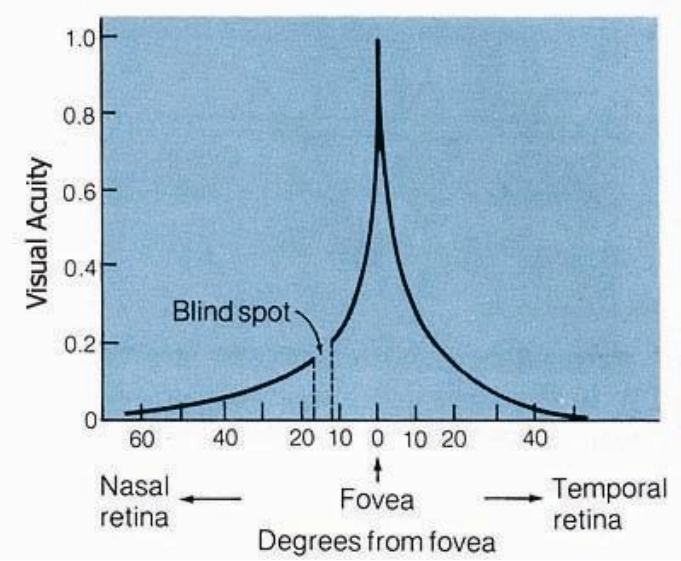


Figura 2.1: Agudeza visual en función de la distancia en grados a la fóvea de un ojo izquierdo con visión normal. Como se puede ver, decae rápidamente fuera de la fóvea. Gráfico extraído de [RC07].

de persecución suave siguen el movimiento de un objeto.

Los movimientos de orientación grandes implican la acción coordinada de los ojos, la cabeza y el cuerpo, pero los movimientos más pequeños, como los que se hacen al mirar una imagen, se hacen con los ojos solamente [Yar67]. Los comandos cerebrales enviados a los músculos del ojo dan como resultado que los ojos hagan una rápida rotación, después de la cual los ojos permanecen estacionarios en su nueva posición. Estos movimientos rápidos son las sacadas y cada posición fija de los ojos posterior a una sacada se denomina *fijación*. Las sacadas dirigen la fóvea hacia un objeto o región de interés que permite un análisis visual detallado posterior de alta agudeza en ese lugar. Los humanos que poseen visión normal hacen varias sacadas cada segundo y sus destinos son seleccionados por un proceso cognitivo del cerebro que no involucra la consciencia de los sujetos [FW12].

El movimiento ocular normalmente alterna entre fijaciones y sacadas. El conjunto de fijaciones y sacadas alternadas durante una observación se denomina *scanpath*, pues es el camino recorrido por la vista al explorar una imagen.

Hace más de medio siglo que se sabe que las ubicaciones de las fijaciones no son aleatorias. Más aún, se ha estudiado que los patrones que describe la vista son distintos según la tarea que se realice [Yar67, CMH09]. Un ejemplo concreto se puede ver en la figura 2.2, que muestra fijaciones de dos personas haciendo dos tareas distintas: memorizar una imagen o buscar un objeto en ella. En la primera tarea el observador se focaliza en cubrir la mayor parte de la imagen con la vista mientras que en la segunda se concentra en encontrar un objeto particular, buscándolo en

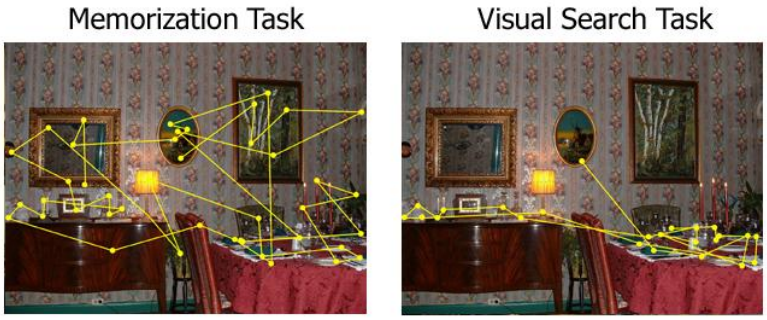


Figura 2.2: Ejemplo extraído de [CMH09]. A la izquierda una persona está intentando memorizar la escena para un test posterior. A la derecha, otra persona está examinando la misma escena pero se le pidió encontrar una taza de café. Los puntos amarillos representan las fijaciones y los segmentos entre ellos, las sacadas.

los lugares más probables. Aún cuando no se le dé ninguna tarea en particular al observador, los puntos fijados no son aleatorios: por ejemplo, en la presencia de imágenes de caras de personas la visión se enfoca en los ojos, la boca y la nariz de las mismas, que son puntos informativos para formar una visión global.

Además, existen indicios de que la ubicación de las fijaciones y la duración de las mismas podría depender del *background* cultural del sujeto [RCY09, CBN05].

2.2. Modelos de saliencia

Evolutivamente es importante para los sistemas biológicos complejos poder detectar rápidamente una potencial presa, predador o parejas en un mundo visual lleno de objetos. Sin embargo, identificar simultáneamente todos los objetivos interesantes en un campo visual es computacionalmente imposible hasta para los cerebros biológicos más complejos e incluso para computadoras. Una solución, adoptada por los primates y otros animales, es restringir los procesos complejos a un área reducida que es considerada interesante o a algunos objetos por vez. Así, después se pueden analizar todos los objetos en una escena visual uno detrás del otro. Esta serialización de una escena es operacionalizada por mecanismos de atención visual. Sin embargo, esta solución a la imposibilidad de procesar una escena en paralelo da lugar a un nuevo problema: ¿cómo decidimos qué región seleccionar para captar nuestra atención primero y analizarla más en detalle? La *saliencia visual* permite al cerebro realizar esta selección con una eficiencia razonable, y hemos evolucionado para computarla de manera automática y sobre todo el campo visual. Así, la atención visual es atraída hacia los lugares más salientes visualmente. La *saliencia visual* es la cualidad perceptual subjetiva que hace que algunos ítems del mundo se destaquen respecto de sus vecinos y capten nuestra atención [Itt07].

(a) (b)

Figura 2.3: Imagen original (a) y su mapa de saliencia predicho por MLNet (b). El nivel de saliencia es representado por la claridad de cada píxel: cuanto más claro, más saliente es la región.

Por ser un mecanismo biológico tan importante y por sus aplicaciones en visión computacional es que surge la necesidad de entender cuáles son las regiones más salientes para el ojo humano. En las últimas décadas se realizaron varios trabajos creando diversos *mapas de saliencia* sobre imágenes: estos son mapas bidimensionales que explícitamente codifican la saliencia de los objetos en un ambiente visual. Nos focalizaremos en los mapas de saliencia para imágenes. En la figura 2.3 se pueden ver dos ejemplos de mapas de saliencia para imágenes. Según qué imagen se analice, los mapas de saliencia tienen diferentes niveles de exactitud en la predicción de las fijaciones humanas. Los modelos de saliencia aún no logran predecir las fijaciones humanas con exactitud en todo tipo de imágenes [JEDT09], y existen indicios de que tal vez nunca lo logren si se toman todas las fijaciones y no únicamente las primeras que realizó el sujeto [HBCM07].

Dentro de los modelos de saliencia encontramos dos categorías: *bottom-up* y *top-down*. La saliencia visual *bottom-up* es aquella en la que la región que capta la atención del sujeto es puramente determinada por las propiedades del estímulo visual presentado, ya sea porque el contraste que tiene una región respecto al resto de la imagen o porque el tipo de imagen presentada llama especialmente la atención de los humanos (por ejemplo, caras humanas o texto). La mayoría de los modelos de saliencia se concentran puramente en identificar características de esta categoría y combinarlas. Sin embargo, según la tarea que estemos realizando puede ocurrir que los sujetos ignoren los lugares inherentemente más salientes en pos de su objetivo. Por ejemplo, si estamos buscando crayones verdes en una caja con muchos colores diferentes los componentes *bottom-up* serán ruido y prevalecerá la búsqueda por regiones verdes de la imagen. Esta búsqueda está dirigida por el usuario, y por eso se denomina *top-down* [Wol94]. Este efecto se relaciona con que los scanpath descritos

por los sujetos dependen de la tarea que estén realizando, y esto es especialmente cierto para las tareas de búsqueda visual.

Si bien hemos mencionado que la mayoría de los mapas de saliencia se concentran en propiedades *bottom-up*, en tareas de búsqueda visual es especialmente cierto que los factores *top-down* pueden afectar las decisiones de los sujetos [TOCH06, DD95].

Veremos a continuación un resumen de algunos modelos de saliencia, algunos *bottom-up* y otros combinando al enfoque *bottom-up* con el *top-down*. En los últimos años, la utilización de *machine learning* mejoró sensiblemente los modelos aprendiendo de las fijaciones realizadas por humanos en imágenes [CBSC16a, JEDT09], aunque lo hacen de diferentes maneras: SAM y MLNet son modelos altamente sofisticados de *deep learning*, mientras que Judd utiliza un modelo de *machine learning* clásico que permite realizar extensiones fácilmente.

2.2.1. Modelo de Judd y colaboradores

Judd realiza un mapa de saliencia por imagen tomando las fijaciones de los humanos y suavizándolas con un filtro gaussiano. Así se tiene un mapa de saliencia que es tomado como *ground truth* para entrenar su modelo. En la figura 2.4 se puede ver un ejemplo de una de las 1003 imágenes de su dataset.

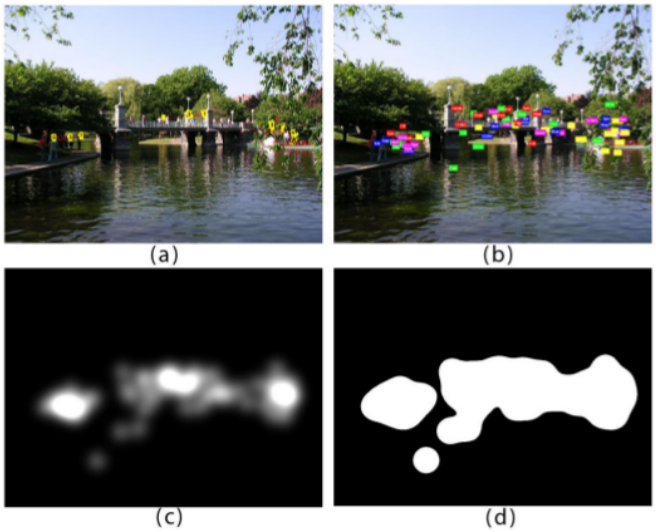


Figura 2.4: Ejemplo extraído de [JEDT09]. En (a) se ve la imagen original, en (b) las fijaciones y en (c) el mapa de saliencia creado con las fijaciones humanas. Este mapa puede ser binarizado, de forma tal de tomar un 20 % más saliente de la imagen como la región saliente y el 80 % restante como la región no saliente. Esto es lo que se ve en (d), y como el lector puede observar el umbral de saliencia puede fijarse en cualquier constante menor a 100.

Judd toma su *dataset* de 1003 imágenes y lo divide en un conjunto de entrenamiento *training set* de 903 imágenes y un *test set* de 100 imágenes. De cada imagen elige 10 píxeles al azar tomados del 20 % más saliente del mapa de saliencia *ground truth* (ver figura 2.4) y los identifica como píxeles salientes o "positivos". De igual manera elige 10 píxeles del 30 % menos saliente y los clasifica como píxeles "negativos". Además, elige píxeles que estén a por lo menos 10 píxeles del borde de la imagen.

Luego, computa 33 features para cada imagen. Estos features se clasifican en tres niveles según su nivel de complejidad e incluyen: tres mapas de colores (un canal verde, uno rojo y otro azul), un mapa de intensidad, otro de orientación y otro de contraste del color según se computa en [IK00], un mapa de distancia al centro de la imagen, un mapa que computa la distancia al horizonte estimado en la imagen, un mapa de detección de personas y otro de autos, etcétera. Se pueden ver los 33 canales descritos en detalle en [JEDT09].

Finalmente entrena una una *Support Vector Machine* (SVM) con las muestras positivas y negativas mencionadas anteriormente. Utiliza un kernel lineal pues experimentalmente usar un kernel RBF o usar *multiple kernel learning* no mejoró la performance del modelo e incrementaría el tiempo de cómputo.

Una gran ventaja de este método es que pueden agregarse o removerse features fácilmente, pudiendo generar distintos modelos a partir de una misma estructura. Esto es interesante en el contexto del presente trabajo ya que permite evaluar la contribución de distintos features, así como también incorporar otros relacionados a la tarea a realizar por el observador.

2.2.2. SAM

El *Saliency Attentive Model* (SAM) [CBSC16b] es un modelo de saliencia basado en *deep learning*. Este modelo utiliza una *Long Short-Term Memory network* (LSTM) para procesar recurrentemtente las features de saliencia en diferentes lugares de las imágenes, atendiendo selectivamente a diferentes regiones de un tensor. Esto permite refinar iterativamente el mapa de saliencia. Para extraer mapas de features de las imágenes de entrada utilizan un modelo de Redes Neuronales Convolucionales (CNN).

2.2.3. MLNet

El *Deep Multi-Level Network* (MLNet) [CBSC16a] es un modelo de saliencia basado en *deep learning*, y está compuesto por tres partes: dada una imagen de

entrada, una red neuronal convolucional (CNN) extrae features de bajo, medio y alto nivel. Luego, una red de codificación construye features específicas de saliencia, pesando mapas de features de los tres niveles y produciendo un mapa de saliencia temporario como resultado. Finalmente, un mapa previamente aprendido es combinado con el mapa temporario para producir la predicción de saliencia final.

2.3. Búsqueda visual

La búsqueda visual es la tarea común de buscar algo en un ambiente visual atestado de objetos. El ítem que busca el observador se denomina *target* (u objetivo, en español), mientras que los elementos no objetivo se denominan *distractores*. A veces la palabra distractor es utilizada más específicamente para referirse a los elementos no objetivo que guardan similitud con el target.

Como se pudo ver en la figura 2.2, los scanpaths que los humanos describen en tareas de búsqueda visual son diferentes a los de otras tareas. Esto tiene que ver con que la atención suele favorecer factores *top-down* por sobre los factores *bottom-up*, que son los que la saliencia visual describe tradicionalmente. La literatura muestra que en tareas de búsqueda visual hay estímulos muy salientes que son ignorados por no ser relevantes para la tarea, pero esto no ocurre para todos los estímulos visuales [WH08].

Si bien la búsqueda visual es realizada sin esfuerzo por los humanos, recrearla en máquinas ha representado y aún representa un desafío para la ciencia. El análisis de la distribución de sacadas y la latencia en las sacadas ha contribuido en el entendimiento actual de la búsqueda visual humana. Las sacadas muestran evidencia de elementos tanto *bottom-up* como *top-down* guiando la búsqueda, y muestran factores que facilitan la búsqueda como es el contexto de una escena [TOCH06].

Debemos diferenciar en este punto las tareas de búsqueda visual, distinguiendo si se realiza sobre escenas naturales o sobre escenas artificiales. Los factores contextuales y semánticos de la imagen son más relevantes en escenas naturales, mientras que en escenas artificiales suelen funcionar mejor los enfoques que se centran en predecir fijaciones en las posiciones más salientes de una imagen (ya que la misma provee poca información semántica). En otras palabras, los modelos de saliencia pueden predecir mejor las fijaciones en escenas artificiales que en naturales. La saliencia sigue siendo un factor relevante a considerar en modelos de búsqueda visual sobre escenas naturales, pero en este caso se la suele combinar con mapas de features específicos de la tarea, como por ejemplo un mapa de apariencia esperada del target [KTZC09] o un mapa de la ubicación esperada del objeto [TOCH06], entre

otras representaciones.

2.4. Modelos de movimientos oculares en imágenes

Existen diversos modelos de movimientos oculares en imágenes que buscan modelar cómo sería el recorrido de la vista en una tarea de búsqueda visual. Algunos de esos modelos se aprovechan explícitamente de características humanas en tareas de búsqueda visual tal como una longitud moderada de las sacadas o la tendencia a no realizar fijaciones en ubicaciones que fueron observadas recientemente (llamado *inhibición de retorno*).

Por otro lado, Najemnik y Geisler [NG05] desarrollaron un modelo de movimientos oculares donde crearon un observador bayesiano ideal sin utilizar estas características como parámetros, pero obteniéndolas como un efecto secundario. En su trabajo observan cómo este observador ideal se parece a los movimientos oculares realizados por los humanos, sugiriendo que evolucionamos para realizar tareas de búsqueda muy eficientemente ya que esto sería esencial para la supervivencia. La tarea de búsqueda visual en la que Najemnik & Geisler experimentaron el modelo fue una tarea de búsqueda visual en una escena artificial: toda la imagen está compuesta por ruido gaussiano y en una de las 25 posiciones predeterminadas aparece el target, que es una gradilla de onda senoidal (ver figura 2.5). Como se puede ver, esta tarea imposibilita tomar features tanto de similitud del target con cada parte de la imagen como features de información contextual.

A continuación veremos en detalle cómo es este modelo bayesiano que utilizaremos a lo largo del trabajo. Lo pondremos a prueba en escenas naturales y lo modificaremos extrayendo información adicional.

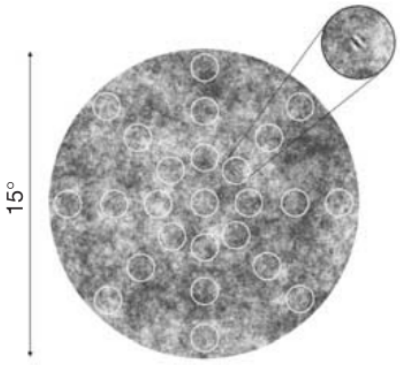


Figura 2.5: Ejemplo de ensayo de la tarea de [NG05].

Para entender el modelo, es importante entender el concepto de *mapa de visibilidad*. La visibilidad de un target varía dependiendo en qué zona se encuentre del campo visual, el nivel de contraste del target y el nivel de contraste del fondo. Un mapa de visibilidad (denominado d') muestra el nivel de visibilidad de todos los puntos del campo visual según dónde se esté fijando la vista. La visibilidad tiene su punto máximo en el centro de la fóvea y cae suavemente al aumentar la distancia a la

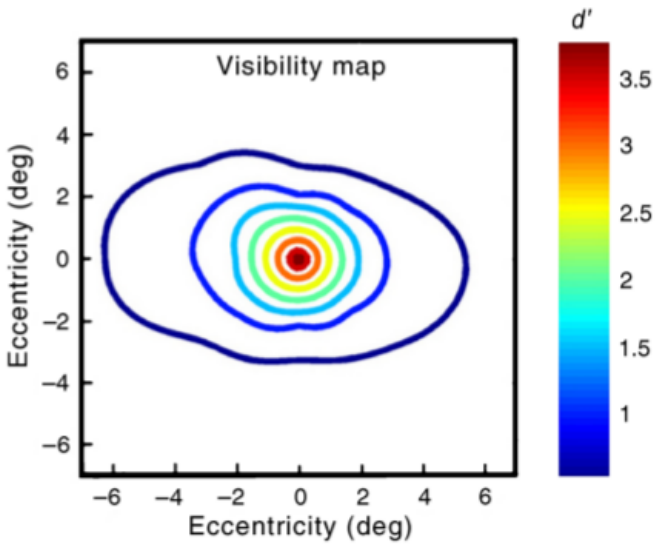


Figura 2.6: Ejemplo de mapa de visibilidad. Extraído de [NG08].

fóvea. La visibilidad cae más rápico en dirección vertical, y por lo tanto la visibilidad es peor en los campos visuales inferior y superior [NG08]. Existe evidencia de que los mapas de visibilidad tienen variaciones entre sujetos [BAG14].

Para integrar las respuestas a lo largo de las fijaciones de forma óptima, el observador ideal acumula las respuestas ponderadas de cada posible ubicación del target y calcula la probabilidad $p_i(T)$ de que cada punto i sea la posición de la fijación T-ésima:

$$s_i(T) = prior(i) \cdot \prod_{t=1}^{T} exp\left(d'^2_{ik(t)} W_{ik(t)}\right)$$

$$p_i(T) = \frac{s_i(T)}{\sum_{j=1}^{n} s_j(T)}$$

donde t es el número de fijación y $d'_{ik(t)}$ y $W_{ik(t)}$ son el mapa de visibilidad y la respuesta del target en la ubicación i cuando el sujeto se encuentra fijando la vista en $k(t)$. T es el número de fijación actual y $k(t)$ representa la ubicación de la fijación en tiempo t. La definición de W será dada más abajo.

Una vez calculada la probabilidad de que el target esté en cada posición, se computa la próxima fijación a realizar considerando cada posible ubicación y selecciona la posición que, dado su conocimiento de las actuales probabilidades *a posteriori* y el mapa de visibilidad maximizará la probabilidad de identificar correctamente la ubicación del target luego de la fijación:

$$k_{opt}(T+1) = \arg\max_{k(T+1)} \sum_{i=1}^{n} p_i(T) p(C|i, k(T+1))$$

donde $p(C|i, k(T+1))$ es la probabilidad de identificar correctamente el target dado que el mismo se encuentra es i y la ubicación de la próxima fijación es $k(T+1)$. $p(C|i, k(T+1))$ puede calcularse de la siguiente forma:

$$p(C|i, k(T+1)) = \int_{-\infty}^{+\infty} \phi(w) \prod_{j \neq i} \Phi \left(\frac{2d'_{ik(T+1)} w - 2 \ln \frac{p_j(T)}{p_i(T)} + d'^2_{jk(T+1)} + d'^2_{ik(T+1)}}{2d'_{jk(T+1)}} \right) dw$$

La derivación de esta fórmula se puede ver en el material suplementario de [NG05]. En esta fórmula ϕ es la función de densidad de la normal estándar y Φ es su función de distribución.

Es de relevancia entender cómo se modela W. $W_{ik(t)}$ es la respuesta del template de la i-ésima posible ubicación dado que el sujeto está fijando en $k(t)$ en la fijación número t. En otras palabras, representa cuán parecida es la posición i al target para el observador que se encuentra fijando su vista en $k(t)$. $W_{ik(t)}$ tendrá distribución Gaussiana y sin pérdida de generalidad se fija el valor esperado de $W_{ik(t)}$ como 0,5 cuando el target está presente en esa ubicación y $-0{,}5$ en caso contrario. Además, la varianza de $W_{ik(t)}$ se define como la inversa de la visibilidad para representar que cuanto menos visible sea la ubicación i, menos precisión tendrá el sujeto sobre su observación y por ende puede cometer errores con más frecuencia.

Así, $W_{ik(t)} \in \mathcal{N}(\mu_{ik(t)}, \sigma^2_{ik(t)})$ donde:

$$\mu_{ik(t)} = \begin{cases} 0{,}5 & si\ i = \ ubicación\ del\ target \\ -0{,}5 & caso\ contrario \end{cases}$$

$$\sigma^2_{ik(t)} = \frac{1}{d'^2_{ik(t)}}$$

Este modelo logra inhibición de retorno así como longitudes de sacadas moderadas. Se logra inhibición de retorno porque suponiendo que ya se observó en la posición i, no se encontró el target y se considera volver a fijar allí, podemos deducir que $\mu_{ik(t)} = -0{,}5$, que $k(t) = i$ para algún $t \leq T$ y por ende $W_{ik(t)} = W_{ii} \in \mathcal{N}\Big(-0{,}5, \frac{1}{d'^2_{ii}}\Big)$. Como la visibilidad es máxima cuando el punto a considerar es el

mismo que está siendo actualmente observado se tiene que W_{ii} tendrá muy poca varianza entorno a su valor esperado de $-0{,}5$. Esto implicará que $exp\left(d'^2_{ii} W_{ii}\right) > 0$ sea un valor reducido, dando lugar a que $p_i(T)$ sea ínfimo.

Más aún, se logra el efecto de longitudes de sacada moderadas ya que una sacada muy corta implicaría una visibilidad muy alta de la próxima fijación, dando muy poca varianza a $W_{ik(T)}$ entorno a su valor esperado de $-0{,}5$ (asumiendo que el target no está allí) y haciendo que las probabilidades a posteriori de las ubicaciones cercanas sean reducidas. Por el contrario, la probabilidad a posteriori de las ubicaciones muy lejanas no tienden a incrementarse ya que una sacada muy larga haría que $d'_{ik(T)}$ sea muy cercano a cero.

Una propiedad interesante del modelo es que por su naturaleza probabilística es capaz de predecir varias sucesiones de movimientos oculares según como se fije W.

2.5. Métodos de comparación de scanpaths

Dados uno o más modelos de movimientos oculares, es de interés entender cuán similares son los recorridos visuales a los que realizaría un humano. Cada uno de estos recorridos visuales es llamado *scanpath* en la literatura. No hay consenso sobre qué método usar para comparar scanpaths, y distintos trabajos presentan distintas métricas. Cada una de estas, a su vez, captura diferentes propiedades de los recorridos. A continuación presentamos algunas de las más utilizadas.

- *Cantidad de fijaciones hasta encontrar el target.* En muchas tareas la cantidad de fijaciones para encontrar el target no varía demasiado de sujeto a sujeto, permitiendo utilizar esto como métrica [NG05, CGDLB08]. Típicamente en estos trabajos la cantidad de fijaciones esperada para encontrar el target es la misma en todos los ensayos. En el presente trabajo extenderemos la métrica para tareas donde esta propiedad no se cumple, como suele ser el caso de ensayos con escenas naturales.

- *String edit distance.* Esta métrica se basa en dividir la imagen en secciones de interés (ROI, por sus siglas en inglés), asignarle a cada ROI un símbolo y luego transformar una sucesión de fijaciones en una sucesión de símbolos, donde cada símbolo corresponde a la ROI a la que pertenece cada fijación. A continuación se pueden comparar dos sucesiones de fijaciones (ahora, dos tiras de símbolos) calculando la mínima cantidad de operaciones requeridas para obtener un string partiendo de otro. Según qué operaciones se permitan,

se obtienen distintas métricas de *edit distance*. La métrica de edición de strings más utilizada es la distancia Levenshtein [LMB13, BS97], que entiende como operación a la inserción, eliminación o sustitución de un caracter. Formalmente, $\text{lev}_{a,b}(i,j)$ es la distancia Levenshtein entre los primeros i caracteres del string a y los primeros j caracteres del string b y se define como:

$$\text{lev}_{a,b}(i,j) = \begin{cases} \text{máx}(i,j) & \text{si mín}(i,j) = 0, \\ \text{mín} \begin{cases} \text{lev}_{a,b}(i-1,j) + 1 \\ \text{lev}_{a,b}(i,j-1) + 1 \\ \text{lev}_{a,b}(i-1,j-1) + 1_{(a_i \neq b_j)} \end{cases} & \text{caso contrario} \end{cases}$$

Existen otras distancias de strings que suelen considerarse, según qué operaciones se permitan y el costo que se asigne a cada una de ellas. Algunas de las métricas más conocidas son la *Longest Common Subsequence (LCS)*, *Damerau-Levenshtein distance*, *Hamming distance*, entre otras.

Un problema a considerar es cómo definir las ROI. Algunos trabajos se abocan a definir ROI según su semántica en la imagen [PS00], mientras que otros crean una grilla rectangular donde cada casilla se considera una sección. El primer enfoque es severamente más complejo en escenas naturales y es por eso que se optó por la segunda opción en este trabajo.

- *Ángulos entre fijaciones*. Se ha visto que los ángulos realizados entre fijaciones tienen patrones particulares [CGDLB08]. Cuando el modelo no utiliza explícitamente esto como parámetro, se puede utilizar como métrica de comparación de scanpaths. En 5.2.2 podremos ver el comportamiento para nuestra tarea.

- *Mannan distance*. La distancia de Mannan es una métrica de comparación de dos scanpaths que contrasta las regiones fijadas por cada uno de ellos ignorando el orden en que se realizaron las fijaciones. Más en detalle, esta métrica calcula la similitud entre dos scanpaths computando la distancia entre cada fijación en un scanpath y su vecino más cercano en el otro scanpath [LMB13]. Esto implica que a más de una fijación puede asignarle el mismo vecino más cercano. Si bien esto es una ventaja ya que puede utilizarse la métrica a scanpaths de longitudes diferentes, se ha mostrado que la distancia de Mannan no se comporta bien cuando el número de fijaciones en los dos scanpaths es extremadamente diferente.

Capítulo 3

Objetivos

Como se ha mencionado en la introducción, al observar una imagen natural y en particular al buscar un objeto en ella, los puntos en los que se detiene la mirada no son al azar. En cambio, estos reflejan propiedades relevantes de la imagen y la tarea. El objetivo general del presente trabajo es comprender qué propiedades son las que guían los movimientos oculares, y qué reglas se utilizan para decidir dónde fijar la mirada en base a dichas propiedades. Para ello se construirán modelos capaces de predecir el comportamiento de la mirada sobre imágenes no analizadas previamente. En particular, este proyecto involucra distintas etapas, que describimos a continuación.

En primer lugar, se deberá generar un corpus de datos de movimientos oculares sobre imágenes naturales en una tarea de búsqueda visual. Esto es necesario ya que existen diversas bases de datos de movimientos oculares sobre escenas naturales en tareas de observación pero no existen, en nuestro conocimiento, bases de datos en tareas de búsqueda visual con las características requeridas. Por otro lado, con el objetivo de evaluar la información disponible para el observador respecto de la posición del target, se realizarán manipulaciones sobre la tarea comportamental.

En segundo lugar, se implementarán modelos de saliencia para predecir la posición de las fijaciones en cada imagen. Se compararán distintos métodos del estado del arte en el tema, el mejor de los cuales se utilizará en la siguiente etapa.

Sin embargo, los modelos antes mencionados no son capaces de predecir la secuencia de dichas fijaciones. Por lo que, aunque muy interesantes desde el punto de vista computacional, resultan incompletos como modelos de la cognición humana. Tomando las ideas pioneras de Najemnik y Geisler, que a pesar de su amplia repercusión no fueron replicadas en imágenes naturales, el objetivo final del presente trabajo es combinar los modelos de saliencia con un protocolo bayesiano para

establecer los puntos de fijación futuros.

Para ello, es necesario ser capaces de comparar secuencias de puntos de fijación. Dado que este no es un problema que tenga una única solución, en función del aspecto de la secuencia que sea de interés, se implementarán y compararán distintas métricas. Además, estas métricas tuvieron que ser adaptadas para nuestra tarea y para poder comparar una secuencia de puntos de fijación con un conjunto de secuencias, lo que no era posible utilizando las métricas encontradas en la literatura sin ninguna modificación.

Capítulo 4

Métodos

Como no existe un dataset abierto con una tarea similar a la que esbozamos en Objetivos y detallaremos a continuación, nos vimos obligados a recolectar un corpus de imágenes de interiores e implementar la tarea de búsqueda visual nosotros mismos, así como también tomar los datos sobre un conjunto de sujetos. Describiremos primero el protocolo de recolección de imágenes para la construcción del corpus de imágenes con sus respectivos target y luego detallaremos el método utilizado en la tarea.

Finalmente, vimos la necesidad de entender la clase de objeto a la que pertenece cada uno de los targets para poder desarrollar un modelo predictivo eficiente. Para esto desarrollamos una tarea *online* en la que los usuarios ingresan una descripción breve del objeto que están viendo. El detalle de esta tarea se explica hacia el final de este capítulo.

Todos los datos recolectados se hacen públicos junto con la publicación de este trabajo.

4.1. Corpus de Interiores

Recolectamos 134 imágenes de interiores de Wikimedia commons, LabelMe y blogs de diseño de interiores. Las imágenes fueron seleccionadas de forma tal que tuvieran muchos objetos en ellas y así las búsquedas tomaran varias fijaciones. También nos aseguramos de que no aparecieran figuras humanas ni texto, dado que estos dos elementos serían distractores en las búsquedas [JEDT09]. Las dimensiones originales de todas las imágenes son de 768×1024 píxeles o superiores, y fueron reescaladas y recortadas para que todo el corpus fuera de 768×1024 píxeles.

Además, recortamos entre 1 y 4 targets posibles por imagen, de diferentes

dimensiones pero todos cuadrados. En la selección de los target tuvimos especial cuidado en evitar elegir objetos de los cuales hubiera varias copias muy similares en la imagen, ya que el objetivo del trabajo no es el de apelar a la memoria perfecta de los sujetos.

De los hasta 4 target recortados por imagen se seleccionó uno aleatoriamente por imagen entre aquellos que fueran de 72×72 píxeles o menores. Para uniformizar dimensiones se expandió la región considerada del target a 72×72 píxeles de la imagen original centrados en el centro del target originalmente recortado.

4.2. Experimento de búsqueda visual

4.2.1. Participantes

30 sujetos participaron del experimento (4 mujeres, 26 hombres), todos ellos estudiantes o docentes de la Facultad de Ciencias Exactas y Naturales de la Universidad de Buenos Aires. Se descartaron los datos de dos sujetos por problemas técnicos en el guardado de los datos, quedando un dataset de 28 sujetos entre 19 y 34 años (3 mujeres, 25 varones), de edad promedio 25 años. Además, 4 sujetos realizaron el experimento en dos partes debido a un *memory leak* en el software desarrollado para el experimento. El error fue corregido para los demás sujetos. Los participantes no recibieron remuneración económica.

4.2.2. Procedimiento y adquisición de datos

Todos los participantes se sentaron a una distancia de aproximadamente 55 cm de un monitor de 17 pulgadas con resolución 1280×1024 en una habitación a oscuras y utilizaron una mentonera para estabilizar su cabeza de modo tal que los ojos miraran directamente al tercio superior de la pantalla. Se adquirió la posición de la pupila con un equipo de seguimiento ocular Eye Link 1000 (SR Research, Ontario, Canadá). El mismo estaba conectado a una computadora dedicada que almacenaba los datos y realizaba la detección de sacadas *online*. La adquisición de datos se realizó de forma monocular y a 1000Hz. La deteción se sacadas se realizó con el algoritmo nativo de EyeLink, utilizando los parámetros recomendados para tareas cognitivas.

4.2.3. Descripción de la tarea

El experimento consta de 134 ensayos divididos en bloques de 45, 45 y 44 ensayos respectivamente. Al comienzo de cada bloque se efectúa una calibración con 9 puntos para referenciar la detección de la pupila a las coordenadas en la pantalla.

Antes de cada ensayo aparece un punto en el centro de la pantalla que el participante deberá mirar fijo mientras aprieta la barra espaciadora. Esto tiene como objetivo comprobar que el Eye Tracker continúa calibrado. Esta etapa se denomina *drift correction*. Si en esta etapa se detecta que el equipo perdió la posición del ojo, se vuelve a realizar una calibración. Esto puede ocurrir por ejemplo por movimientos corporales del participante.

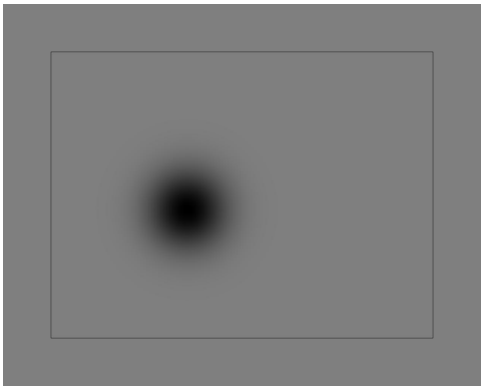
Luego se presenta un recorte de la imagen por 3 segundos en el centro de la pantalla. Este recorte es el target que deberá buscar el sujeto y es de de 72×72 píxeles (ver sección 4.1). Los recortes se presentan al doble de su tamaño original (ver figura 4.1a).

A continuación se presenta un punto de fijación negro en algún lugar de la pantalla (ver figura 4.1b). El participante debe mirarlo para poder pasar a la siguiente pantalla: este punto es un nuevo chequeo de que el sistema de seguimiento ocular esté funcionando correctamente, además de servir para uniformizar los puntos de partida de los scanpaths de todos los sujetos. Se elige un punto por imagen (el mismo para todos los sujetos)f. Dicho punto es escogido uniformemente al azar entre todos los puntos de la pantalla que se encuentran a 300 píxeles del centro del target.

(a) Muestra de target

(b) Fijación forzada

(c) Imagen completa



(d) Respuesta subjetiva

Figura 4.1: Muestra de las cuatro etapas del experimento en orden

Recién después de esta etapa el participante puede buscar el target en una imagen extensa de un interior, de 768×1024 píxeles (ver figura 4.1c). El ensayo termina cuando se encuentra el target o cuando se alcanza la cantidad máxima de sacadas, lo que ocurra primero. Como se mencionó en la sección anterior, la máquina de registro está constantemente detectando y contando sacadas. En particular, detecta el fin de la sacada (comienzo de la fijación) correspondiente y termina el ensayo, ocultando la imagen en el refresco de la pantalla siguiente a este evento.

Por último, se pide al participante que indique dónde cree que se encuentra el target, sin importar si detectamos que lo encontró o no. Para ello, el participante tiene un círculo negro de centro y radio variable (ver figura 4.1d), y se le pide que escoja el centro y el radio de forma tal que el círculo interseque con posición donde cree que se encuentra el target. El participante primero escoge el centro presionando el botón izquierdo del *mouse* sobre el punto deseado y luego determina la longitud del radio a utilizar con los botones del *mouse*. Una vez seleccionado el centro del círculo ya no se puede modificar la decisión.

Al finalizar cada ensayo hay un tiempo no acotado de descanso del que solo se sale cuando el sujeto lo decide. Se le pide al participante que no mueva la cabeza en ningún momento dentro de un bloque de ensayos, y esto incluye los descansos entre ensayo y ensayo.

El número máximo de sacadas se varió entre 2, 4, 8 o 12 sacadas. Al comienzo del experimento se decide uniformemente al azar cuántas sacadas serán permitidas en cada ensayo: $13,43\,\%$ de los ensayos permitirán 2 sacadas, $14,93\,\%$ permitirán 4 sacadas, $29,85\,\%$ permitirán 8 sacadas y $41,79\,\%$ permitirán 12 sacadas[1]. La máxima cantidad de sacadas por ensayo fue elegida al azar e independiente de la imagen, por lo que para la misma imagen hay participantes a los que se le permitió realizar distinta cantidad de fijaciones. Además, como se aleatoriza la cantidad de sacadas máxima por ensayo el participante no sabe cuántas fijaciones tendrá de antemano. Tampoco sabe que el tiempo de búsqueda depende de la cantidad de fijaciones efectuadas: esto se ocultó para que los participantes hagan búsquedas visuales lo más naturales posible.

La tarea con su distribución final de cantidad de sacadas permitidas fue tomada con 17 sujetos. Antes realizamos una puesta a punto de la dificultad de la prueba, en la que se tomó esta misma tarea pero con otras distribuciones de cantidad de sacadas permitidas por ensayo y con menos imágenes (ver tabla 4.1). Esta puesta a punto fue hecha con el objetivo de maximizar la cantidad de ensayos con varias

[1] Para las primeras pruebas se utilizó otra distribución de la cantidad de sacadas, ver tabla 4.1 para más detalle.

fijaciones donde los sujetos no hayan encontrado el target.

	Cantidad de sujetos	Cantidad de imágenes	Cantidad de sacadas						
			2	**3**	**4**	**8**	**12**	**16**	**64**
Tarea 1	3	108	20,4 %		20,4 %	20,4 %		20,4 %	18,4 %
Tarea 2	2	108	25 %		25 %	25 %		12,5 %	12,5 %
Tarea 3	6	108	25 %	25 %	25 %	25 %			
Tarea final	17	134	13,4 %		14,9 %	29,9 %	41,8 %		

Tabla 4.1: Distribuciones de cantidad de sacadas utilizadas en la puesta a punto del experimento.

Los ensayos que tengan como límite máximo de sacadas 2, 4, 8 o 12 fueron tomados en cuenta para todos los análisis mientras que los otros fueron utilizados solamente cuando fue posible: por ejemplo, en una métrica que explicaremos más adelante truncamos todos los scanpaths hasta la cuarta sacada, y para ese caso pudimos hacer uso de los datos con límite máximo de sacadas 16 y 64.

4.3. Experimento de detección de clase de objetos

Luego de la tarea de búsqueda visual vimos la necesidad de entender a qué clase de objeto pertenece cada uno de los target mostrados. El uso que se le dará a esta información será desarrollado en el capítulo 6. Intentamos utilizar software de detección automática de objetos, pero fue imposible debido a que se requieren miles de imágenes de cada clase particular de objetos para lograr hacer un detector eficiente. Los sistemas de detección de objetos del estado del arte consiguen clasificar hasta 100 categorías de objetos comunes, pero la mayoría de estas categorías no aparecen en nuestro dataset. Entonces optamos por realizar una tarea *online* en la que los usuarios describan con palabras la clase del objeto observada.

La tarea *online* puede encontrarse en `http://objetos.gpoesia.com`.

4.3.1. Descripción de la tarea

En esta tarea *online* cada target es mostrado por 3 segundos (el mismo tiempo que en el experimento original) y se le pide a los usuarios que respondan con tres frases cortas distintas a la pregunta "¿qué es el objeto de la imagen?". Además se da la opción de no responder, que debe ser utilizada únicamente si no reconoce el objeto mostrado. Las imágenes fueron mostradas en un orden aleatorio y distinto para cada sujeto. Los sujetos tuvieron la libertad de realizar la tarea en tantos días como desearan, pudiendo continuarla donde la dejaron.

4.3.2. Participantes

Obtuvimos datos de 17 sujetos, todos ellos residentes de la Ciudad de Buenos Aires y alrededores y hablantes nativos de español. De estos 17 sujetos, 8 terminaron las 134 imágenes y los 9 restantes completaron el experimento parcialmente. En promedio obtuvimos 10.5 respuestas por imagen.

Capítulo 5

Análisis del comportamiento

En este capítulo exploraremos diferentes aspectos del experimento obteniendo estadísticas sobre los datos. Separaremos los análisis según si refieren puramente a las fijaciones efectuadas o si incluyen el reporte subjetivo de dónde creen los sujetos que se encuentran los target. Denominamos los análisis de la primera categoría como *reportes objetivos* mientras los de la segunda serán *reportes subjetivos*.

5.1. Preanálisis de los datos

Como se mencionó anteriormente, cada ensayo termina cuando se detecta una fijación sobre la región en la que se encuentra el target. Para los análisis posteriores extenderemos levemente el concepto de "target encontrado": existen algunos ensayos (5.45 % del total) donde el sujeto no posó la vista sobre el target y sin embargo logra adivinarlo correctamente. En estos casos, notamos que hubo al menos una fijación que estuvo tan cerca del target que llegó a fijar parte del mismo con la visión foveal y por eso consideramos que el target en realidad fue encontrado. Esto se efectuó para los ensayos en donde hubiera habido al menos una fijación a distancia menor a 45 píxeles del centro del target, que representa aproximadamente 1.25° en el campo visual.

5.2. Análisis sobre aspectos objetivos

5.2.1. Longitud de sacadas a lo largo del tiempo

En primer lugar fue posible observar que la distribución de la longitud de las sacadas cambia según el número de fijación en el que nos encontremos. Esta reducción en la longitud promedio de las sacadas con el paso de las fijaciones ya fue observada en la bibliografía para otras tareas de búsqueda visual [Jac86, OHVE07],

y se denominan movimientos oculares *coarse-to-fine*. Este fenómeno fue estudiado especialmente en escenas artificiales aunque también hay trabajos en escenas naturales, y en nuestro trabajo este fenómeno también se observa (ver figura 5.1). Una hipótesis sobre el motivo de este comportamiento es que en la vida real la saliencia de los objetos no es conocida previamente. Así, una estrategia de movimientos oculares *coarse-to-fine* permite encontrar rápidamente objetos muy salientes, y si el target es poco saliente se habrán desperdiciado pocas fijaciones descartando esta posibilidad muy probable. Esta distribución será utilizada más adelante como un factor de uno de los modelos a analizar (ver figura 5.2).

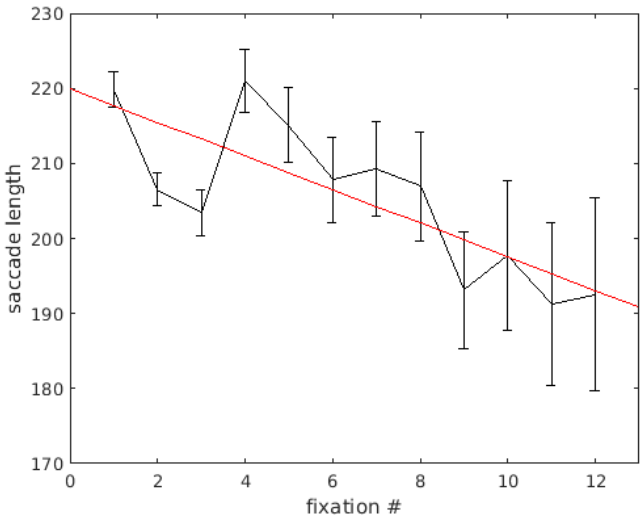


Figura 5.1: Longitud promedio de las sacadas y error estándar de la media según número de fijación.

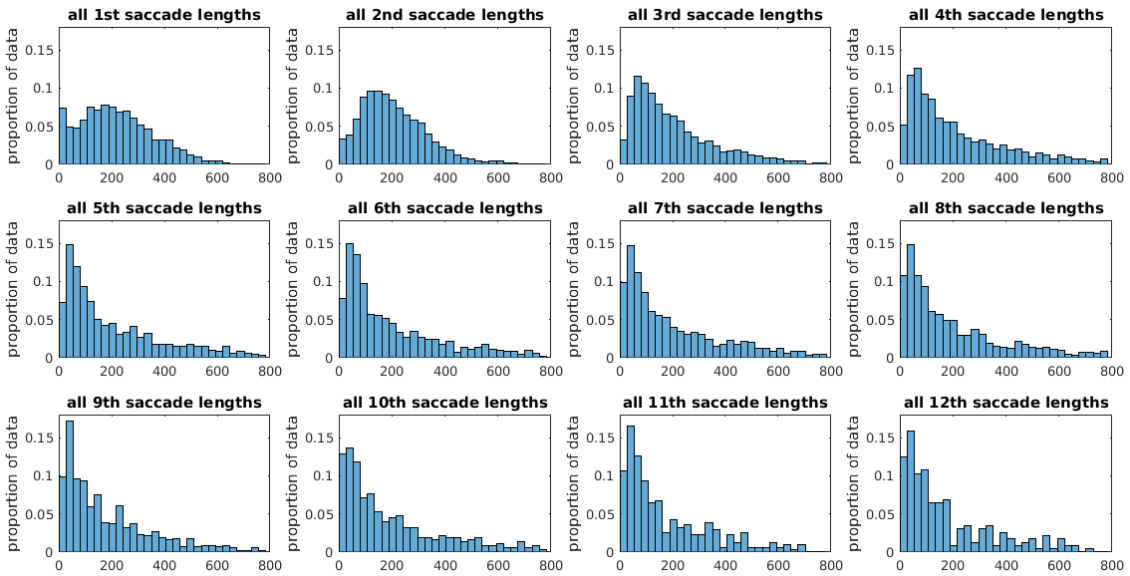


Figura 5.2: Histogramas de longitud de las sacadas según número de fijación.

5.2.2. Ángulos entre fijaciones

Si pensamos a cada scanpath como una serie de fijaciones $f_1, f_2, \dots f_n$, donde $f_i \in [0, 1024] \times [0, 768]$, entonces podemos definir los ángulos de un scanpath como $\alpha_i = f_{i-1}\widehat{f_i}f_{i+1}$ para $i = 2 \dots n-1$.

Analizando la probabilidad de que cada uno de estos ángulos sean realizados por los participantes, notamos que es aproximadamente el doble de probable que realicen ángulos de amplitud reducida o cercana a un ángulo llano que que realicen movimientos cercanos a un ángulo recto (ver figura 5.3). Esta tendencia se mantiene si se filtran los ángulos α_i tal que $\overline{f_{i-1}f_i}$ o $\overline{f_i f_{i+1}}$ son reducidos.

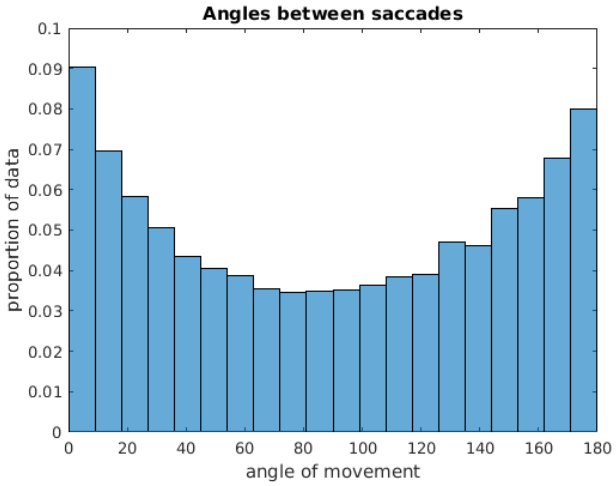


Figura 5.3: Histograma de amplitud de ángulos sin importar el número de fijación. Los α_i fueron medidos en grados.

Este resultado puede interpretarse como que los participantes son reticentes a cambiar ortogonalmente la trayectoria de su búsqueda, y en cambio hacen giros suaves - aunque a veces también vuelven sobre su trayectoria, si así lo consideran necesario.

El mismo comportamiento parece mantenerse sin importar el número de ángulo que estemos considerando, salvo en el primer ángulo: esto se puede deber a un sesgo por la fijación pedida inicialmente, o bien a que es raro realizar ángulos reducidos al comienzo de la exploración (ver figura 5.4).

5.2.3. Ángulos absolutos de las sacadas

De forma similar a la sección anterior, podemos definir los ángulos absolutos de un scanpath como el ángulo entre cada vector $\overline{f_i f_{i+1}}$ y el vector $(0, 1)$, que representa la vertical.

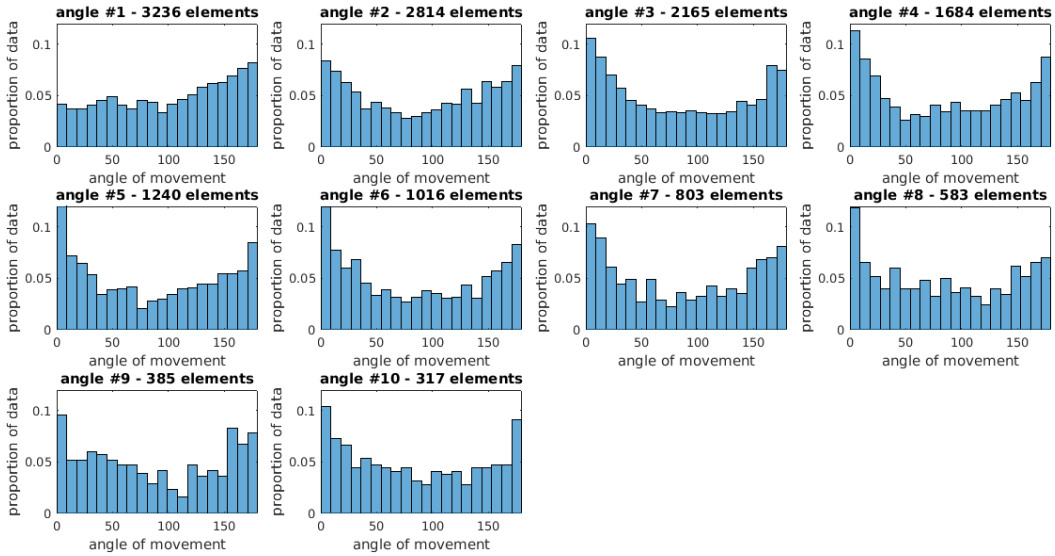


Figura 5.4: Histograma de amplitud de ángulos separando por número de ángulo. Los α_i fueron medidos en grados.

De esta forma podremos analizar si son más probables los movimientos verticales u horizontales en un scanpath. Hipotetizamos que el escaneo de una imagen con movimientos predominantemente horizontales es el más usual.

En la figura 5.5 vemos que los ángulos absolutos cercanos a los 90° son más del triple de probables que los ángulos cercanos a 0° o 180°. En otras palabras, los ángulos horizontales son mucho más probables que los verticales, sugiriendo que los humanos tienden a hacer escaneos predominantemente horizontales como hipotetizábamos. Esto puede estar relacionado con un sesgo propio de los humanos o con que las imágenes tienen dispuestos los lugares de interés de forma horizontal (por ejemplo, varios objetos a lo largo de una mesa).

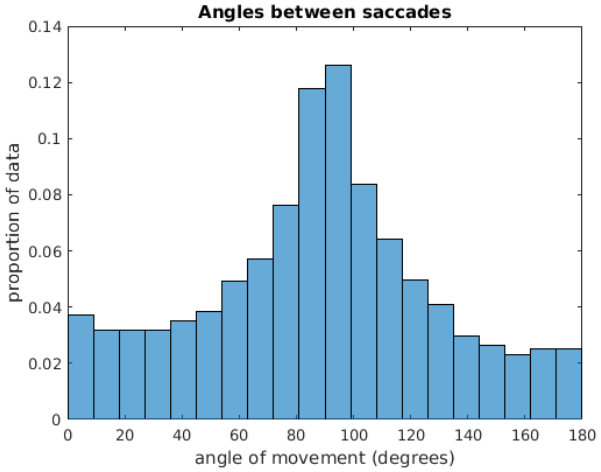


Figura 5.5: Histograma de amplitud de ángulos absolutos sin importar el número de fijación.

Este comportamiento se mantiene si observamos cada sacada por separado, aunque al igual que en el análisis anterior, no se observa en la primera sacada (ver figura 5.6).

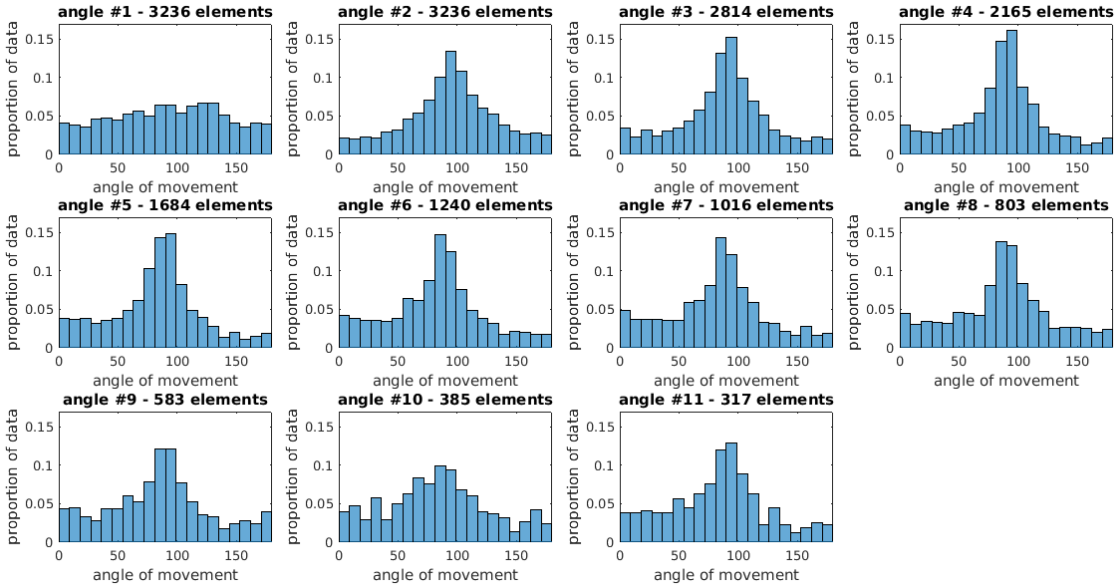


Figura 5.6: Histograma de amplitud de ángulos absolutos separando por número de ángulo. Los ángulos fueron medidos en grados.

5.2.4. Sesgos en las fijaciones de los sujetos

Nuestro dataset indica un sesgo de las fijaciones humanas hacia el centro de las imágenes, extendiéndose sobre la franja central horizontal de las imágenes, aunque menos marcadamente (ver figuras 5.7 y 5.8). Esto resulta consistente con lo observado en la literatura para otros *datasets* [TBG05, Tat07, ZTM+08]. Este sesgo es parcialmente atribuible al *setup* de nuestro experimento, en donde los sujetos están centrados con respecto al centro del pantalla y el *drift correction* aparece en el centro de la pantalla, pero también es producto de que la mayoría de las fotografías suelen tener objetos de mayor saliencia en el centro que en los bordes de las imágenes. Decimos que es solo parcialmente atribuible al *setup* del experimento pues en trabajos donde estos factores son controlados, el sesgo de la fijación central se mantiene [Tat07]. Se mostró que este sesgo de los observadores humanos tiene muchos factores que lo conforman, entre ellos: el centro de la escena puede optimizar el procesamiento de información de bajo nivel de la misma; el centro de la escena puede ser una ubicación conveniente para comenzar la exploración; puede reflejar una tendencia del ojo a re-centrarse en su órbita [Tat07].

En nuestro *dataset* el sesgo de fijación central se observa fuertemente en las primeras fijaciones, mientras que el sesgo de fijaciones en la franja central horizontal

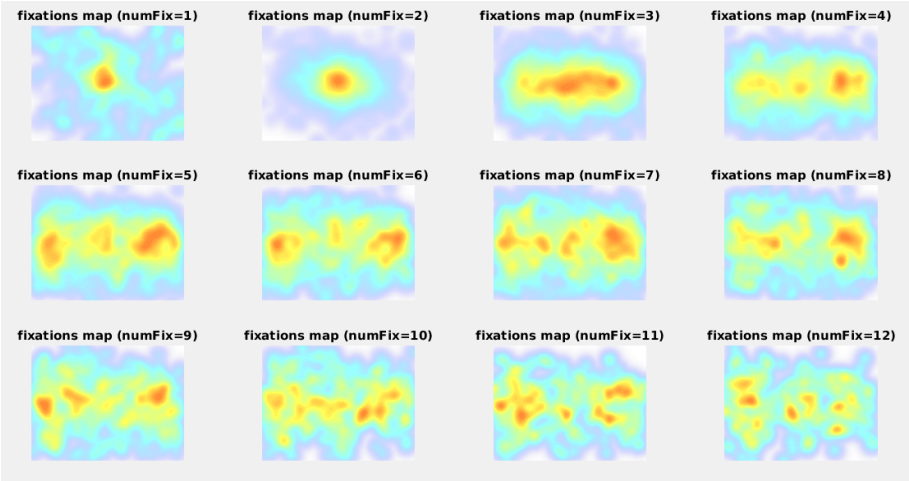


Figura 5.7: *Heatmap* de posición de fijaciones separado por número de fijación.

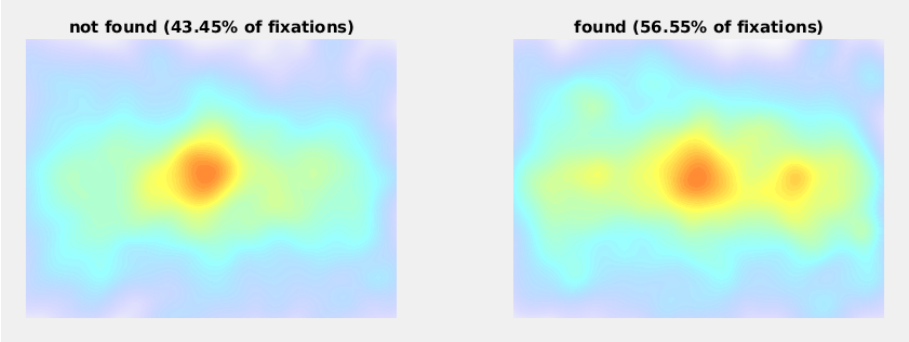


Figura 5.8: *Heatmap* de posición de fijaciones separado según si el target fue encontrado o no.

se mantiene a lo largo de todas las fijaciones (ver figura 5.7). El sesgo de la fijación central se observa muy particularmente en la segunda fijación, que es la primera fijación no forzada. Esto podría tener que ver con los factores de sesgo humano observados en [Tat07], que llevan a los sujetos a querer comenzar su búsqueda visual desde el centro de la imagen. Este sesgo se mantiene a través de los participantes, aunque en algunos es más marcado que en otros.

El sesgo de fijación central se mantiene sin importar cuál haya sido el resultado de la búsqueda visual, lo que indica que no es un factor en el éxito o fracaso de la búsqueda del target (ver figura 5.8).

Además, se observa un fuerte sesgo de fijación central en la primera fijación, que es la fijación forzada. Esto se debe a que algunos participantes mueven rápidamente la mirada al centro luego de fijar en el punto forzado y la grabación de la primera fijación se produce cuando ya están mirando al centro. Se observa que la distancia del punto de la primera fijación al centro de la imagen influye en cuán bien los

participantes respetan la instrucción de mantener la mirada en el punto pedido (ver figura 5.9 y 5.10).

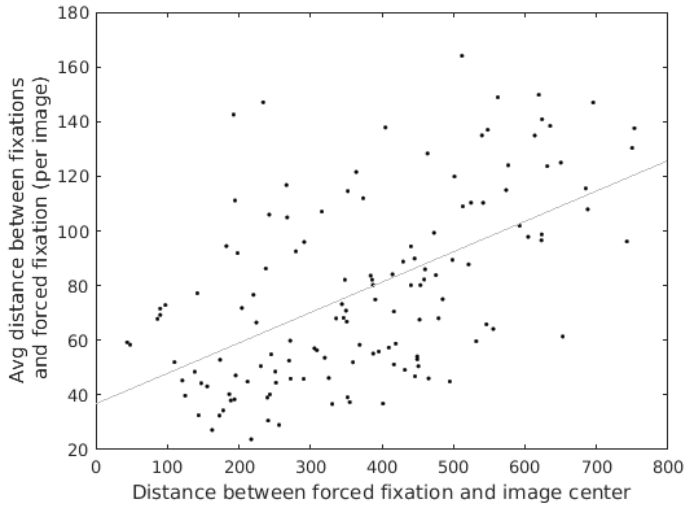
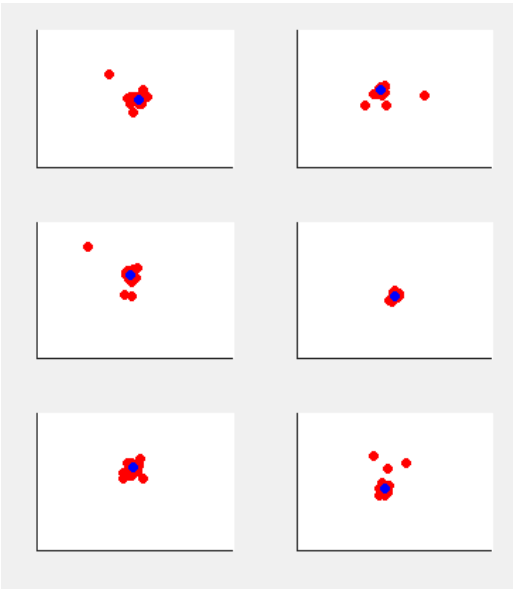
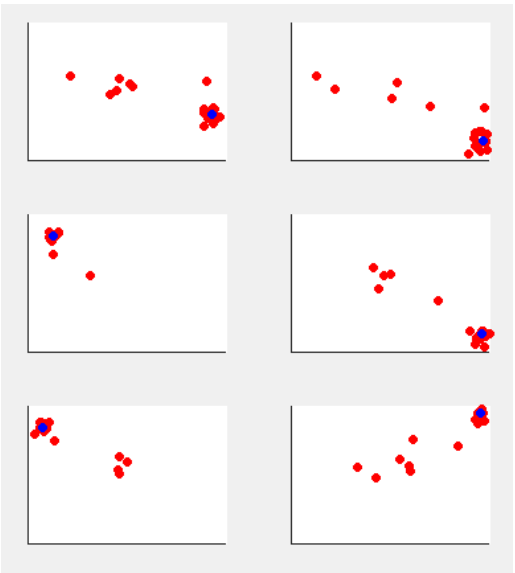


Figura 5.9: Distancia entre la fijación pedida y el centro de la imagen respecto de la distancia media de las fijaciones a la fijación pedida. Se observa $r = 0,5605$ con $p < 10^{-11}$



(a) distancia <100 entre fijación pedida y centro



(b) distancia >400 entre fijación pedida y centro

Figura 5.10: Ejemplos de primeras fijaciones en imágenes. En azul se ve la posición de la fijación pedida, en rojo se ven las fijaciones de los participantes. Se observa que en imágenes cuya fijación pedida se encuentra más cerca del centro, los usuarios respetaron más la instrucción (es decir, mantuvieron la fijación por más tiempo).

5.3. Análisis sobre reportes subjetivos

5.3.1. Cambios en la incerteza según cantidad de fijaciones

Observamos una notable diferencia en la longitud del radio del círculo de respuesta entre los ensayos donde el target fue encontrado y los que no (ver figura 5.11). Esto es un fuerte indicio de que los usuarios están utilizando este indicador para medir el nivel de confianza en su respuesta, y que como era de esperar, el nivel de confianza es mucho menor cuando el target no fue encontrado.

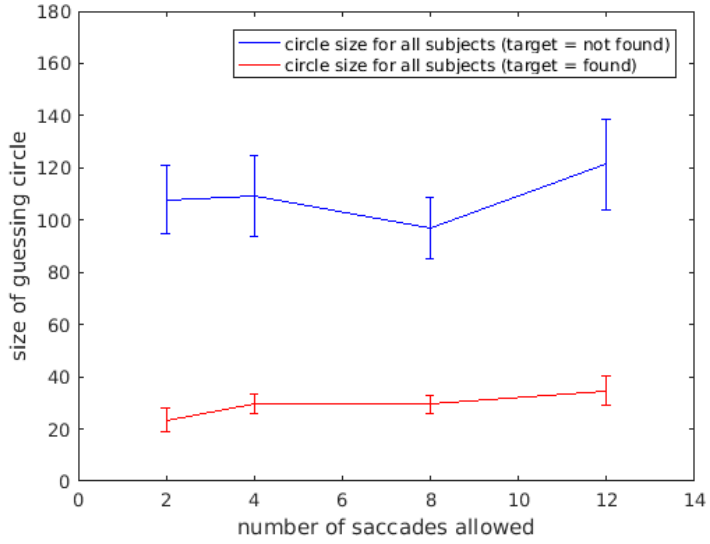


Figura 5.11: Radio promedio y error estándar de la media del círculo de respuesta.

No hay cambios notables en la longitud media del radio de respuesta según el número de sacada permitido. Es importante notar que hay más consistencia en la longitud del radio de respuesta en los ensayos en los que el target fue encontrado que en los que no fue encontrado: cuando el target no fue encontrado existen diferentes niveles de confianza (estos además varían de persona a persona), mientras que cuando el target fue encontrado el radio siempre es reducido.

Esto último indica que además de posar la vista sobre el target el sujeto llegó a reconocerlo. Recordemos que el ensayo termina cuando el sujeto realiza una fijación en la porción de la imagen donde se encuentra el target, no dejando demasiado tiempo para procesar la información de la última fijación ($43{,}2ms \pm 0{,}5ms$). El hecho de que aún así los sujetos hayan podido reconocer el objeto podría ser un indicio de que al momento de decidir dónde realizar la última fijación ya tienen cierta seguridad sobre que allí encontrarán el objeto que estaban buscando. Esto es consistente con otros estudios previos que también sugieren que el proceso de detección del target ocurre en paralelo con el de decisión de la próxima fijación. Más aún, Kotowicz et al. [KRK10] han mostrado en otra tarea de búsqueda visual con reporte subjetivo

que la precisión para detectar el target es alta sin importar la cantidad de tiempo que se dé al sujeto para procesar la información de la última fijación, incluso para tiempos menores a 10ms luego de alcanzar el target con la vista.

5.3.2. Correlaciones entre tiempos de respuesta e incerteza

Hipotetizamos que el tiempo demorado en colocar el círculo de respuesta correlaciona con el tamaño del círculo, pensando en este último como una representación del nivel de incerteza. Creemos que al tener incerteza en la respuesta se demorará más tiempo en responder tanto el centro del círculo como el tamaño del mismo.

Para poner a prueba esta hipótesis, calculamos la correlación entre el tiempo demorado para responder la ubicación del centro del círculo (lo abreviamos t_{centro}) y el tiempo demorado para responder la longitud del radio del círculo (lo abreviamos t_{radio}) para cada sujeto. Lo separamos por sujeto pues cada uno de ellos puede tener un comportamiento diferente. Para tener una noción de la distribución de esta muestra graficamos los 28 valores en un boxplot. Existe una correlación moderada entre estas dos variables: la mediana de los sujetos es de $r = 0{,}4248$ (ver figura 5.12).

Se observa una correlación fuerte entre t_{radio} y el tamaño del círculo de respuesta (ver figura 5.12, mediana de $r = 0{,}6852$). Esta correlación tan fuerte es atribuible a la mayor cantidad de *clicks* necesarios para responder círculos de mayor tamaño.

Esta correlación disminuye cuando correlacionamos t_{centro} con la longitud del radio del círculo (ver figura 5.12, mediana de $r = 0{,}3172$). Esto sugiere que el tiempo pasado para elegir el centro del círculo no es un fuerte indicador del tamaño del círculo a colocar. De todos modos, si bien el tamaño del círculo es un indicador de la incerteza éste no es perfecto: hay sujetos que hicieron menor uso del tamaño del círculo que otros que siguieron más fielmente las instrucciones del experimento. Esto creemos que se debe a que ingresar círculos mayores requiere unas fracciones de segundo más que ingresar círculos menores, y algunos sujetos buscaron terminar la tarea más rápidamente.

5.3.3. Influencia de los ensayos anteriores en la respuesta actual

Buscamos analizar cómo afecta no encontrar el target en un ensayo en las respuestas posteriores. Esto fue un factor que tuvimos en cuenta a la hora de diseñar el experimento: evitamos incluir *feedback* de la *performance* de los sujetos durante la

tarea pues consideramos que esto podría influir en el nivel de confianza los sujetos. Este nivel de confianza se ve expresado en el tamaño del círculo que coloquen al final de cada ensayo.

Queremos ver cómo afecta haber respondido con un alto nivel de incerteza en los n ensayos anteriores en la respuesta actual. Una hipótesis es que tener varias respuestas seguidas en las que no tuvieron confianza afecta negativamente en la confianza de las personas y las lleva a responder con menos confianza.

Sea t un umbral de radio del círculo a partir del cual un círculo es considerado grande y que interpretamos como una expresión del nivel de incerteza de los sujetos. Sea p_t la probabilidad de responder con un círculo de radio mayor a t, tomando todos los sujetos y ensayos. Esto lo computamos simplemente calculando qué proporción de los ensayos cumplen esta condición. Si asumimos que las respuestas de los ensayos son independientes, la probabilidad de que el radio del círculo de un ensayo sea mayor a t dado que los radios de los círculos de los n ensayos inmediatamente anteriores fueron mayores a t es p_t. Llamamos *probabilidad esperada* a este valor. Calculamos empíricamente la proporción de ensayos totales que cumplen que $radio_i > t$ dado que $radio_{i-1} > t, radio_{i-2} > t, \ldots, radio_{i-n+1} > t$ para algún i (lo llamaremos *probabilidad efectiva*), y graficamos la diferencia relativa para diferentes valores de n y t. Si la hipótesis de que los ensayos son independientes fuera cierta, la diferencia debería entre la probabilidad efectiva y la esperada debería ser nula.

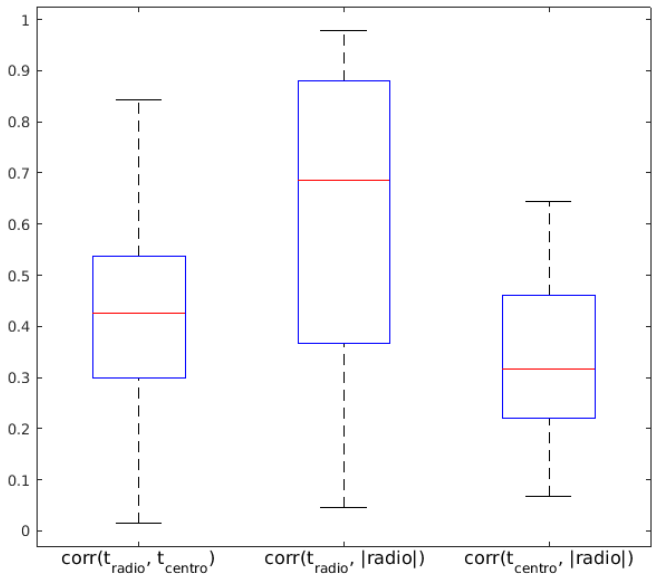
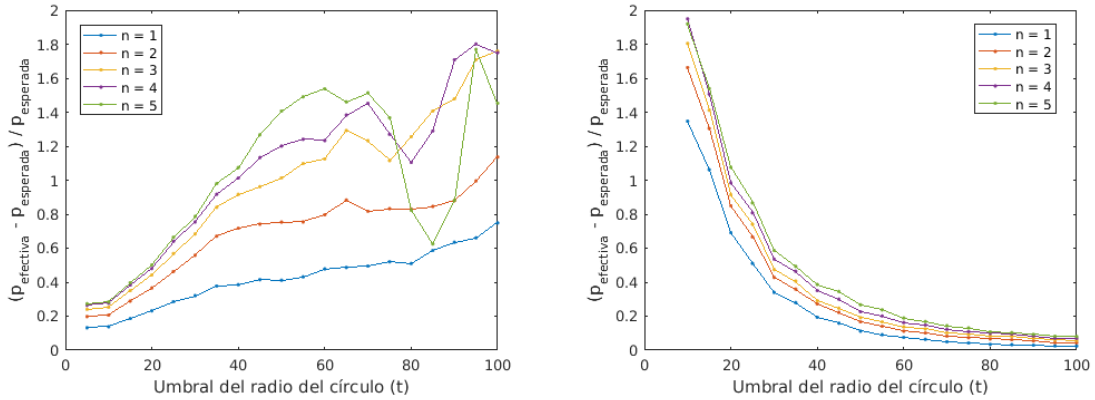


Figura 5.12: Correlación dos a dos entre tres variables: t_{radio}, t_{centro} y la longitud del radio (denominada $|radio|$).



(a) Diferencia relativa entre la probabilidad efectiva y esperada cuando los n ensayos anteriores tienen un radio de respuesta mayor a t.

(b) Diferencia relativa entre la probabilidad efectiva y esperada cuando los n ensayos anteriores tienen un radio de respuesta menor a t.

Figura 5.13: Se observa cómo el nivel de confianza observado en los ensayos anteriores correlaciona con el nivel de confianza del ensayo inmediatamente siguiente. Ver texto para explicación del método utilizado.

Como se puede ver en la figura 5.13a, la probabilidad efectiva es siempre superior a la esperada. Este efecto es más fuerte cuanto más ensayos anteriores se tomen, es decir que cuantos más ensayos consecutivos con un nivel de confianza bajo tenga un sujeto, mayor es la probabilidad de que coloque un círculo de radio mayor en el próximo ensayo. Para t muy elevados las curvas pierden la predictibilidad que vemos para valores de t más pequeños, y creemos que esto se debe a que no hay muchos ejemplos de $n = 4$ y $n = 5$ círculos consecutivos mayores a t. A modo de ejemplo, menos del 8 % de los ensayos tienen un radio de respuesta mayor a 100 píxeles.

Este mismo efecto se observa, aunque en mucha menor medida, cuando se analiza el caso opuesto: ¿tener varios ensayos consecutivos de alta confianza impacta positivamente en la confianza del sujeto? En este caso p'_t se define como la probabilidad de responder con un círculo de radio menor a t, tomando todos los sujetos y ensayos. Los resultados obtenidos se pueden ver en la figura 5.13b.

5.3.4. Aprendizaje a lo largo de las fijaciones

Una hipótesis que manejamos fue que los sujetos aprenderían a realizar más eficientemente la tarea con el paso de los ensayos. Incluso varios sujetos mencionaron al final del experimento que sentían haber mejorado con la práctica. Para poner a prueba esto se asignó un puntaje representando la dificultad de cada ensayo y se sumaron los puntajes de los ensayos en los que fue encontrado el target preservando el orden original. Este puntaje es el complemento del porcentaje de ensayos exitosos

para esa cantidad de sacadas máximas permitidas. Por ejemplo, como el 15,83 % de los ensayos con 2 sacadas permitidas son exitosos, encontrar el target en un ensayo de dos sacadas permitidas da $100 - 15{,}83 = 84{,}17$ puntos. Los puntajes asignados se pueden ver en la tabla 5.1.

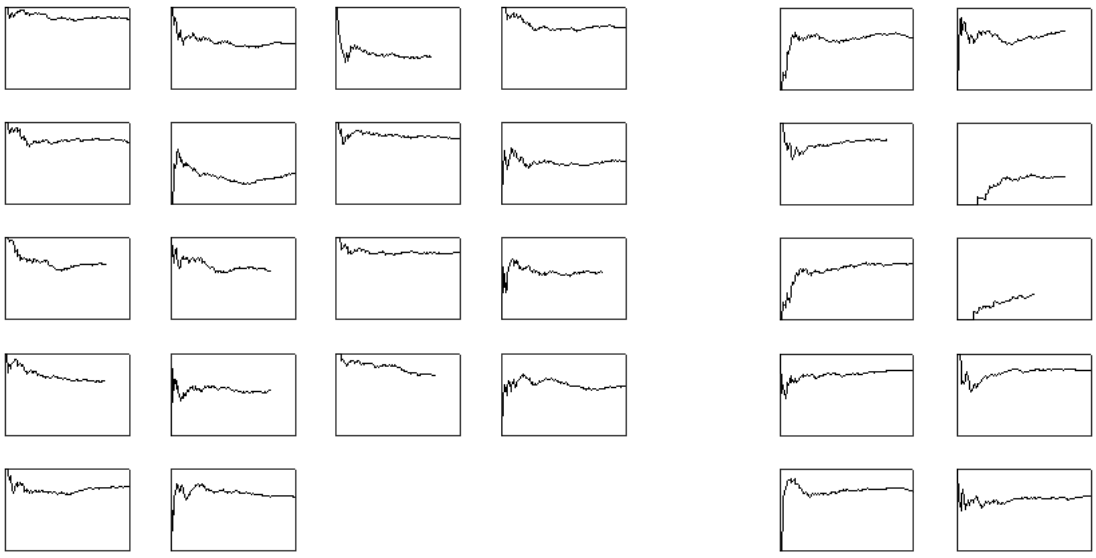
	cantidad de sacadas permitidas						
	2	*3*	*4*	*8*	*12*	*16*	*64*
puntaje	15.83	27.83	39.85	69.47	77.21	84.68	92.46

Tabla 5.1: Puntaje asignado a cada ensayo según la cantidad de sacadas permitidas.

Como los primeros sujetos realizaron el experimento con otra distribución de sacadas máximas permitidas que los últimos, uniformizamos el análisis dividiendo el puntaje acumulado obtenido en cada ensayo por el puntaje máximo que podría haber alcanzado hasta ese ensayo.

Luego de estabilizarse en los primeros ensayos, podemos ver que la evolución de los puntajes tiene pendiente casi nula para la mayoría de los sujetos (ver figura 5.14). Separamos los sujetos según si la correlación entre el puntaje y el número de ensayo dio positivo o negativo para graficar. Además, realizamos más experimentos separando los ensayos en mitades y en tercios, calculando separadamente los puntajes de cada porción. No se observó una diferencia significativa en los puntajes de los sujetos.

Es importante aclarar que en nuestro experimento las imágenes fueron ordenadas uniformemente al azar, así como la cantidad de fijaciones permitidas: se ordenó al azar un vector con la lista de fijaciones máximas permitidas para ese sujeto.



(a) (b)

Figura 5.14: En (a) se observan los sujetos cuya correlación entre el número de ensayo y el puntaje obtenido fue negativo. En (b) se observan los que obtuvieron correlación positiva. El puntaje siempre está en el rango $[0, 1]$ ya que fue normalizado dividiendo por el puntaje máximo posible a alcanzar.

Capítulo 6

Modelos estáticos

En este capítulo y en el siguiente analizaremos diferentes modelos para predecir la ubicación de las fijaciones humanas en distintas imágenes de interiores. En este capítulo trataremos modelos que no utilizan información de las fijaciones anteriores para decidir la próxima fijación a realizar: solo utilizan información que se puede deducir de la imagen original y del target. Denominamos *modelos estáticos* a esta clase de algoritmos. Como consecuencia de la definición anterior es que decimos que los modelos estáticos buscan predecir las regiones donde es más probable que se sitúen las fijaciones de los sujetos, ya que no tienen una noción de orden entre las fijaciones.

6.1. Introducción y medidas de performance

En este capítulo trataremos modelos que solo intentan predecir las áreas que serán más observadas por los participantes. Esto está muy relacionado con el concepto de saliencia en una imagen, ya que una ubicación es saliente si llama la atención visual humana. Si bien en una tarea de búsqueda visual los patrones de fijaciones son distintos que en otros tipos de tareas, esto no significa que se dejen de observar completamente las ubicaciones salientes (ver sección 2.3). Por todo lo anterior es que compararemos varios modelos de saliencia para encontrar el que mejor prediga nuestro *dataset*. Los modelos de saliencia suelen ser diseñados para casos en el que el participante solo observa las imágenes sin ninguna tarea en particular. Como nuestra tarea es de búsqueda visual, intentaremos aprovechar al máximo la información que el target nos provee para poder superar los modelos del estado del arte.

Focalizaremos especialmente en mapas que predigan la tercera fijación porque ese es el mapa que vamos a tomar como *prior* para los modelos dinámicos. Decidimos

tomar la tercera fijación y no las anteriores pues las primeras dos fijaciones están sesgadas por el sesgo de la fijación central (además, la primera lo está por la fijación forzada). También haremos mención al poder de predicción de estos modelos en fijaciones posteriores a la tercera fijación.

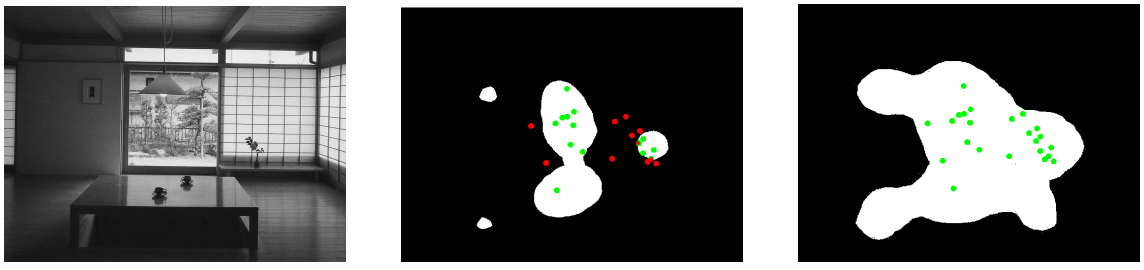
Para comparar la performance de los distintos mapas de saliencia, los mismos son tratados como un clasificador binario en cada pixel de la imagen. Un porcentaje dado de los píxeles de una imagen se clasifican como positivos y el resto se clasifican como negativos. La sección clasificada como positiva será la porción donde el modelo predice que caerán todas las fijaciones de los participantes, es decir, la porción más saliente. Las fijaciones humanas se grafican en un *heatmap* y son tratadas como otro mapa de saliencia que refleja la performance máxima alcanzable.

Posteriormente se grafica para cada método la fracción de verdaderos positivos (TPR, del inglés *True Positive Rate*) promedio obtenido en todas las imágenes y el desvío estandar del promedio para varios umbrales de porcentaje de saliencia. Estos umbrales son denominados *Percent salient* en los gráficos.

$$True\ Positive\ Rate\ (TPR) = \frac{True\ Positives\ (TP)}{True\ Positives\ (TP) + False\ Negatives\ (FN)}$$

En la figura 6.1 se puede ver un ejemplo de dos umbrales de saliencia distintos y las fijaciones que son consideradas *true positives* y *false negatives* para cada umbral.

En otras palabras, al calcular el TPR se observa qué porción de las fijaciones caen en la región saliente.



(a) Imagen original

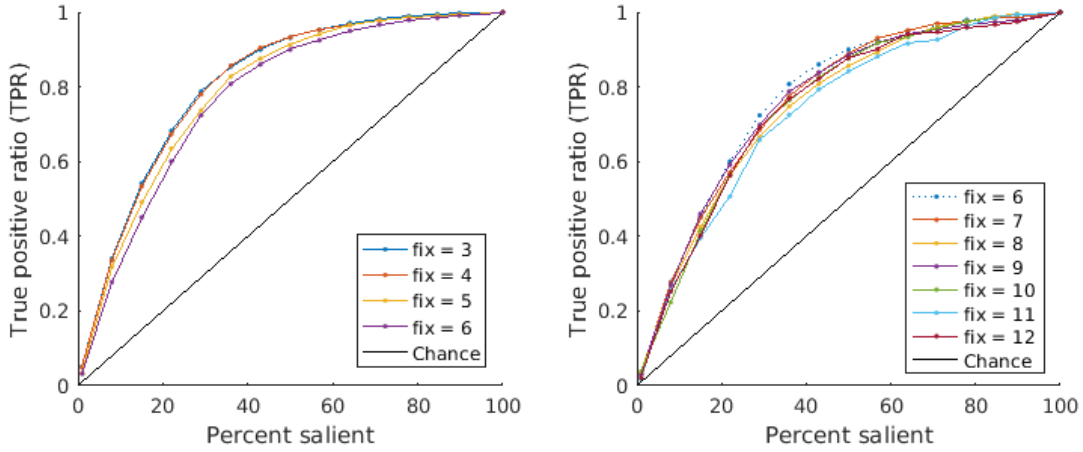
(b) 8.14 % más saliente de la imagen según SAM

(c) 27.73 % más saliente de la imagen según SAM

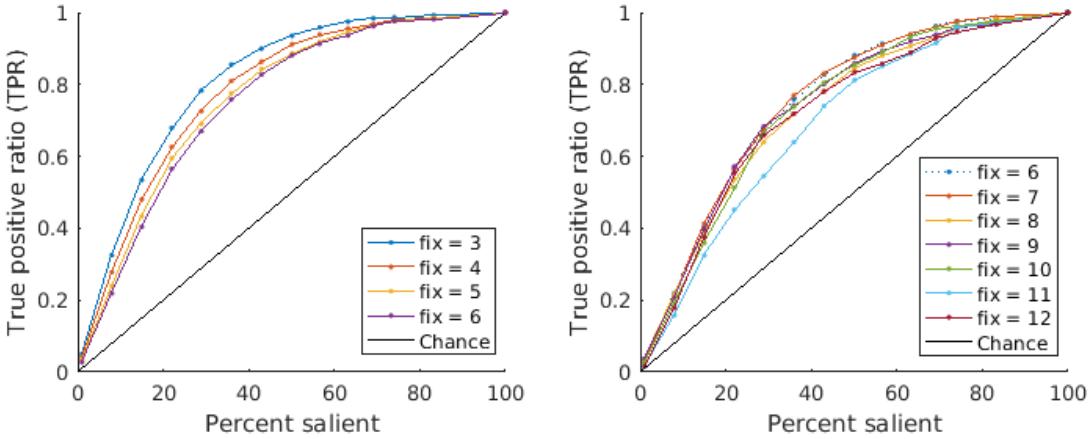
Figura 6.1: Ejemplo de umbrales de saliencia. En verde y rojo se ven las terceras fijaciones de los sujetos: en verde se observan las fijaciones en posiciones salientes (*true positives*) y en rojo las fijaciones en posiciones no salientes (*false negatives*).

6.1.1. Mapas de saliencia como predictores de fijaciones

Tomamos dos modelos de saliencia del estado del arte y comparamos la performance para predecir las fijaciones de cada número de fijación. Se observa que el nivel máximo de predicción ocurre en la tercera fijación y con la progresión de las fijaciones la performance se degrada hasta estabilizarse en la séptima fijación (ver figura 6.2).



(a) Performance de MLNet



(b) Performance de SAM

Figura 6.2: TPR medio para todas las fijaciones y porcentaje de la imagen considerada saliente. Se puede ver en ambos casos que la mejor predicción se alcanza en la tercera fijación y la performance se va degradando con el paso de las fijaciones. Luego de la séptima fijación la performance se estabiliza. Se separaron los gráficos en dos para una mejor observación de las curvas: notar que la fijación 6 se repite en ambos gráficos para una mejor comparación.

6.1.2. Features del modelo de Judd original original

Además de los dos modelos recién mencionados tomamos el modelo de Judd et al. explicado en la sección 2.2.1. Este modelo está compuesto por varios modelos más simples como predictores de saliencia. Cada uno de estos modelos individualmente no tiene una precisión alta pero como ayudan a predecir distintos aspectos de la toma de decisión podremos combinarlos para intentar obtener mejores resultados. Los modelos más importantes que utilizó Judd como componentes de su trabajo se explican a continuación. Luego detallaremos cómo expandiremos el modelo de Judd para alcanzar resultados mejores que los del trabajo original en nuestro *dataset*.

- **Distancia al centro:** en este mapa, cada píxel representa la distancia del mismo al centro de la imagen. Este mapa busca modelar el sesgo de la fijación central (ver figura 5.2.4).
- **Horizonte:** mediante un software de detección automática del horizonte se calcula dónde se encuentra la línea del horizonte en cada imagen [TS01, Hoi07, SKT+08]. Luego se aplica un filtro gaussiano sobre esto, difuminando el horizonte. Así, los puntos más probables a ser fijados son los que están sobre la línea del horizonte y la probabilidad desciende proporcionalmente a la distancia a éste. Este mapa se creó bajo el supuesto de que numerosos objetos se encuentran sobre la línea del horizonte y por lo tanto es un lugar donde los humanos naturalmente buscan objetos salientes.
- **Subbandas:** se utilizan varias subbandas de *steerable pyramids* [SF95] como features porque se conocen que están relacionadas con la atención visual y que son fisiológicamente plausibles [JEDT09]. En este caso se utilizan las subbandas en 4 orientaciones y 3 escalas, generando 12 features. Una *steerable pyramid* es una implementación de un banco de filtros pasa-banda que tiene como objetivo descomponer linealmente una imagen en subbandas según su orientación y escala.
- **Itti & Koch:** como la intensidad y la orientación se consideran features importantes para la saliencia *bottom-up* desde hace mucho tiempo, se incluyen dos mapas que modelan estas cualidades (uno modela la intensidad y el otro la orientación). Estos mapas son extraídos de [IK00].

Todos estos son features utilizados por Judd et al. [JEDT09], que los combina con una Support Vector Machine para obtener el modelo final de predicción de saliencia. Todos ellos junto con el mapa obtenido por MLNet y SAM conformarán los features que denominaremos *el modelo de Judd original*. Estos features son los

del modelo presentado en 2009 salvo porque se removieron los mapas de canales de colores y los mapas de detección de autos, rostros y personas pues estos no tenían sentido para nuestra tarea. Además, utilizamos MLNet y SAM en vez de otros modelos de saliencia pues los modelos originales ya no eran del estado del arte (en el caso de Rosenholtz [Ros99]) o no fue posible conseguir el código (en el caso de Cerf et al. [CHEK08]). En este último caso el paper de Judd et al. no da ningún motivo en particular para usar este modelo y no otro, y es por eso que decidimos utilizar otro modelo de saliencia en vez de implementar el de Cerf et al. de cero.

Como la distribución de las fijaciones parece tener características diferentes según el número de fijación (ver figura 6.2b), se entrenará el modelo de Judd separando el modelo en tres grupos: la tercera fijación, las fijaciones 4 a 6 y las fijaciones 7 a 12. Además, se descartan la primera y segunda fijación pues la primera refleja la fijación forzada y la segunda fijación se encuentra fuertemente sesgada por el sesgo de la fijación central (ver figura 6.3).

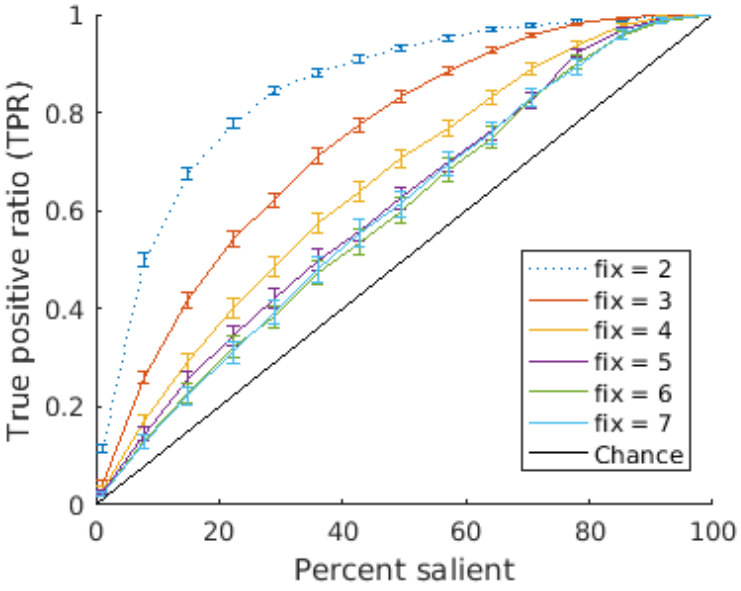


Figura 6.3: TPR medio vs. porcentaje de la imagen clasificada como saliente para el mapa de saliencia que predice mayor nivel de saliencia cuanto menor es la distancia al centro de la imagen. La barra de error indica el desvío estándar de la media. Se observa un fuerte poder predictivo en la segunda fijación, producto del sesgo de la fijación central. Se omiten las fijaciones posteriores a la séptima pues la performance es muy similar a la de ésta.

6.1.3. Expansión de features del modelo de Judd

Como mencionamos en la sección 2.2.1, es fácil utilizar este modelo entrenándolo con otro conjunto de features. Dado que en nuestra tarea la exploración está guiada por la búsqueda visual de un target cuya imagen se conoce, podemos

construir mapas de features utilizando esta información. Estos mapas se clasificarán en dos tipos:

I. Mapas teniendo en cuenta el aspecto netamente visual del target (contraste de color, orientación del objeto)

II. Mapas teniendo en cuenta la semántica de la imagen mostrada. Esto incluye el reconocimiento de la clase de objeto a la que pertenece el target y la unión de este conocimiento con el de las experiencias previas del sujeto. Por ejemplo, si se muestra la imagen de una taza de café y el sujeto la reconoce, entonces seguramente buscará la taza sobre alguna superficie de apoyo y no en el techo.

   Todos los enfoques que encontramos para realizar mapas de esta categoría se focalizan en clases de objetos específicos para los que existe un detector de objetos ya entrenado (por ejemplo, tazas, personas, cuadros o autos) [KTZC09, TOCH06]. Nuestro objetivo aquí será proponer un modelo que no necesite de entrenar un detector para la clase de objeto del target para funcionar. Esto es clave ya que en nuestra tarea sería imposible entrenar un detector de objetos para cada clase de target posible pues hay casi tantas clases de objetos como imágenes en nuestro dataset, y porque además se necesitan miles de imágenes de entrenamiento para lograr un detector de una clase de objetos particular. El problema de la detección de clases de objetos aún no fue resuelto y es uno de los problemas centrales en visión computacional.

Se crearán mapas de features por cada una de estas dos categorías y estas se sumarán a las features del modelo de Judd original. Explicaremos ambas categorías a continuación.

6.1.3.1. Features de aspectos visuales del target

Para los mapas de features de tipo I se construirán dos mapas: uno será el mapa de la cross-correlation entre la imagen original y el target (utilizamos la implementación de MATLAB, `corr`); el otro será una medida de similitud de bajo nivel entre la imagen original y el target.

Esta medida de similitud de bajo nivel coloca una grilla con casillas de 6×6 píxeles tanto en el target como en la imagen original y promedia los valores de los píxeles dentro de cada casilla (ver ejemplo en la figura 6.4). Llamamos *imagen en bajo nivel* a la imagen original una vez que se promediaron los valores de cada casilla de 6×6 píxeles.

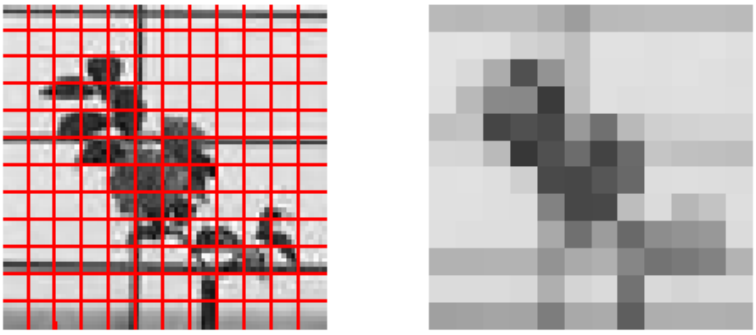


Figura 6.4: A la izquierda se ve la grilla sobre el target, en rojo. Cada casilla es de 6×6 píxeles. A la derecha se ve el promedio de los valores de cada píxel dentro de una misma casilla, amplificado. Como el target original es de 72×72 píxeles, la reducción será de 12×12.

Luego se calcula la distancia euclidiana entre cada grupo de 12×12 casillas en la imagen en bajo nivel con el target en bajo nivel, pensando a cada bloque de 12×12 como un vector de 144 dimensiones. Como resultado se tendrá una matriz de 128×170 con las distancias entre cada bloque de 12×12: a menor distancia, mayor similitud de bajo nivel del bloque con la imagen original. Finalmente se reescala la imagen a 768×1024, que son las dimensiones originales de la imagen y las dimensiones de todas las features.

6.1.3.2. Features de aspectos semánticos del target

Para los mapas de features de tipo II se detectará a qué clase de objeto pertenece cada target con una palabra en español. Se medirá la distancia de la palabra que representa a cada target respecto de algunas palabras que denotan posición en una escena de interiores (por ejemplo, mesa, piso, techo, ventana, pared - las llamaremos *palabras posicionales*). Luego se tomará la palabra posicional más cercana y ésta representará la posición probable del target: esto hipotetizamos que será un indicio de los sectores de la imagen asociados a la palabra posicional. El mapa de features será entonces el mapa de las regiones más fuertemente asociadas a la palabra posicional. Veamos en detalle cómo realizar cada uno de estos puntos.

Para detectar a qué clase de objeto pertenece cada target tomamos los datos del experimento explicado en la sección 4.3. Para cada target se seleccionó la palabra más mencionada, siempre y cuando hubiera tenido por lo menos 5 menciones. Las imágenes que no alcanzaron este límite o cuya opción más elegida fue la de no reconocer el objecto mostrado se denominan *inconcluyentes* y se omiten para todo posterior análisis.

A continuación debemos generar un mapa que muestre las regiones más

fuertemente asociadas a cada palabra posicional. Para ello tomamos una base de datos de más de 80000 imágenes segmentadas y anotadas creado por el MIT-CSAIL: LabelMe. LabelMe [RTMF08] es una herramienta web abierta que permite segmentar imágenes formando polígonos que bordeen distintos objetos y anotando palabras que los describan. Este corpus tiene más de 80000 imágenes anotadas por la comunidad. Por su característica de ser abierto, algunas de esas imágenes están anotadas parcialmente o tienen algunas segmentaciones erróneas, pero sin embargo una de las bases de datos más utilizadas para visión computacional. La figura 6.5 muestra un ejemplo de segmentación de imágenes. Todas las anotaciones de la base de datos están en inglés.

Figura 6.5: Ejemplo de imagen segmentada en LabelMe.

Para entender las regiones más asociadas a cada palabra posicional uniformizamos todas las imágenes de la base de datos para que fueran de 192×256 píxeles y graficamos todos los segmentos anotados con esa palabra posicional sobre una misma imagen. Sumamos la cantidad de segmentos que se encuentran sobre cada uno de los 192×256 píxeles: las regiones con más probables serán aquellas donde caigan más segmentos. Luego reescalamos los mapas para que todos tengan la misma suma. En la figura 6.6 se pueden ver los mapas de las 16 palabras posicionales que utilizamos en este trabajo.

Finalmente, resta saber qué mapa de regiones asignarle a cada imagen. Como ya mencionamos, será el mapa de la palabra posicional "más cercana" a la palabra que representa el target de la imagen. Representamos a cada palabra como un vector de 300 dimensiones y calculamos la distancia entre cada palabra y cada posible palabra posicional utilizando el coseno del ángulo comprendido entre los dos vectores (similitud coseno). Los vectores fueron extraídos utilizando un modelo pre-

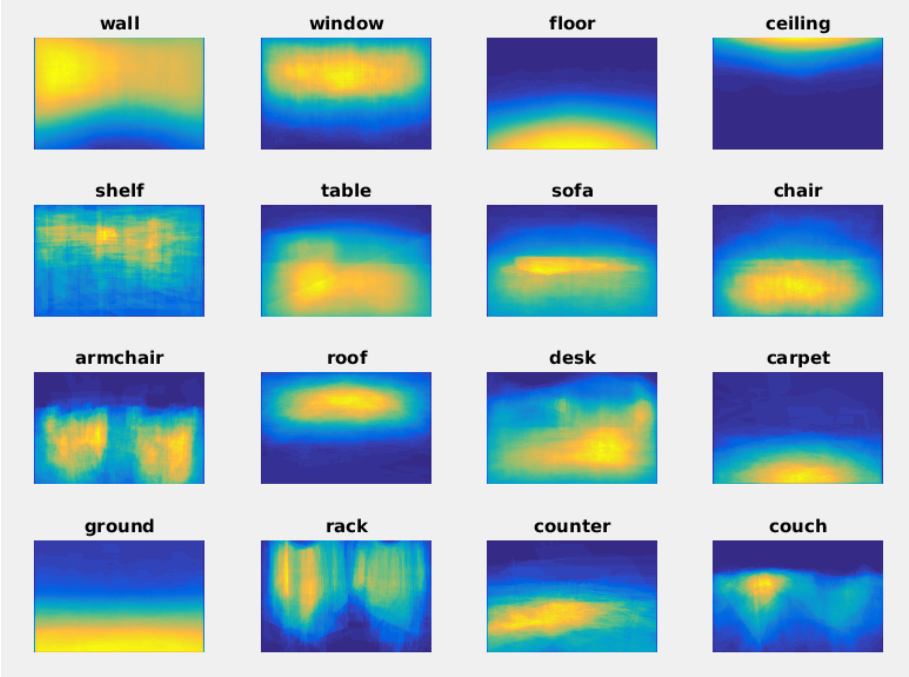


Figura 6.6: Mapa de regiones asociadas a cada palabra posicional.

entrenado de FastText [BGJM16] sobre la Wikipedia en español. Además, tradujimos las palabras posicionales al español para que estén en el mismo idioma que las palabras que describen a los target.

Los mapas de las palabras inconcluyentes fueron definidos como un mapa del horizonte, asumiendo que el horizonte se encuentra en el centro de la imagen.

Para intentar compensar posibles cambios en la posición de la cámara al tomar la fotografía consideramos usar un software de detección de la línea del horizonte [TS01, Hoi07, SKT+08] y trasladamos los polígonos para que la línea del horizonte siempre esté centrada horizontalmente. Luego detectamos la línea del horizonte para cada una de las imágenes de nuestro dataset y trasladamos el mapa obtenido para que coincida con la altura de la línea del horizonte de cada imagen. Se completaron las posibles filas faltantes en el mapa (resultado de la traslación) con varias copias de la primera o última fila del mapa, dependiendo de la dirección de la traslación.

Realizar esta corrección tuvo un leve impacto en los mapas de features resultantes pero influyó negativamente en la performance del modelo final. Creemos que esto está relacionado con que el modelo de detección de línea del horizonte fue entrenado en imágenes de exteriores y se conoce que las dinámicas de los dos tipos de imágenes son diferentes [QT09]. Analizamos visualmente los horizontes detectados para las 134 imágenes de nuestro dataset y observamos errores en varias imágenes. Si bien no logramos buenos resultados incluyendo la detección del horizonte, creemos que entrenando el modelo con imágenes de interiores esto podría mejorarse. Esto no

fue realizado porque implicaría crear un set de entrenamiento de miles de imágenes de interiores y marcar el horizonte en todas ellas, pero queda como trabajo futuro.

6.1.3.3. Resumen de features utilizadas

Nuestro modelo de Judd extendido tendrá 23 features. 20 de estas son las del modelo de Judd original descrito en la sección 6.1.2, y a éstas se suman dos features que expresan el aspecto visual del target y una feature referida al aspecto semántico del target. Sobre esta última quedan posibles trabajos futuros que podrían ayudar a mejorar su performance tal como el entrenamiento de detectores de línea del horizonte en escenas de interiores, la traducción al inglés en donde existen mejores modelos preentrenados o el entrenamiento de un modelo `word2vec` sobre un corpus en el que juzguemos que las descripciones de escenas de interiores serán frecuentes (por ejemplo, un corpus compuesto de novelas). Veremos cuánto aporta cada feature agregada por nosotros al resultado final del modelo en la sección 6.2.4.

6.2. Resultados

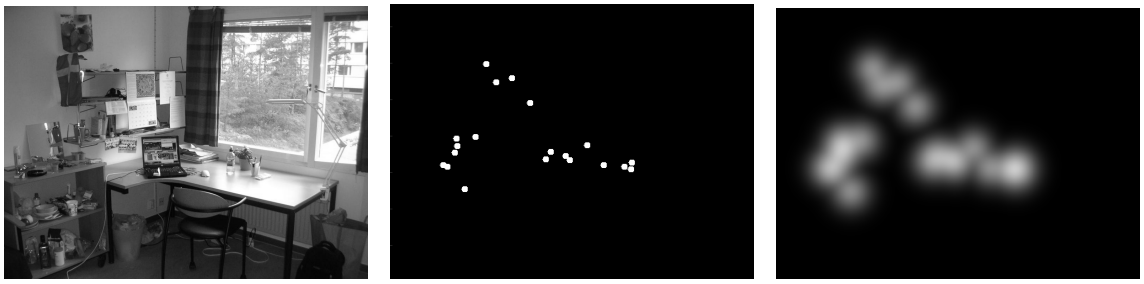
A continuación veremos los resultados obtenidos por los modelos con las features propuestas anteriormente para distintos rangos de fijaciones. Además, veremos cuán consistentes son los humanos entre sí en sus fijaciones y cuánto aportó cada feature individualmente al modelo original de Judd. Más aún, aprovecharemos los modelos anteriormente descriptos para predecir los centros de los círculos de respuesta humanos. Todos los resultados se analizan con la misma metodología, que describiremos en detalle.

6.2.1. Metodología

Se graficará el TPR de cada modelo y cada umbral de saliencia, sobre las imágenes y sujetos que se especifiquen en cada caso. Además se graficará el TPR del mapa formado por las fijaciones de los humanos, umbralizado. Este mapa es un filtro gaussiano aplicado sobre una imagen negra salvo por los píxeles en donde fijó la vista algún humano (ver ejemplo en la figura 6.7). En nuestro caso, aplicamos un filtro gaussiano $\sigma = 25$, que representan aproximadamente 0,71° del campo visual.

6.2.2. Consistencia entre humanos

En varias tareas se ha visto que las fijaciones entre humanos suelen ser consistentes [JEDT09], aunque no necesariamente las realicen en el mismo orden. Es importante notar que las imágenes de los *benchmark* de saliencia más famosos



(a) (b) (c)

Figura 6.7: Tercera fijación de todos los sujetos en la imagen `grayscale_4_other.jpg`, que puede verse en (a). En (b) se ven los puntos fijados por los sujetos y en (c) el mapa de las fijaciones de los humanos, obtenido aplicándole a la imagen de la izquierda un filtro gaussiano con $\sigma = 25$.

(MIT300, CAT2000 [BJB+]) están compuestos por numerosas imágenes de baja entropía y algunas de alta entropía. Una imagen se dice de *baja entropía* cuando la información está concentrada en pocos puntos, lo que implica que la consistencia de las fijaciones de los humanos aumenta ya que no hay demasiados lugares interesantes para observar detalladamente. Nuestro *dataset*, por el contrario, está compuesto de una gran mayoría de imágenes de alta entropía ya que esto fue necesario para lograr una tarea de búsqueda visual no trivial. Así, si bien en nuestra tarea también se observa una alta consistencia entre los humanos, el AUC dará un resultado menor que en la literatura.

Para comprobar la consistencia entre humanos creamos un mapa de saliencia por imagen conformado por las fijaciones de 24 humanos y lo utilizamos para predecir las fijaciones de los 4 humanos restantes. Creamos un mapa con todas las fijaciones de los 24 sujetos y otros mapas con la i-ésima fijación de los 24 sujetos para $i = 2, 3, 4, 8, 10, 12$. Se observa una alta consistencia entre humanos tanto en todas las fijaciones como en la segunda, tercera y cuarta, con un decrecimiento en las posteriores fijaciones (ver figura 6.8). Los mapas de las fijaciones posteriores tienen menos ejemplos pues la mayor parte de los 24 sujetos no llegan hasta las últimas fijaciones (ya sea porque encontraron el objeto o porque el experimento se cortó antes). Además, al juntar todas las fijaciones en un mismo mapa, las primeras fijaciones tienen más peso pues una gran porción de los scanpath no llegan a las últimas fijaciones. Eso explicaría por qué la predicción total es tan alta.

A continuación nos focalizaremos en predecir las fijaciones de los sujetos para imágenes que no estén en el conjunto de entrenamiento.

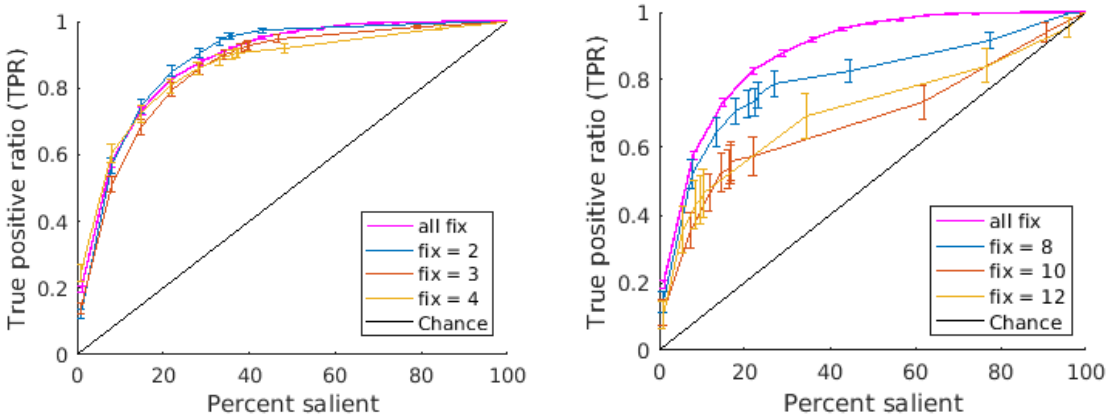


Figura 6.8: TPR medio y desvío estándar de la media de los mapas de saliencia conformados por las fijaciones de los 24 humanos para predecir las fijaciones de los 4 humanos restantes.

6.2.3. Predicciones de las regiones de las fijaciones

Como mencionamos anteriormente, dividimos las fijaciones en tres grupos: la tercera fijación, las fijaciones 4 a 6 y las fijaciones 7 a 12. La metodología del análisis se describe a continuación.

Separamos el conjunto de sujetos en dos subconjuntos al azar: uno de entrenamiento con $\frac{3}{4}$ de los sujetos y otro de prueba con el $\frac{1}{4}$ restante. Entrenamos el modelo de Judd extendido con un conjunto de entrenamiento de 100 imágenes elegido aleatoriamente, tomando únicamente los sujetos del conjunto de entrenamiento (es decir, 100 imágenes con $\frac{3}{4}$ de los sujetos). Luego evaluamos todos los modelos con las 34 imágenes restantes sobre el $\frac{1}{4}$ de los sujetos que separamos para el conjunto de prueba. Esta metodología asegura que no hay superposición ni de imágenes ni de sujetos entre el conjunto de entrenamiento y de prueba, es decir que no hay forma de que nuestro algoritmo haya aprendido de los datos de prueba. Repetimos este procedimiento para 4 conjuntos de sujetos de entrenamiento diferentes y 5 conjuntos de imágenes de entrenamiento diferentes para evitar analizar los efectos de un conjunto particular. Así, el conjunto de entrenamiento resulta estar compuesto por las fijaciones de $\frac{3}{4}$ de los sujetos en 100 imágenes, y el conjunto de prueba está compuesto por las fijaciones de $\frac{1}{4}$ de los sujetos en 34 imágenes.

Es importante notar que esta metodología reduce la cantidad de datos en el conjunto de entrenamiento y el de prueba, por lo que será de interés hacer un compromiso en pos de mostrar resultados sobre más datos. El compromiso que efectuaremos será el de experimentar sin separar los sujetos en dos subconjuntos. En este caso no se podrá graficar la performance de los humanos, pues este mapa tendría información sobre todas las fijaciones en las imágenes de prueba.

Siempre que entrenamos un modelo de Judd extendido lo hacemos con 5 modelos y la performance de cada uno de ellos describe una curva. Calculamos el área bajo cada curva (lo usamos como métrica de performance) y graficamos únicamente el modelo que corresponde a la mediana de las performances.

6.2.3.1. Modelado de la tercera fijación

Observamos que el modelo de Judd extendido logra resultados marginalmente superiores a SAM y MLNet, y según el caso alcanza performance similares a los humanos (ver figura 6.9).

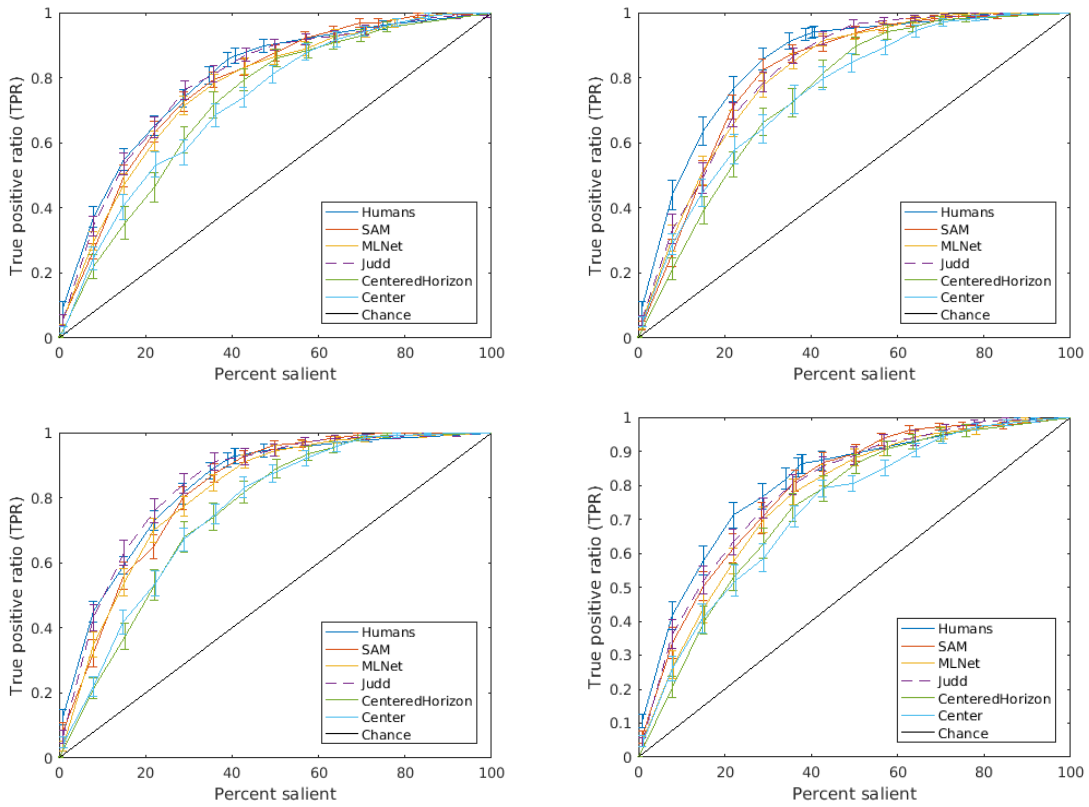


Figura 6.9: TPR medio y desvío estándar de la media de la tercera fijación. Cada gráfico corresponde a un conjunto distinto de sujetos de entrenamiento. Se entrenaron 5 modelos de Judd extendidos por cada conjunto de sujetos y se grafica el 3° en cuanto a performance (la mediana).

Para obtener resultados sobre más datos experimentaremos sin separar los sujetos en dos subconjuntos. En este caso se puede observar que en las imágenes de prueba se obtiene una performance significativamente mejor con Judd extendido que con los otros modelos (ver figura 6.10). Esta mejoría puede deberse a varios factores, entre ellos: una mayor cantidad de datos para entrenar el modelo y para evaluarlo; conjuntos de imágenes de prueba poco favorecedores; tener información

sobre el comportamiento de todo el conjunto de personas en 100 imágenes, lo que puede ayudar a deducir las fijaciones en las imágenes restantes.

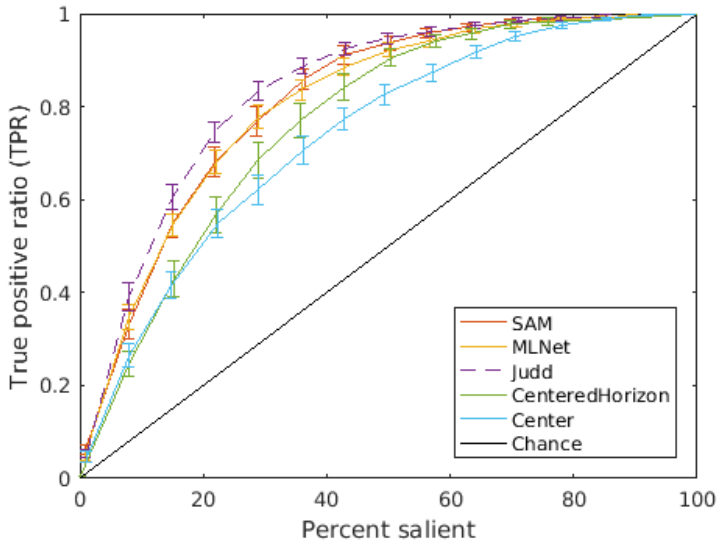


Figura 6.10: TPR medio y desvío estándar de la media sobre las terceras fijaciones de todos los sujetos.

6.2.3.2. Modelado de las fijaciones 4 a 6

El modelo de Judd extendido logra mejorar marginalmente la performance de los modelos de saliencia para algunos conjuntos de entrenamiento y logra una performance semejante para otros (ver figura 6.11). Tomando la mediana de las performances de los 5 modelos de Judd extendido entrenados se observa una mejora muy marginal de la performance respecto a los modelos de saliencia.

6.2.3.3. Modelado de las fijaciones 7 a 12

En este caso no podremos graficar los resultados separando los sujetos en un subconjunto de entrenamiento y de prueba debido a que la cantidad de fijaciones por imagen es muy reducida. Como la mayoría de los ensayos no llegan hasta la 7° fijación, quedan varias imágenes del conjunto de entrenamiento o prueba sin ninguna fijación para entrenar el modelo de Judd (recordemos que se crean dos mapas por imagen, uno para el conjunto de entrenamiento de sujetos y otro para el conjunto de prueba).

Observamos que la performance de nuestro modelo de Judd extendido se degrada notablemente, a tal punto que MLNet lo supera en performance (ver figura 6.12). Esto sugiere que si bien nuestras features ayudan a predecir las primeras fijaciones, otros factores pesan más en fijaciones posteriores. Estos son factores que apuntaremos a cubrir con los modelos dinámicos.

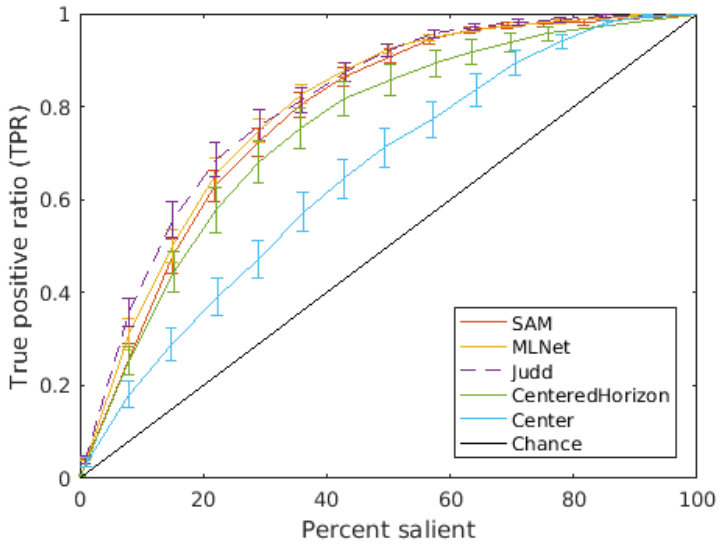


Figura 6.11: TPR medio y desvío estándar de la media sobre las fijaciones 4 a 6 de todos los sujetos.

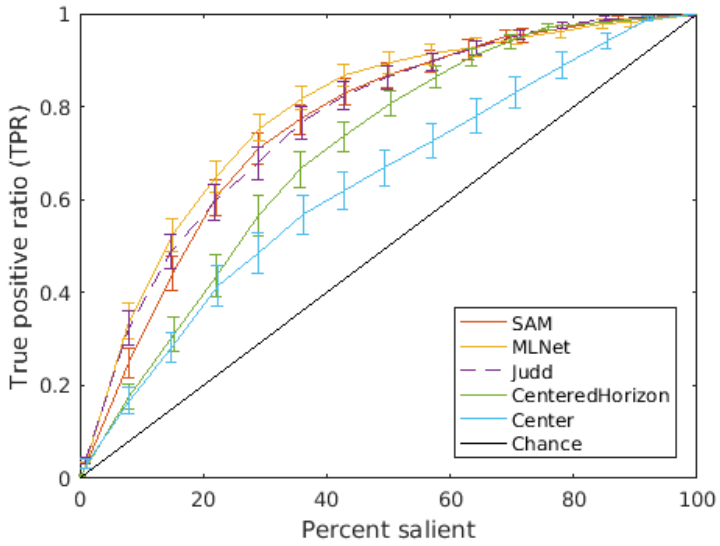


Figura 6.12: TPR medio y desvío estándar de la media sobre las fijaciones 7 a 12 de todos los sujetos.

6.2.4. Aporte de cada feature agregada al modelo de Judd original

Para calcular cuánto aportó cada feature agregada al modelo de Judd original entrenamos 8 modelos, cada uno con un subconjunto distinto de las 3 features que agregamos. Cada uno de estos 8 modelos fue entrenado 10 veces, cada vez con un conjunto distinto de 100 imágenes de entrenamiento. Todos fueron entrenados sobre las terceras fijaciones de todos los sujetos. Para cuantificar la performance de cada modelo calculamos el área bajo curva truncando la curva en distintos umbrales de

saliencia. De esta forma, podemos analizar no solo el área final sino la evolución hacia ella. Se promedia para cada uno de los 8 modelos el área bajo la curva obtenida para cada uno de los 10 conjuntos de entrenamiento tomados en cuenta.

Graficamos las áreas debajo de la curva promedio para los 8 modelos en 4 umbrales de saliencia diferentes, que pueden verse en la figura 6.13. Si bien graficamos solo 4 umbrales, analizamos varios más. En general se observa una diferencia reducida en el área de la curva entre los modelos para todos los umbrales de saliencia, aunque para porcentajes bajos se alcanzan diferencias mayores en las comparaciones. La performance no es consistente: no siempre el modelo con todas las features es mejor que las otras versiones, aunque siempre es mejor que el modelo sin ninguna feature agregada.

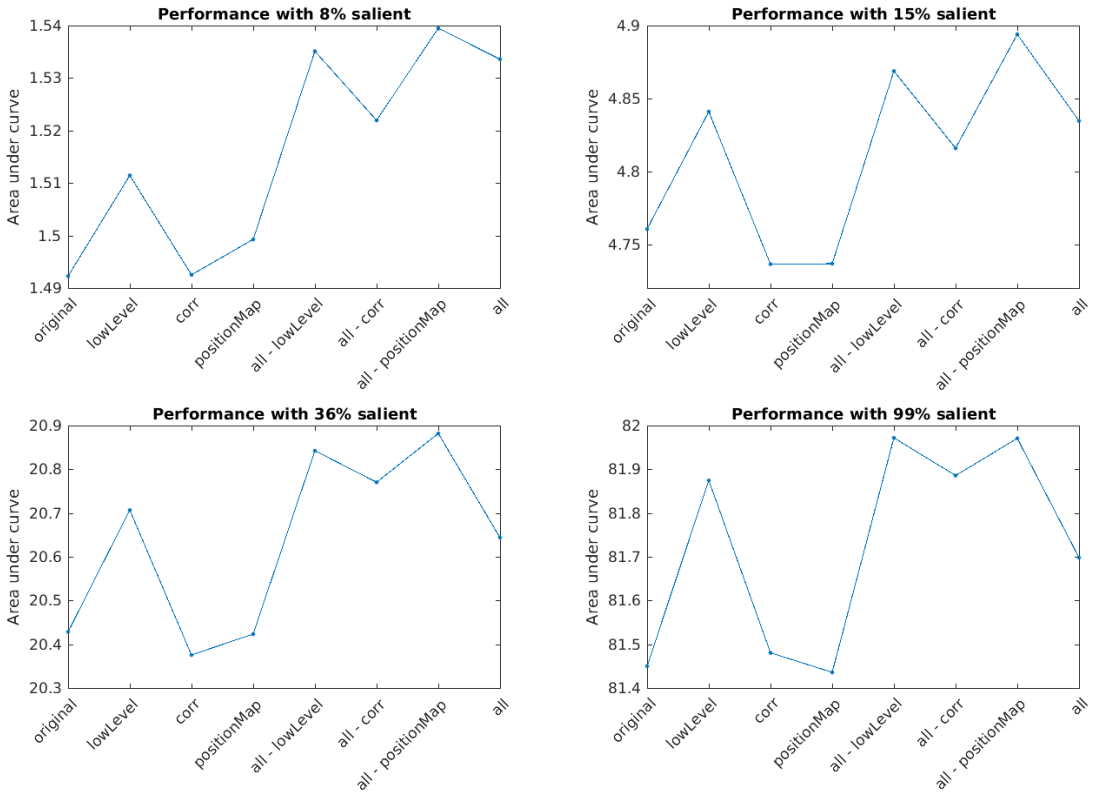


Figura 6.13: Comparación de áreas abajo de la curva para el modelo de Judd original sumándole todos los subconjuntos posibles del conjunto {`corr`, `lowLevel`, `positionMap`} y truncando las curvas en distintos niveles de saliencia. Las dos primeras features son las desarrolladas en la sección 6.1.3.1 mientras que la tercera es la explicada en la sección 6.1.3.2.

La feature de similitud de bajo nivel es la que mejor resultado da de las tres features al agregarla individualmente (ver detalle en la sección 6.1.3.1, en los gráficos la llamamos *lowLevel*), a tal punto que las otras dos no logran mejorar la performance

del modelo original. Esto no significa que estas dos features sean inútiles, ya que el modelo que entrena con todas las features menos la de similitud de bajo nivel alcanza muy buenos resultados (hasta a veces mejores que el modelo completo). De todas formas, volvemos a remarcar que las diferencias entre magnitudes son pequeñas, lo que no permite descartar la utilidad de ninguna de las features.

6.2.5. Predicción del centro del círculo de respuesta

Así como utilizamos el modelo de Judd para predecir las fijaciones, podemos utilizarlo para aprender los puntos que los sujetos seleccionaron como centro de su círculo de respuesta. También utilizaremos los otros modelos de saliencia y evaluaremos el nivel de predicción de cada uno de ellos, aunque ni MLNet ni SAM podrán aprender de estos datos. Esto nos permitirá saber cuán salientes son las ubicaciones de los centros de los círculos de respuesta y así poder describir mejor las características del reporte subjetivo.

Para este punto agregamos un feature más al modelo de Judd expandido, que es un mapa de distancia al target. Es equivalente al mapa de distancia al centro, pero se encuentra centrado en la posición del target. Esto será útil para modelar a los sujetos que encontraron el target y clickean en él.

Para nuestro primer experimento tomamos los centros de los círculos de respuesta de los 28 sujetos y creamos un mapa para cada imagen, de igual forma que hicimos con sus fijaciones. Separamos las 134 imágenes en 100 imágenes para el conjunto de entrenamiento y 34 imágenes para el conjunto de prueba. Entrenamos 5 modelos de Judd expandidos, utilizando cada vez un conjunto de entrenamiento diferente, que fue elegido aleatoriamente. Luego graficamos la performance de cada uno de los métodos en el conjunto de prueba de cada modelo. Se puede ver que si bien MLNet y SAM tienen un poder predictivo no despreciable, sugiriendo que muchas veces los sujetos clickean en regiones salientes, el modelo de Judd expandido logra una performance muy superior (ver figura 6.14). Además, para umbrales de saliencia muy pequeños el feature nuevo (“Target”, en verde) predice muy bien gran parte de las respuestas: estas son las respuestas de los sujetos que encontraron el target.

Es de interés ver cómo se comporta nuestro algoritmo si quitamos las respuestas de los sujetos que encontraron el target. Así, repetimos el mismo procedimiento pero dejando solo las respuestas de los ensayos donde registramos que el target no fue encontrado. La performance de los modelos es menor que en el caso anterior, pero se mantiene muy superior a los modelos de saliencia (ver figura 6.15). En ambos casos se ve que si bien el nivel de predicción es significativamente mejor a los modelos

de saliencia, aún no llega al nivel de los humanos. Además, al ser nuestro modelo basado en *machine learning* la interpretación cualitativa de qué áreas son las más propensas a ser seleccionadas como centro del círculo se dificulta notablemente. Sin embargo, observando los pesos dados por el algoritmo a cada feature vemos que tanto las features de saliencia (SAM especialmente) como las features que agregamos al modelo de Judd tuvieron un factor determinante.

Como usamos los 28 sujetos para crear todos nuestros mapas, en realidad lo que podemos concluir es que conociendo las respuestas de todos los sujetos en 100 imágenes podemos deducir con alta confianza dónde serán las respuestas de esos mismos sujetos en las imágenes restantes. Sin embargo una pregunta se mantiene abierta: ¿podemos deducir las respuestas de sujetos nuevos sin haber utilizado ninguna de sus respuestas en el conjunto de entrenamiento? Con el objetivo de responderla realizamos un último experimento, en el que separamos los sujetos en dos conjuntos: uno de 21 sujetos (el conjunto de entrenamiento) y otro de 7 sujetos (el conjunto de prueba). Entrenamos 5 modelos de Judd expandidos de forma idéntica a lo anterior y graficamos la performance de cada modelo de Judd junto con los modelos de saliencia tradicionales y teniendo en cuenta únicamente las 34 imágenes del conjunto de prueba del modelo de Judd. En este caso, solo hicimos pruebas tomando todas las respuestas de los sujetos (sin importar si encontraron o no el target) ya que al filtrar solo los casos donde el target no fue encontrado quedaban muy pocos datos en cada conjunto: recordemos que el conjunto de prueba consta de las respuestas de 7 sujetos en 34 imágenes y aplicar un filtro solo lo reduciría más.

De igual forma que antes, nuestro modelo logra predecir mejor los centros de los círculos de respuesta significativamente mejor que los otros (ver figura 6.16), alcanzando una performance casi equivalente a la de los humanos para los cuatro conjuntos de sujetos de entrenamiento elegidos al azar. Se graficaron dos modelos para cada conjunto de sujetos de entrenamiento, pero se entrenaron tres modelos más, todos ellos con un comportamiento similar.

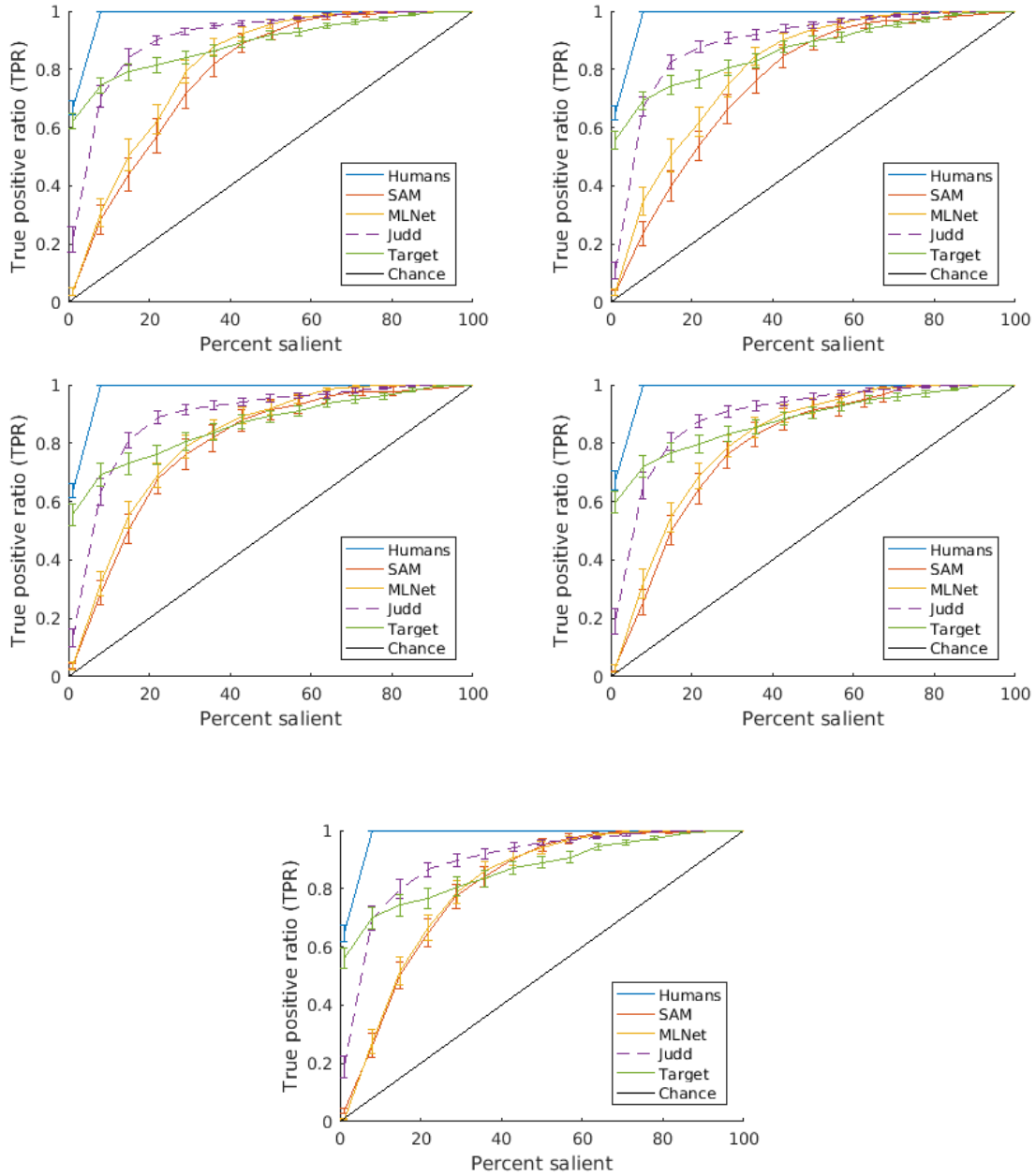


Figura 6.14: TPR medio y desvío estándar de la media del experimento donde se toman los centros de los círculos de respuesta de todos los sujetos y se crea un mapa por imagen. Se toman 100 imágenes como conjunto de entrenamiento y se evalúa la performance sobre las otras 34: cada uno de los gráficos corresponde a un conjunto de 100 imágenes diferente, elegido al azar.

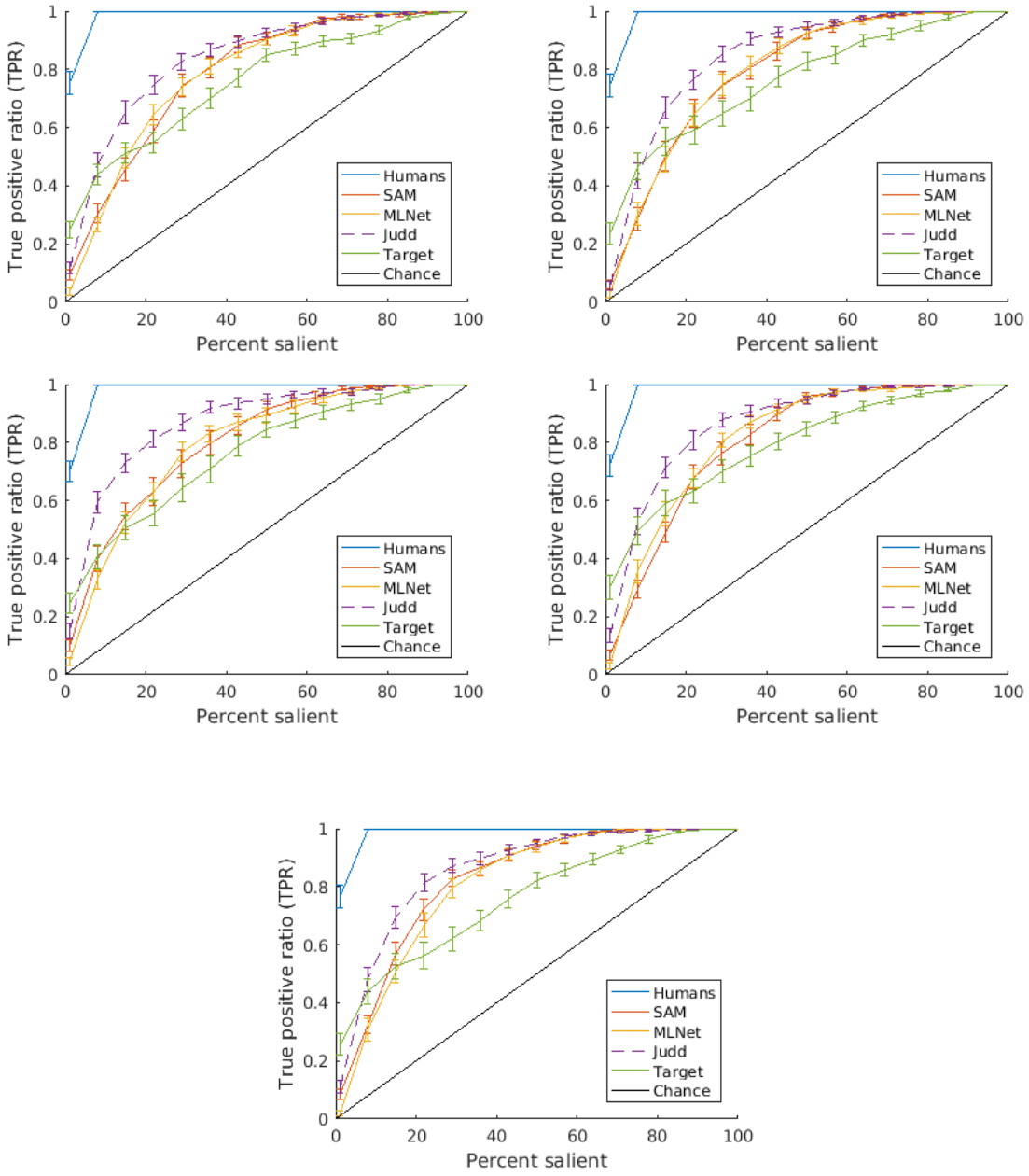


Figura 6.15: TPR medio y desvío estándar de la media del experimento donde se toman los centros de los círculos de respuesta de todos los sujetos y se crea un mapa por imagen, tomando únicamente aquellos ensayos donde no detectamos que el target haya sido encontrado. Se toman 100 imágenes como conjunto de entrenamiento y se evalúa la performance sobre las otras 34: cada uno de los gráficos corresponde a un conjunto de 100 imágenes diferente, elegido al azar.

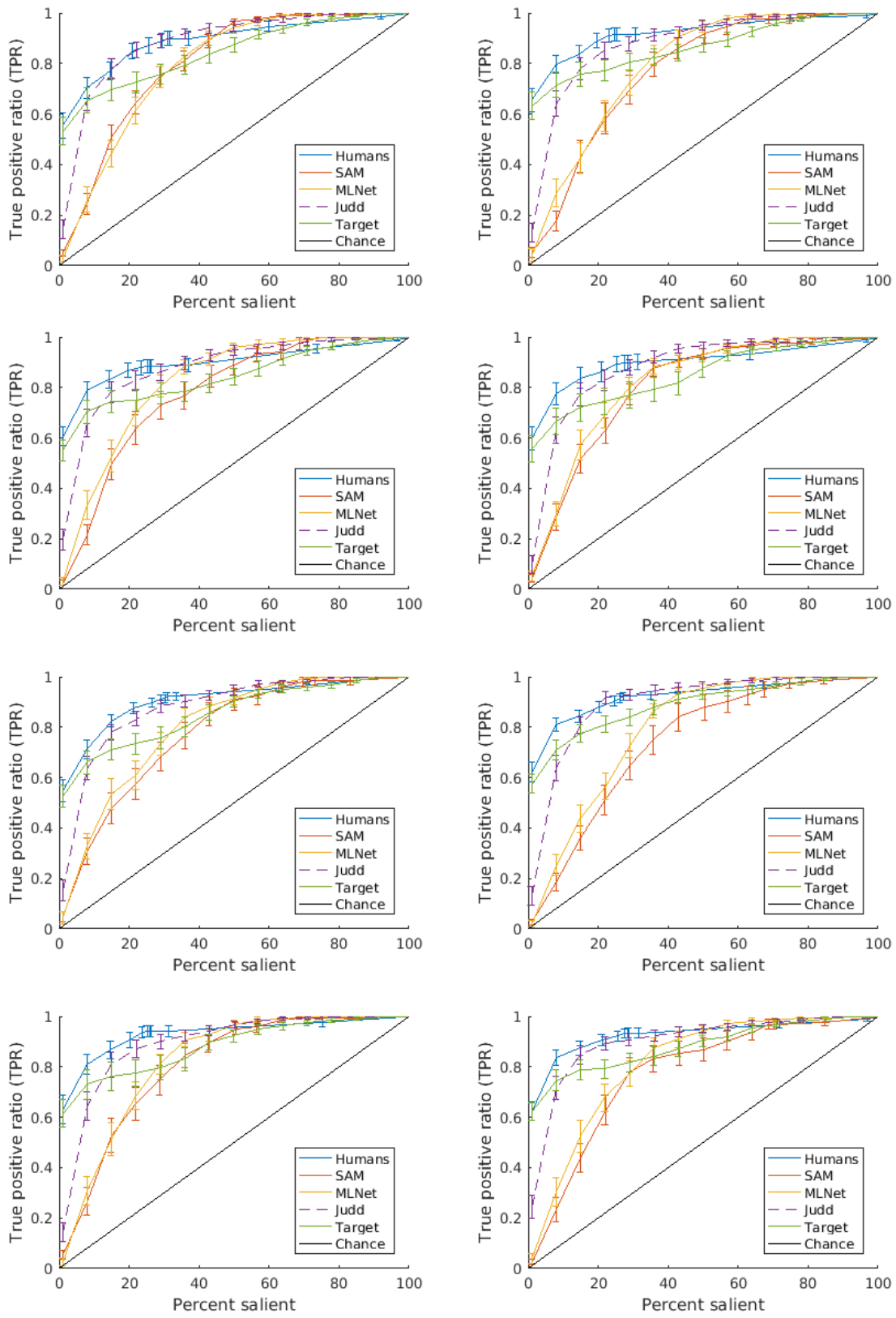


Figura 6.16: TPR medio y desvío estándar de la media del experimento donde se toman los centros de los círculos de respuesta de $\frac{3}{4}$ de los sujetos como conjunto de entrenamiento y se crea un mapa por imagen. Se toman 100 imágenes como conjunto de entrenamiento y se evalúa la performance sobre las otras 34 en el $\frac{1}{4}$ de los sujetos no utilizado. En cada fila se muestran dos modelos de un conjunto de sujetos de entrenamiento distinto, elegido al azar. Los dos modelos de cada fila fueron entrenados con un conjunto de 100 imágenes distintos, elegidos al azar.

Capítulo 7

Modelos dinámicos

En el capítulo anterior analizamos diferentes modelos que ayudan a predecir las regiones donde es más probable que caigan las fijaciones de los sujetos. En esta sección buscaremos crear modelos que aprovechen esa información para predecir scanpaths posibles para una imagen. Es decir que además de predecir las áreas donde recaerán las fijaciones buscaremos predecir posiciones concretas de las fijaciones y un orden probable de las mismas. Para construir los modelos dinámicos más sofisticados aplicaremos el mejor de los modelos estáticos para utilizarlo como predicción de saliencia inicial en modelos que utilicen las fijaciones previamente realizadas como factor de decisión para la próxima fijación.

A continuación presentaremos cinco modelos. Los primeros dos son modelos baseline. El tercero será un modelo *greedy* pues siempre realizará la fijación en el lugar que considera más probable, sin tener en cuenta fijaciones futuras. Los últimos modelos son modelos bayesianos: el primero será el modelo de Najemnik & Geisler, presentado en la sección 2.4, y luego propondremos una modificación a un aspecto del modelo.

Posteriormente analizaremos los resultados de los cinco modelos sobre varias métricas de comparación de scanpaths que desarrollamos específicamente para nuestra tarea y que describiremos en detalle.

7.1. Presentación de los modelos

7.1.1. Modelo de sesgo de la fijación central

Este modelo consiste en tomar una gaussiana bidimensional como mapa de probabilidad, cuyo pico se encuentra en el centro de la imagen. Este modelo se basa las observaciones descritas en la sección 5.2.4, que muestran que los humanos tienen

un sesgo hacia realizar fijaciones en el área central de la imagen.

En este modelo cada fijación es elegida al azar siguiendo el mapa bidimensional antes descrito. Se utilizó una gaussiana bidimensional $\mathcal{N}\left(\begin{bmatrix} x_0 \\ y_0 \end{bmatrix}, \begin{bmatrix} 2600 & 0 \\ 0 & 4000 \end{bmatrix}\right)$ donde (x_0, y_0) es el punto fijado. Esta gaussiana imita el mapa de visibilidad descrito por Najemnik & Geisler (ver ejemplo en la figura 2.6).

7.1.2. Modelo dinámico estadístico

El modelo dinámico estadístico consiste en elegir un mapa de saliencia e interpretarlo como un mapa de probabilidades. Para seleccionar cada fijación se toman las N posiciones más salientes de la imagen según ese mapa y se calcula la probabilidad de que tal movimiento sea realizado. Esta probabilidad involucra dos aspectos: la longitud de la sacada y el ángulo formado por la misma (si hay por lo menos tres fijaciones). Se selecciona la posición cuyo producto entre la saliencia y las probabilidades ya dichas sea máximo. Finalmente, se actualiza el mapa de probabilidades, bajando sensiblemente la probabilidad de las posiciones más visibles desde la última fijación.

Algorithm 1 Modelo Estadístico

procedure ModeloEstadístico(max_fijaciones, fijacion_inicial)
  $p \leftarrow$ mapa de saliencia
  $f_1 \leftarrow fijacion\_inicial$, $i \leftarrow 2$
  while $i < max\_fijaciones \wedge f_i \notin target$ **do**
    $candidatos \leftarrow N$ posiciones más salientes según p
    $f_{i+1} \leftarrow \underset{c \in candidatos}{\text{máx}} \{saliencia(c) \cdot probAngulo(f_{i-1}\widehat{f_i}c, i) \cdot probSacada(\overline{f_i c}, i)\}$
    $p \leftarrow actualizarProbabilidades(p, f_{i+1})$
    $i \leftarrow i + 1$

Para actualizar las probabilidades se considera que el mapa de visibilidad es una gaussiana bidimensional y se reduce la probabilidad proporcionalmente a esa gaussiana de forma tal de dejar una probabilidad muy baja pero no nula en el punto que acaba de ser fijado.

7.1.3. Modelo *greedy*

Este modelo realiza una fijación en la posición más probable según el *prior* utilizado (en nuestro caso, el mapa de saliencia de Judd expandido). Esto se contrasta con el modelo de Najemnik & Geisler, que elige la posición que maximizará la probabilidad de identificar la ubicación del target luego de esa fijación. En este sentido, el modelo *greedy* es más ingenuo pues no considera cómo quedará el mapa de probabilidades luego de realizar la fijación.

Algorithm 2 Modelo Greedy

```
procedure ModeloGreedy(max_fijaciones, fijacion_inicial)
    p ← mapa de saliencia
    f_1 ← fijacion_inicial, i ← 2
    while i < max_fijaciones ∧ f_i ∉ target do
        f_{i+1} ← máx_{c∈posiciones} saliencia(c)
        p ← actualizarProbabilidades(p, f_{i+1})
        i ← i + 1
```

7.1.4. Modelo de Najemnik & Geisler

Replicamos el modelo de Najemnik & Geisler explicado en la sección 2.4, pero a diferencia de su versión original lo haremos para escenas naturales. Experimentaremos tomando como *prior* el modelo de saliencia MLNet y el modelo de Judd extendido con nuestras features explicadas en el capítulo anterior. Como d'_{ij} tomaremos una gaussiana bidimensional centrada en $j = (x_j, y_j)$, definida como $\mathcal{N}\left(\begin{bmatrix} x_j \\ y_j \end{bmatrix}, \begin{bmatrix} 2600 & 0 \\ 0 & 4000 \end{bmatrix}\right)$. Como posibles ubicaciones de las fijaciones tomaremos una grilla de puntos equiespaciados por δ píxeles y cuyo punto superior izquierdo será top_left. Más precisamente, definimos al conjunto de posiciones que consideraremos de la siguiente manera:

$$\mathcal{P} = \left\{ top\_left + \delta \cdot (i,j) \;\middle|\; 0 \leq i \leq \left\lfloor \frac{1024 - top\_left_1}{\delta} \right\rfloor, 0 \leq j \leq \left\lfloor \frac{768 - top\_left_2}{\delta} \right\rfloor \right\}$$

Así se reduce la dimensionalidad del problema, ya que sería imposible tomar cada píxel como una posible ubicación de la fijación por el tiempo de cómputo que se requeriría. Decidimos utilizar $\delta = 50$ y $top\_left = (25, 25)$, de forma tal que los puntos están separados por 1,4° del campo visual. Estos valores de δ y top_left dan una grilla de 15×20 posibles ubicaciones de fijaciones. Se seleccionó este δ y no uno menor ya que el algoritmo es muy costoso computacionamente: en el trabajo original solo había 25 ubicaciones posibles para las fijaciones, y si bien optimizamos la implementación no pudimos reducir su complejidad teórica.

W es casi el mismo que en el modelo original, y lo denominaremos W^1. La única modificación surge de que fijamos $Im(d') = [0, 1]$, así que para que la varianza esté acotada adicionamos una constante.

$$\mu^1_{ik(t)} = \begin{cases} 0{,}5 & si\ i \in\ ubicación\ del\ target \\ -0{,}5 & caso\ contrario \end{cases}$$

$$\sigma^1_{ik(t)} = \frac{1}{a \cdot d'_{ik(t)} + b} \text{ con } a = 3 \text{ y } b = 4$$

Así $W^1_{ik(t)} \in \mathcal{N}(\mu^1_{ik(t)}, \sigma^1_{ik(t)})$. Más adelante experimentaremos con otros pares (a, b) además de $(3, 4)$.

7.1.4.1. Implementación del algoritmo original y optimizaciones

Implementamos el modelo propuesto por Geisler desde cero, ya que no existen implementaciones públicas de este modelo. Con este trabajo hacemos pública la implementación del modelo de Geisler. En la sección 2.4 se puede ver la explicación detallada del modelo, y a continuación vemos un pseudocódigo:

Algorithm 3 Modelo de Najemnik y Geisler

procedure ModeloNajemnikGeisler(max_fijaciones, centro_target, fijacion_inicial, modoW, modoPrior)
  $prior \leftarrow$ inicializarPrior(modoPrior)
  $d' \leftarrow$ mapa de visibilidad
  $k_1 \leftarrow fijacion\_inicial$
  $T \leftarrow 2$
  while $T < max\_fijaciones \wedge k_T \notin target$ **do**
    $W \leftarrow respuestaTarget(modoW,\ d',\ centro\_target)$
    for $i \in \{1, \ldots, n\}$ **do**
      $s_i(T) \leftarrow prior(i) \cdot \prod_{t=1}^{T} exp\left(d'^2_{ik(t)} W_{ik(t)}\right)$
      $p_i(T) \leftarrow \frac{s_i(T)}{\sum_{j=1}^{n} s_j(T)}$
    for $posible\_k \in \{1, \ldots, n\}$ **do**
      for $i \in \{1, \ldots, n\}$ **do**
        $$p(C|i, posible\_k) \leftarrow \int_{-\infty}^{+\infty} \phi(w) \prod_{j \neq i} \Phi\left(\frac{2d'_{i\ posible\_k} w - 2\ln\frac{p_j(T)}{p_i(T)} + d'^2_{j\ posible\_k} + d'^2_{i\ posible\_k}}{2d'_{j\ posible\_k}}\right) dw$$
    $k_{T+1} \leftarrow \arg\max_{posible\_k} \sum_{i=1}^{n} p_i(T) \cdot p(C|i, posible\_k)$
    $T \leftarrow T + 1$
  return k

Optimizamos el algoritmo aprovechando la vectorización de MATLAB, en particular en el primer ciclo for y en el cómputo de W y d'. Además, una optimización clave fue la del cálculo de la integral: vectorizamos el cálculo de la integral y calculamos analíticamente el intervalo donde no se puede asegurar que el valor puntual de la integral sea 0, de forma tal que solo integraremos sobre ese intervalo. Este cálculo se basa en que la función de densidad de la normal estándar phi cumple que $phi(w) < 10^{-87}$ si $w \notin [-20, 20]$, y que la función de distribución acumulada $\Phi(x)$ cumple que $\Phi(x) < 10^{-88}$ para $x < -20$.

Más en detalle, utilizamos la segunda propiedad de la forma que se describe a continuación. Primeramente, reescribimos el argumento de Φ en la integral como $\Phi(m_j w + b_j)$ donde

$$m_j = \frac{d'_{ik(T+1)}}{d'_{jk(T+1)}}$$

$$b_j = \frac{-2 \ln \frac{p_j(T)}{p_i(T)} + d'^2_{jk(T+1)} + d'^2_{ik(T+1)}}{2d'_{jk(T+1)}}$$

Sabemos que si al menos un factor de la productoria es ínfimo, toda la productoria lo será pues $0 \leq \Phi(x) \leq 1$. Así, basta con que para algún j valga $m_j w + b_j < -20$ para que toda la productoria sea nula y no haga falta evaluar la integral sobre ese w. Dicho de otra forma, para todo j debe valer que $m_j\ w + b_j \geq -20$.

Si $m_j > 0$, tenemos que $m_j\ w + b_j \geq -20 \Leftrightarrow w \geq \frac{-20 - b_j}{m_j}$. Por lo tanto, $w \geq \underset{j\ |\ m_j>0}{\text{máx}} \frac{-20 - b_j}{m_j}$. Además, como ya afirmamos que $w \in [-20, 20]$, podemos concluir que:

$$w \geq \text{máx}\left(\underset{j|m_j>0}{\text{máx}} \frac{-20 - b_j}{m_j}, -20\right)$$

Similarmente, para $m_j < 0$ se deduce que

$$w \leq \text{mín}\left(\underset{j|m_j<0}{\text{mín}} \frac{-20 - b_j}{m_j},\ 20\right)$$

Estos cálculos permitieron reducir sensiblemente la cantidad de cálculos necesarios para alcanzar una buena precisión numérica, logrando reducir el tiempo total a un tercio del tiempo original. Esto se debe en especial a que logramos no hacer ningún cálculo en los casos donde detectamos que el valor de la integral será 0.

7.1.5. Modelo Geisler modificado

El modelo anteriormente mencionado no tiene en cuenta la presencia de distractores que existe en escenas naturales, ya que en la tarea original no existían distractores. Los *distractores* son puntos en una imagen que no son el target pero

que tienen similitud con él. Es decir que los sujetos podrían confundirse si el nivel de visibilidad es bajo.

Para reflejar esto decidimos modificar W de modo tal que su valor esperado no solo dependa de la presencia del target sino de la similitud con él. Como medida de similaridad entre cada porción de la imagen y el target utilizamos *cross-correlation*, de forma tal de que a cada elemento de $\mathcal{P}$ se le asigna un valor de similitud al target. De ahora en más, llamamos a esta función $corr : \mathcal{P} \to [-0{,}5, 0{,}5]$. Numéricamente, calculamos la cross-correlation de todos los puntos de la imagen original y tomamos el valor de $corr(i)$ como la correlación máxima de todos los puntos j que cumplen que $\forall i_1 \in \mathcal{P}, i_1 \neq i$, $\|i - j\|_2 \leq \|i_1 - j\|_2$.

$$\mu^2_{ik(t)} = \mu^1_{ik(t)} \cdot \left(d'_{ik(t)} + \frac{1}{2}\right) + corr_i \cdot \left(\frac{3}{2} - d'_{ik(t)}\right) \in [-1, 1]$$

$$\sigma^2_{ik(t)} = \sigma^1_{ik(t)}$$

Así definimos un nuevo W, $W^2 \in \mathcal{N}(\mu^2_{ik(t)}, \sigma^2_{ik(t)})$.

7.2. Métodos de comparación de scanpaths adaptados a nuestra tarea

En nuestra tarea, diferentes imágenes toman diferente cantidad de fijaciones hasta lograr encontrar el target. Esta característica la hace diferir de otras tareas de la literatura, especialmente de tareas sobre escenas artificiales, y nos lleva a repensar las métricas de comparación de scanpaths ya discutidas. Además, otra diferencia crucial es que en nuestro experimento no todos los ensayos terminan cuando el target fue encontrado, ya que algunos terminan cuando se alcanzó la máxima cantidad de fijaciones permitidas. Presentaremos a continuación tres métricas, cada una de ellas con el objetivo de medir la similitud entre humanos y los modelos propuestos en tres aspectos diferentes.

Calcularemos el valor de todas las métricas para cada uno de los humanos, de manera de poder tener referencia de cuáles son los parámetros usuales humanos. Ahora bien, dado el valor de una métrica para un modelo dinámico, ¿cómo sabemos si éste se encuentra dentro de los parámetros humanos usuales? Veremos que para las tres métricas analizadas la distribución de la muestra de todos los humanos se asemeja a una distribución normal. Comprobaremos esto utilizando varios tests de normalidad, que son tests de hipótesis que contrastan los datos contra la hipótesis

nula de que los datos tienen una distribución normal. Además, mostraremos boxplots de los datos para tener una representación gráfica de los datos.

Como las muestras tienen distribuciones semejantes a una normal podemos utilizar el *z-score* como una métrica significativa de la similitud de un modelo dinámico con el comportamiento de los humanos. El *z-score* indica a cuántos desvíos estándar de la media se encuentra un valor. Formalmente podemos definir el $z-score$ de un valor x como

$$z = \left|\frac{x - \mu}{\sigma}\right|$$

donde μ es la media y σ el desvío estándar de la población, que estimaremos con la muestra obtenida con los humanos.

Esta medida de dispersión será utilizada en las tres métricas propuestas, que explicamos a continuación.

7.2.1. Métrica de número de fijaciones esperadas para encontrar el target, solo para ensayos exitosos

Para el caso de la métrica de comparación por cantidad de fijaciones calcularemos la cantidad media de fijaciones que se requiere para encontrar el target en cada imagen, extrayendo los datos sobre un conjunto de sujetos que denotaremos *conjunto de entrenamiento*. Luego consideraremos la cantidad de fijaciones que demoró el nuevo sujeto para encontrar el target (asumiendo que lo haya hecho) y calcularemos la diferencia de esta magnitud con la cantidad promedio de fijaciones para esa imagen. Finalmente tomaremos la media y el desvío estándar sobre todas las imágenes y analizaremos si los parámetros de este sujeto están dentro de los parámetros humanos.

Formalizando lo anterior, si $cf_{s,m}$ es la cantidad de fijaciones que demoró el sujeto s en encontrar el target de la imagen m y

$$E = \{(s, m) \mid \text{el sujeto } s \text{ encontró el target en la imagen } m\}$$

se tiene que

$$\mu_m = \begin{cases} \underset{j:(j,m)\in E}{\text{avg}} \; cf_{j,m} & si \; |\{j : (j, m) \in E\}| \geq 3 \\ +\infty & \text{caso contrario} \end{cases}$$

$$mean\_length_s = \operatorname*{avg}_{m:(s,m)\in E} |cf_{s,m} - \mu_m|$$

$$stdev\_length_s = \operatorname*{stdev}_{m:(s,m)\in E} |cf_{s,m} - \mu_m|$$

Para evitar datos espurios solo se consideran los μ_m en donde $|\{j : (j,m) \in E\}| \geq 3$, y por eso su definición es una función partida.

Para determinar cuáles son los valores de $mean\_length_s$ y $stdev\_length_s$ considerados característicos de los humanos se toman todos los sujetos menos uno como conjunto de entrenamiento y se calcula $mean\_length_s$ y $stdev\_length_s$ para el sujeto restante. Este procedimiento se repite para todos los sujetos. Así, a la hora de evaluar si un nuevo sujeto es considerado humano o no se toman todos los humanos como conjunto de entrenamiento y se comparan los valores obtenidos por el modelo dinámico con los $mean\_length_s$ y $stdev\_length_s$ que calculamos previamente.

Para entender cuán cerca de la performance humana está un modelo utilizamos el *z-score* como explicamos anteriormente. Nuestra muestra estará compuesta de los $mean\_length_s$ para todo $s \in Sujetos$. Se puso a prueba la muestra con tres tests de normalidad (el test de Kolmogorov-Smirnov, el test de Lilliefors y el de Jarque-Bera) y ninguno de ellos pudo rechazar la hipótesis nula al 5 % de significación estadística. Gráficamente, se puede ver un esbozo de la distribución de $mean\_length_s$ y $stdev\_length_s$ en las figuras 7.1 y 7.2 respectivamente.

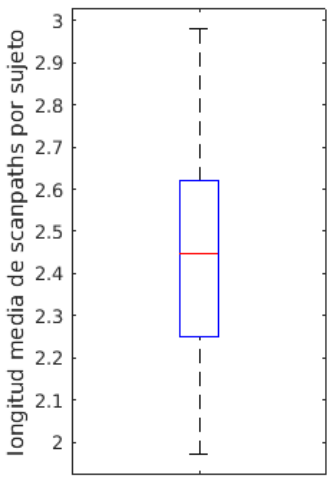


Figura 7.1: Boxplot de $mean\_length_s$ para todo $s \in Sujetos$

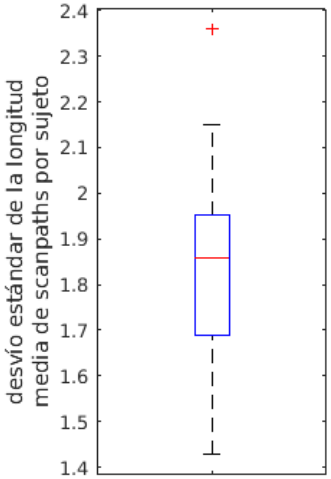


Figura 7.2: Boxplot de err_length_s para todo $s \in Sujetos$

Calculamos la media y el desvío estándar de la muestra para finalmente calcular el *z-score* para un nuevo modelo t:

$$\mu_1 = \operatorname*{avg}_{s \in Sujetos} mean\_length_s$$

$$\sigma_1 = \operatorname*{stdev}_{s \in Sujetos} mean\_length_s$$

$$z^t_{length} = \left| \frac{mean\_length_t - \mu_1}{\sigma_1} \right|$$

En la sección de resultados veremos el $z1$ de cada modelo dinámico y éste será un factor de decisión para seleccionar el mejor de los modelos.

Como el lector podrá observar, esta métrica tiene el problema de que solo se pueden considerar scanpaths en los que el target fue encontrado. Más aún, para comparar la performance de un nuevo sujeto en cierta imagen requerimos que éste haya encontrado el target y que al menos tres otros sujetos lo hayan hecho también. Esto significa que el nivel de dificultad de la tarea determina cuán útil es esta métrica. En la versión final del experimento los sujetos encontraron 61,74 % en promedio, haciendo que esta métrica tenga valor. Más aún, para los conjuntos de entrenamiento sobre los que experimentamos existen entre 2 y 4 imágenes que son descartadas por no alcanzar los requerimientos mínimos de scanpaths completos: esto significa que más del 97 % de las imágenes son consideradas para nuestro análisis.

Para subsanar el defecto de la métrica recién explicada es que proponemos otras dos métricas a continuación, que serán utilizadas en conjunto con la anterior.

7.2.2. Métrica de performance sobre todos los ensayos

Como se puede observar en la figura 7.3, existe cierta consistencia entre sujetos en el porcentaje de ensayos en los que encuentran el target según cuántas fijaciones se permitan. Esto se evidencia al calcular la proporción de ensayos en los que el target fue encontrado ya que el desvío estándar de la media es reducido y la proporción de targets encontrados se incrementa consistentemente conforme aumenta la cantidad de fijaciones. Con esto en mente es que decidimos introducir la métrica que veremos a continuación.

Sea $\mu_c \in [0, 1]$, $c = 2, 4, 8, 12$, la proporción de los ensayos con i sacadas permitidas en los que el target fue encontrado, incluyendo a todos los sujetos del conjunto de entrenamiento en el cálculo. Sea $\mu^s_c \in [0, 1]$, $c = 2, 4, 8, 12$ la proporción de ensayos del sujeto s con i sacadas permitidas en las que el target fue encontrado.

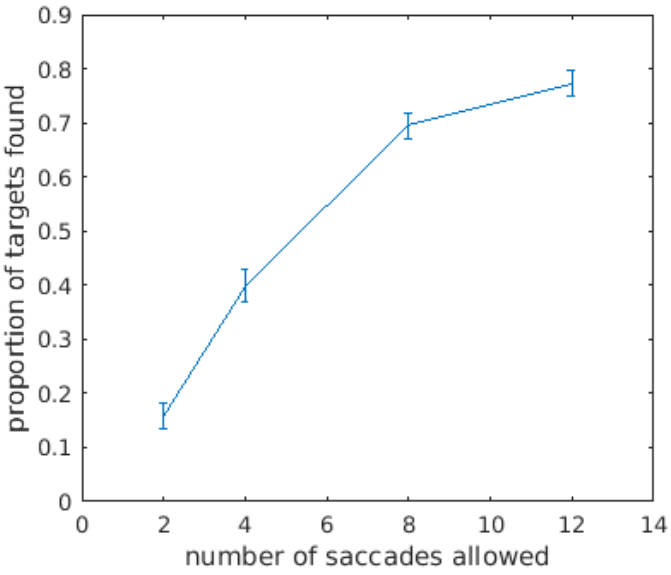


Figura 7.3: Proporción de imágenes en las que el target fue encontrado según la cantidad de sacadas permitidas. Las barras indican el desvío estándar de esta proporción.

En base a esto calculamos la media y el desvío estándar:

$$\mu_c = \operatorname*{avg}_{s \in Sujetos} \mu_c^s$$

$$\sigma_c = \operatorname*{stdev}_{s \in Sujetos} \mu_c^s$$

Así, dado un nuevo modelo t, calcularemos μ_c^t y su *z-score*, que indicará cuán similar es el modelo a los humanos. Más precisamente, el *z-score* se define como:

$$z^{c,t}_{\substack{proportion \\ found}} = \left| \frac{\mu_c - \mu_c^t}{\sigma_c} \right| \quad \text{para } c \in \{2, 4, 8, 12\}$$

En este punto estamos asumiendo que los μ_c^s son muestras de una distribución similar a la normal $\mathcal{N}(\mu_c, \sigma_c)$. Sabemos que la muestra no puede pertenecer a una normal pues no pueden existir datos fuera del rango $[0, 1]$, pero vemos que la distribución de los datos se asemeja suficiente. Para comprobar esta afirmación utilizamos tres tests de normalidad: el test de Kolmogorov–Smirnov, Lilliefors y Jarque-Bera, todos ellos configurados para rechazar la hipótesis nula al 5 % de significación estadística. A la hora de efectuar los tests separamos los datos en cuatro categorías, según la cantidad de sacadas máxima permitida. Los primeros dos tests no rechazaron la hipótesis nula para ninguno de los cuatro límites de cantidad de sacadas, mientras que el Jarque-Bera rechazó la hipótesis nula para las sacadas

máximas 8 y 12. Esto sugiere que los datos no están fuertemente sesgados, y que por lo tanto el *z-score* será una métrica valiosa.

Reforzando el concepto vertido por los test de normalidad pero en un sentido gráfico, mostramos un boxplot de los μ_c^s separados por c, su cantidad máxima de sacadas permitidas (ver figura 7.4). Se puede ver que en ninguna de las cuatro categorías la distribución es fuertemente sesgada, requerimiento para poder usar *z-score*. Para las sacadas máximas 8 y 12 se observa la presencia de un outlier. Removiendo este outlier, el test de normalidad de Jarque-Bera ya no consigue rechazar la hipótesis nula: esto sugiere que posiblemente con más sujetos estos rechazos del test de Jarque-Bera desaparecerían. Por lo tanto, dejamos como trabajo futuro amplificar nuestro conjunto de datos.

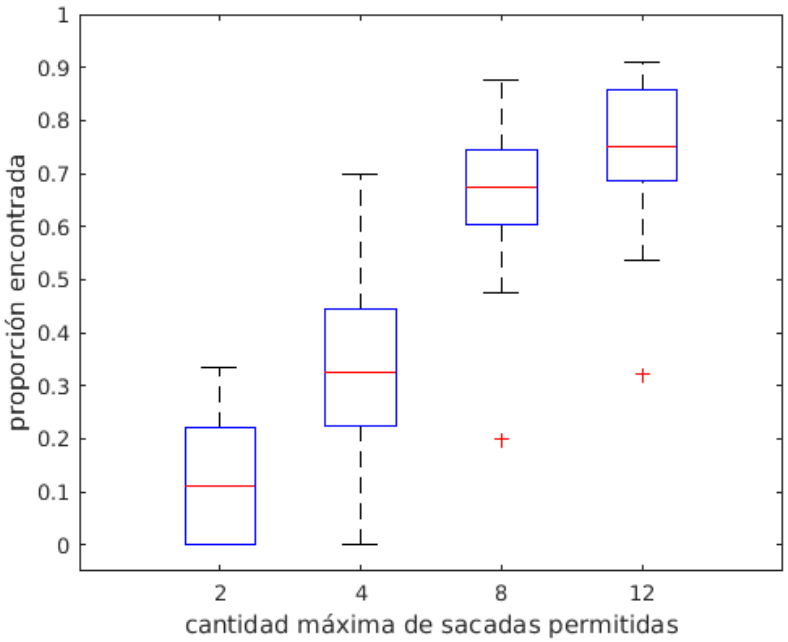


Figura 7.4: Boxplot de μ_c^s para todo $s \in Sujetos$ y $c \in \{2, 4, 8, 12\}$

7.2.3. Métrica de string edit distance

Hasta ahora las métricas descritas se focalizan en los resultados finales de cada ensayo pero no toman en cuenta la posición de cada una de las fijaciones. Por esto es que desarrollamos una métrica basada en string edit distance. En la introducción mencionamos cómo se puede comparar dos scanpaths usando la distancia Levenshtein, pero para poder utilizarlo en nuestro problema debemos extender esta métrica para comparar múltiples scanpaths. Josephson & Holmes [JH02] calculan la distancia Levenshtein para los scanpaths dos a dos y reportan estos resultados pero no lo utilizan como forma de comparación de modelos sino para luego graficar cada punto en un plano bidimensional usando multidimensional

scaling (MDS). Además, en nuestro caso no todos los scanpaths terminan por el mismo motivo, lo que complejiza el análisis.

Inspirados en lo anterior, decidimos calcular la distancia Levenshtein dos a dos separando en cuatro categorías según cantidad de fijaciones: tomamos todos los scanpaths de una misma imagen de longitud por lo menos 2, los truncamos tomando solo las primeras dos fijaciones, calculamos la distancia Levenshtein dos a dos y tomamos la media y la varianza de esa muestra. Lo mismo fue repetido para los scanpaths de longitud al menos 4, 6, 8 y 10. Así, tenemos la media de la distancia Levenshtein por imagen y en cada una de las cinco categorías (denominamos $\mu^{lev}_{m,c}$ a la media de la imagen m en la categoría c), pero aún resta saber cómo utilizar esta información para comparar sujetos con los modelos propuestos. Para esto computamos la media de la distancia Levenshtein de un sujeto respecto a todos los otros sujetos (que llamaremos *conjunto de entrenamiento*) en cada una de las imágenes y categorías. Luego calculamos la distancia de la media recién computada (que llamamos $\mu^{lev}_{m,c,s}$) a la media de todo el conjunto de entrenamiento y la media y la varianza a lo largo de todas las imágenes, que denominamos $mean\_lev_{s,c}$ y $stdev\_lev_{s,c}$. Más formalmente,

$$\mu^{lev}_{m,c,s} = \operatorname*{avg}_{s \neq i} \; lev(scanpath_{s,m}[1..c], scanpath_{i,m}[1..c])$$

$$mean\_lev_{s,c} = \operatorname*{avg}_{1 \leq m \leq |images|} |\mu^{lev}_{m,c} - \mu^{lev}_{m,c,s}|$$

$$stdev\_lev_{s,c} = \operatorname*{stdev}_{1 \leq m \leq |images|} |\mu^{lev}_{m,c} - \mu^{lev}_{m,c,s}|$$

Para nuestros cómputos dividimos la imagen en 70 regiones de interés (ROI en inglés) utilizando una grilla de 7 filas y 10 columnas (7×10). Así, la métrica solo distinguirá dos fijaciones como distintas cuando caigan en regiones diferentes de la imagen. Elegimos un valor no muy elevado ni muy reducido de regiones de interés, ya que un valor elevado haría que leves cambios en la ubicación de la fijación sean frecuentemente computados como un cambio de región, mientras que un valor muy reducido haría perder valor de esta métrica pues demasiadas fijaciones caerían en la misma región. La grilla de 7×10 divide la imagen en secciones de aproximadamente 100×100 píxeles.

Como hicimos para la métrica anterior, argumentaremos que los datos obtenidos no difieren fuertemente de una normal y mostraremos el boxplot correspondiente. Separamos los $mean\_lev_{s,c}$ en las cinco categorías según la cantidad

máxima de fijaciones c y aplicamos los tests de normalidad en cada una de ellas. En la gran mayoría de los casos los tests considerados no pudieron rechazar la hipótesis nula: Kolmogorov–Smirnov no pudo rechazar la hipótesis nula en ningún caso, mientras que Lilliefors rechazó la hipótesis nula para $c = 2$ y Jarque-Bera lo hizo para $c = 2$ y $c = 6$. Si bien la afirmación es más débil que en el caso anterior, todavía se puede argumentar que los datos no están fuertemente sesgados, que es lo importante para luego poder aplicar *z-score*. Visualmente podemos reforzar esto con el boxplot de la figura 7.5. Formalmente, definimos el *z-score* de esta métrica de la siguiente manera:

$$\mu_c = \operatorname*{avg}_{s \in Sujetos} mean\_lev_{s,c}$$

$$\sigma_c = \operatorname*{stdev}_{s \in Sujetos} stdev\_lev_{s,c}$$

$$z_{lev}^{c,t} = \left| \frac{\mu_c - \mu_c^t}{\sigma_c} \right| \quad \text{para } c \in \{2, 4, 6, 8, 10\}$$

Una observación interesante es que la edit distance media es en general menor a 1, siendo que la distancia Levenshtein de dos strings de longitud c puede ser cualquier número entero en el intervalo $[0, c]$: esto es un fuerte indicador de consistencia entre los sujetos. Además, $stdev\_lev_{s,c}$ tiende a ser muy reducido y consistente salvo para $c = 10$, que es una buena señal de consistencia, aunque no lo utilicemos como métrica (ver figura 7.6).

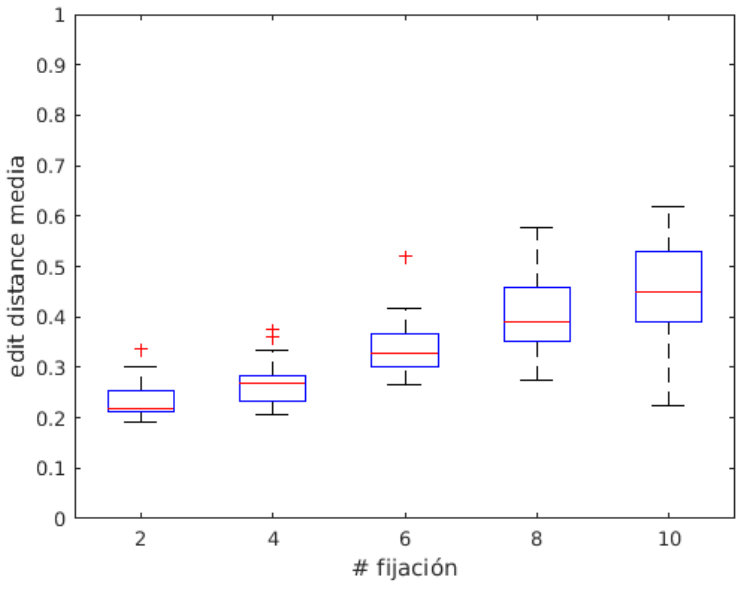


Figura 7.5: Boxplot de $mean\_lev_{s,c}$ para todo $s \in Sujetos$ y $c \in \{2, 4, 6, 8, 10\}$

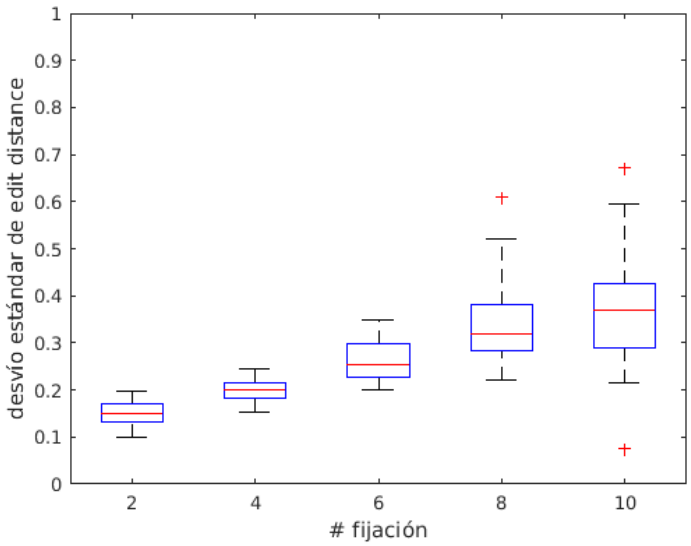


Figura 7.6: Boxplot de $stdev\_lev_{s,c}$ para todo $s \in Sujetos$ y $c \in \{2, 4, 6, 8, 10\}$

Excluimos el caso $c = 12$ pues se observaba un gran aumento en la varianza de los $mean\_lev_{s,12}$, que podría ser atribuible a que la cantidad de ensayos con datos para esta categoría es menor que para las anteriores. Además, al utilizar la métrica para comparar un nuevo modelo con los humanos notamos que muchas veces la cantidad de scanpaths de longitud mayor o igual a 12 no era suficiente para conseguir computar $mean\_lev_{t,12}$, perdiendo el valor de la métrica.

7.3. Resultados de los modelos dinámicos

En esta sección veremos la performance de cada modelo dinámico para las tres métricas mencionadas anteriormente en las categorías respectivas: z^{t}_{length}, $z^{c,t}_{\substack{proportion \\ found}}$ para $c \in \{2, 4, 8, 12\}$, $z^{c,t}_{lev}$ para $c \in \{2, 4, 6, 8, 10\}$, siendo t un modelo dinámico. Primero presentaremos los modelos menos sofisticados (el modelo del sesgo de la fijación central, el modelo estadístico y el modelo greedy) y su performance. Luego presentaremos el modelo de Geisler y el modelo de Geisler modificado, realizando algunas pequeñas variaciones para asegurar que la performance no esté degradándose por culpa de una mala elección de constantes o una mala elección del modelo estático utilizado como base.

Es por esto que utilizamos dos modelos estáticos distintos como base para los dos modelos dinámicos basados en Najemnik & Geisler: utilizamos el modelo de Judd extendido y el de MLNet. Vimos que el modelo de Judd extendido fue muy superior en performance y en consecuencia ejecutamos los siguientes modelos solo con él como *prior*. Cuantitativamente, la suma de los *z-score* para el algoritmo de Geisler original con constantes $(3, 4)$ utilizando MLNet es de 8.61 mientras que el mismo algoritmo pero utilizando el modelo de Judd extendido obtuvieron una suma de *z-score* de 5.14, lo que representa una reducción del 40 % y por ende una gran mejora en la performance.

A continuación variamos las constantes de $\sigma^{1}_{i,k(t)}$ en los dos modelos bayesianos. Originalmente, $\sigma^{1}_{i,k(t)}$ fue definido de la siguiente forma:

$$\sigma^{1}_{i,k(t)} = \frac{1}{a \cdot d'_{ik(t)} + b} \text{ con } a = 3 \text{ y } b = 4$$

Para nuestros experimentos variamos (a, b) por otros pares, a saber: $(a, b) = (3, 2)$, $(a, b) = (3, 6)$, $(a, b) = (4, 8)$ y $(a, b) = (5, 10)$.

Además probamos amplificar la visibilidad modificando los parámetros de la gaussiana, con resultados muy negativos que por lo tanto omitimos. Existen otras formas más sofisticadas de modelar la visibilidad, por ejemplo el modelo Retina-V1 [BAG14] que tiene en cuenta tanto la distancia angular a la fóvea como el contraste del target con la imagen en cada punto para crear d'. No experimentamos con este modelo para la tesis pues estimamos que computarlo para cada imagen y cada posible punto de fijación demoraría más de un mes de procesamiento: lo incluimos como trabajo futuro. De todas formas, la gaussiana que utilizamos es semejante a la visibilidad mostrada por los autores del modelo Retina-V1 en casos promedio.

Originalmente ejecutamos todos los modelos dinámicos con 12 sacadas permitidas y luego seleccionamos al azar los ensayos que tendrían 2, 4 y 8 sacadas máximas, respetando la distribución de la versión final del experimento. Estos ensayos se truncaron para dejar tantas fijaciones como la cantidad máxima de sacadas permitiera. Para evitar analizar un *z-score* fuertemente sesgado por la elección de los límites máximos de sacadas repetimos 10 veces este procedimiento y lo que se grafica será el promedio de los 10 *z-scores*.

7.3.1. Análisis de los modelos respecto de z^t_{length}

La primera métrica, z^t_{length}, logró discriminar muy bien entre los modelos bayesianos y los otros tres modelos (ver figura 7.7). Todos los modelos bayesianos menos $(a, b) = (3, 2)$ obtuvieron $z^t_{length} < 1{,}26$ (esto representa una distancia de a lo sumo 0,31 fijaciones a la media), mientras que los modelos estadístico, greedy y del sesgo de la fijación central obtuvieron $z^t_{length} > 2{,}35$ (esto representa una distancia de al menos 0,58 fijaciones a la media). Recordemos que como el *z-score* representa la distancia a la media de los humanos en unidades de desvío estándar, obtener valores reducidos es buena señal. Por lo tanto, esto significa que los modelos bayesianos reflejan bien este aspecto de la búsqueda visual humana ya que en general suelen estar a menos de un desvío estándar de la media sobre la que se analiza.

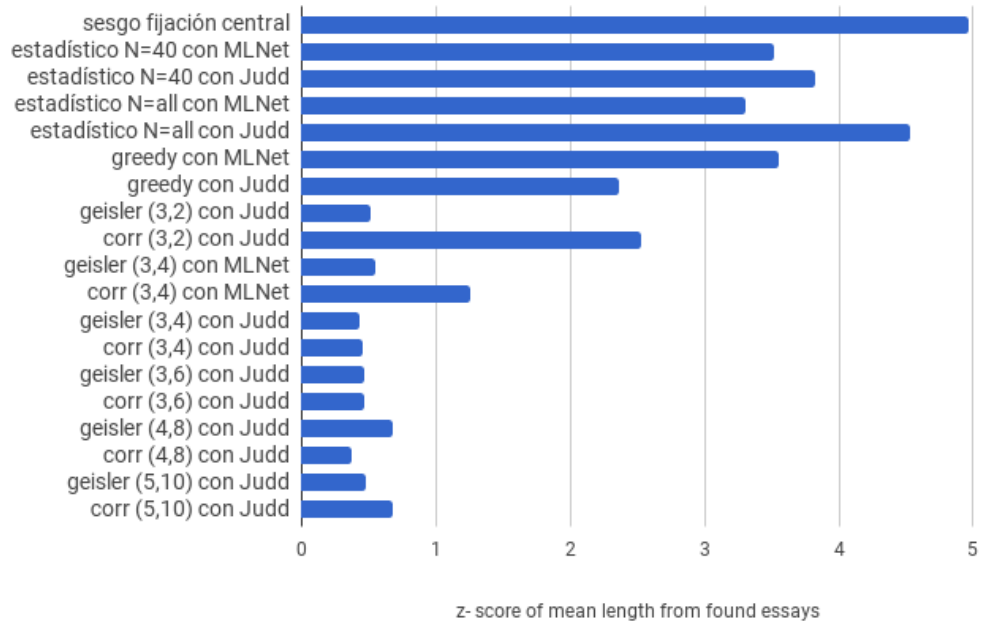


Figura 7.7: Gráfico de barra mostrando z^t_{length} para cada modelo dinámico t. En el gráfico, *geisler* refiere al modelo de Najemnik & Geisler original mientras que *corr* refiere a nuestra modificación. *Judd* alude a que fue utilizado el modelo de Judd extendido con nuestras features como modelo estático. Cuanto menor sea z^t_{length}, mejor.

Como era esperable, los modelos más simples no lograron predecir el comportamiento humano ni siquiera en la longitud de los scanpaths de los ensayos en los que se encontró el target.

7.3.2. Análisis de los modelos respecto de $z^{c,t}_{\substack{proportion\\found}}$

Esta métrica logró diferenciar muy bien entre los modelos bayesianos y los otros tres. Los modelos bayesianos modelan muy bien este aspecto de la búsqueda visual humana, a tal punto de que casi todos los *z-score* son menores a 1, o dicho de otro modo, el valor del modelo está a menos de un desvío estándar de la media de los humanos (ver figura 7.8).

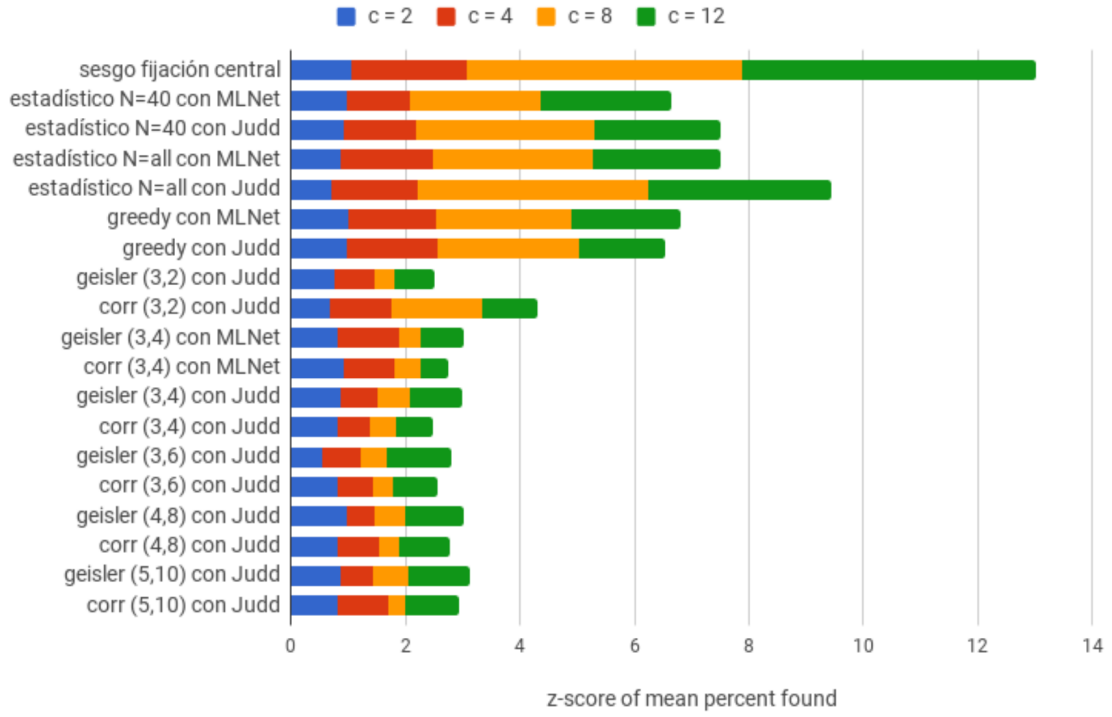


Figura 7.8: Gráfico de barra mostrando $z^{c,t}_{\substack{proportion\\found}}$ para cada modelo dinámico t en las cuatro categorías $c \in \{2, 4, 8, 12\}$. Cada color representa una de las categorías, y se puede ver la suma de esos *z-scores*. En el gráfico, *geisler* refiere al modelo de Najemnik & Geisler original mientras que *corr* refiere a nuestra modificación. *Judd* alude a que fue utilizado el modelo de Judd extendido con nuestras features como modelo estático. Cuanto menor sea $z^{c,t}_{\substack{proportion\\found}}$, mejor.

7.3.3. Análisis de los modelos respecto de $z^{c,t}_{lev}$

En la evaluación de esta métrica no graficaremos los valores obtenidos por los modelos no bayesianos, ya que nuevamente tienen performances mucho peores.

Primero focalizaremos la atención en la performance del modelo de Najemnik & Geisler original y el modificado para las constantes $(a, b) = (3, 4)$, que son las que seleccionamos originalmente. Ejecutamos los dos modelos utilizando nuestro modelo de Judd extendido y el modelo de MLNet. Como se puede ver, la performance del modelo con MLNet es notablemente peor, y esto justifica por qué en lo posterior solo experimentamos diferentes constantes sobre los modelos que usan el Judd extendido como *prior* (ver figura 7.9). Esto también es una buena señal de que efectivamente los modelos estáticos desarrollados en el capítulo anterior consiguieron contribuir en algún aspecto a la mejora del modelo.

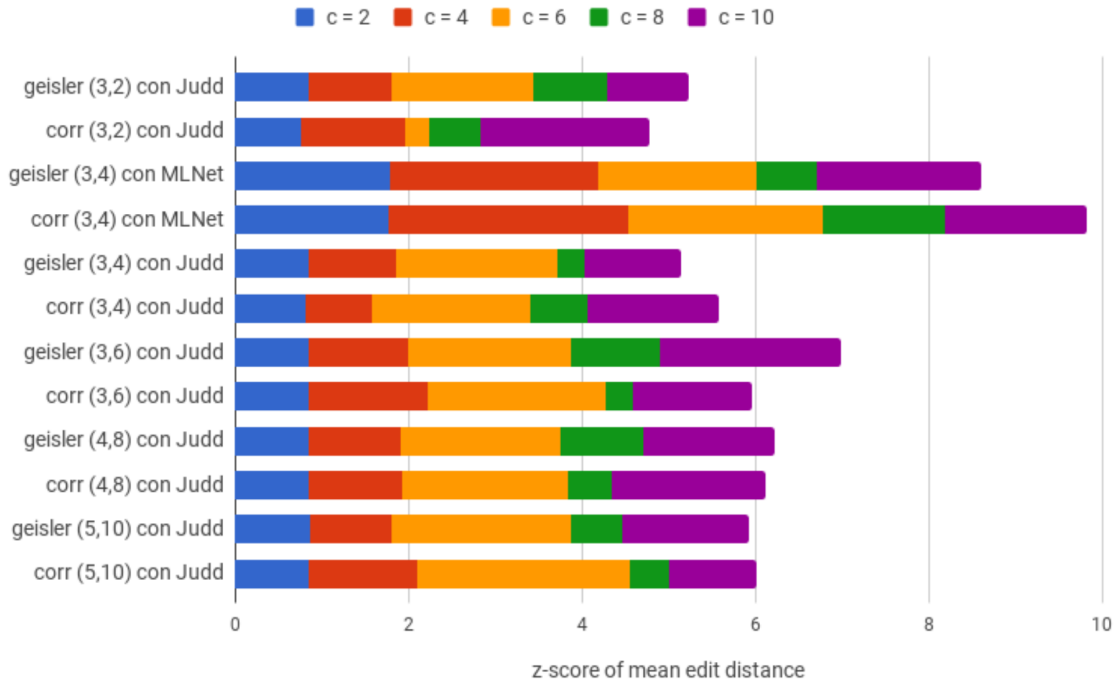


Figura 7.9: Gráfico de barra mostrando $z_{lev}^{c,t}$ para cada modelo dinámico t en las cinco categorías $c \in \{2, 4, 6, 8, 10\}$. Cada color representa una de las categorías, y se puede ver la suma de esos *z-scores*. En el gráfico, *geisler* refiere al modelo de Najemnik & Geisler original mientras que *corr* refiere a nuestra modificación. *Judd* alude a que fue utilizado el modelo de Judd extendido con nuestras features como modelo estático. Cuanto menor sea $z_{lev}^{c,t}$, mejor.

La performance de los modelos es muy similar para aquellos con baja variabilidad, es decir, para pares (a, b) más altos. Recordemos que estas constantes son factores que determinan la inversa de la varianza, así que cuanto menor sea $a \cdot d'_{ik(T)} + b$ mayor será la varianza. Notablemente, la performance parece mejorar un poco para modelos de más alta varianza. Hipotetizamos que este fenómeno puede producirse porque al bajar la variabilidad el modelo simula un poco peor la existencia de distractores y equivocaciones en el juicio de la mejor posición para obtener el target.

La performance del modelo original de Najemnik & Geisler y la del modelo modificado es similar para una misma elección de constantes: para algunas constantes obtiene mejor resultado uno, y para otros otro, lo que no permite concluir una superioridad entre los dos (ver figura 7.9). En este trabajo solo analizamos modificar el modelo de Najemnik & Geisler original alterando W realizando una combinación lineal entre el modelo original y la *cross-correlation*, pero podría intentarse otro tipo de combinaciones. Esto queda como trabajo futuro.

7.3.4. Conclusiones y selección de los mejores modelos

Primeramente podemos ver que el modelo *greedy* tuvo una performance mucho peor que los modelos bayesianos. Recordemos que el *greedy* toma para cada fijación el lugar más probable a encontrar el target, utilizando como mapa de probabilidad la saliencia y actualizándola en cada paso según las ubicaciones que son fijadas. Esto es evidencia de que considerar estratégicamente una fijación más para juzgar cuál es la mejor posición a observar representa una gran ventaja por sobre simplemente mirar la posición más probable en la fijación actual. Este resultado es consistente con la discusión de Najemnik & Geisler en el modelo original [NG05].

Yendo a la comparación entre modelos bayesianos, las primeras dos métricas muestran una muy buena performance de todos los modelos, a excepción del modelo de Geisler modificado con constantes $(a, b) = (3, 2)$. La métrica de edit distance arrojó como resultado que los mejores modelos fueron aquellos basados en Judd extendido con nuestras features y utilizando las constantes $(a, b) = (3, 2)$ o $(a, b) = (3, 4)$, aunque tampoco sean malas elecciones las otras elecciones de constantes. En base a todo esto, podemos concluir que el modelo de Geisler original con constantes $(3, 2)$ o $(3, 4)$ o el de Geisler modificado con constantes $(3, 4)$ son muy buenas elecciones ya que se ajustan bien a las tres métricas. Para obtener resultados más firmes se podrían analizar varias grillas de diferentes dimensiones, ya que dividir en diferentes regiones de interés (ROI) puede modificar la edit distance. En este punto decidimos acotar el análisis, aunque experimentamos con otras dos grillas (6×6 y 15×20, esta última siendo descartada por ser demasiado permisiva con todos los modelos) y no vimos grandes cambios en los resultados.

Capítulo 8

Discusión

Durante el presente trabajo hemos creado una tarea de búsqueda visual, recopilado datos de 28 observadores y analizado los diferentes aspectos esta tarea, así como creado modelos predictivos.

El primer objetivo del trabajo fue el de recopilar los datos para la tarea de búsqueda visual diseñada. Este objetivo fue cumplido y se publicarán los datos para que sea de acceso a toda la comunidad científica. Además de la tarea de búsqueda visual, este conjunto de datos se expandió con los datos obtenidos por la tarea *online* de reconocimiento de objetos. De este modo, además de tener los movimientos oculares y el reporte subjetivo de cada observador, se puede saber qué clase de objeto es el que se estaba buscando para así continuar el trabajo de desarrollar features semánticas para modelos estáticos. Queda como trabajo futuro recopilar datos de más observadores para posiblemente hacer desaparecer los *outliers* observados al analizar las métricas de comparación de scanpaths.

Yendo al aspecto subjetivo de la tarea, dimos un paso adelante en la predicción de las respuestas de los observadores. Vimos que las personas son conscientes cuando encuentran el target en un ensayo, y lo expresan colocando círculos más precisos aunque se les da poco tiempo. Como consecuencia de esto, el tamaño de los círculos de respuesta es un buen indicador para separar los ensayos con targets encontrados de aquellos donde el target no fue encontrado. Sin embargo, vimos que los observadores son más propensos a colocar círculos de respuesta amplios cuando en sus respuestas inmediatamente anteriores también colocaron círculos de respuesta grandes. Es por esto que consideramos que sería difícil realizar un análisis al detalle de los factores que afectan el tamaño del círculo de respuesta, pero es un buen indicador para saber si el target fue encontrado. Respecto a la predicción del centro del círculo de respuesta, logramos predecir las regiones más probables de una imagen para que se ubiquen de los centros de los círculos de respuesta de los observadores, no utilizando

en el entrenamiento ningún dato sobre las personas ni sobre las imágenes sobre las que se evaluó. Para que este resultado sea más contundente sería interesante tomar datos sobre más observadores ya que tuvimos 7 observadores en el conjunto de *testing*. Además, por la naturaleza del algoritmo utilizado para la predicción es difícil interpretar intuitivamente qué regiones de una imagen son más probables a ser seleccionadas como centro del círculo de respuesta, por lo que no consideramos que el análisis esté acabado con este resultado.

En segundo lugar nos dispusimos a crear diferentes modelos predictivos de las ubicaciones de las fijaciones. Los primeros modelos analizados solo se abocaron a predecir las regiones más probables a ser fijadas por los observadores sin tener en cuenta el aspecto temporal. Esto se llevó a cabo con distintos mapas de saliencia: dos de ellos son modelos del estado del arte y el tercero es una expansión de un modelo conocido basado en *machine learning* utilizando features específicas para la tarea. Con este modelo conseguimos mejorar marginalmente los modelos del estado del arte, obteniendo el mayor índice de mejora en la tercera fijación. Esto es especialmente relevante pues es el mapa que utilizaremos como base para modelos más sofisticados, ya que como vimos a lo largo del trabajo las dos fijaciones anteriores se encuentran influenciadas por las indicaciones propias de la tarea o por un sesgo de los sujetos de observar el centro de la pantalla. Respecto a la expansión de features queda mucho trabajo a futuro, especialmente en la creación de mapas que modelen el aspecto semántico del target que se busca en el ensayo. Algunas de las tareas son: entrenar un detector del horizonte para imágenes de interiores (se conocen modelos solo entrenados para exteriores), entrenar un modelo de vector de palabras adaptado específicamente a nuestro caso de uso o traducir todo el experimento al inglés, donde existe más investigación sobre creación de vectores de palabras.

Como los modelos anteriormente mencionados no son capaces de predecir la secuencia de las fijaciones, debimos analizar otros modelos que permitan predecir esto. Para ello utilizamos las ideas de Najemnik y Geisler, que modelaron la búsqueda visual como un protocolo bayesiano, y la combinamos con un mapa de saliencia que obtuvimos de las investigaciones discutidas en el párrafo anterior. Pese a la amplia repercusión de este trabajo, no es de nuestro conocimiento que se hayan replicado las ideas en tareas de búsqueda visual en escenas naturales. Es por esto que incluimos como otro objetivo cumplido del trabajo haber podido replicar este trabajo para una tarea de búsqueda en escenas naturales. Además, desarrollamos más métricas de comparación que las utilizadas por Najemnik y Geisler, que solo toman en cuenta la cantidad de fijaciones requerida hasta encontrar el target. Los mejores modelos logran modelar razonablemente bien el comportamiento humano, encontrándose como máximo a 1,6 desvíos estándar de la media humana.

Dentro del modelo de Najemnik y Geisler, la visibilidad de una ubicación dado que estamos realizando una fijación en otra es un factor clave. En este trabajo aproximamos el mapa de visibilidad como una gaussiana de forma tal que los parámetros reflejaran lo más fielmente posible los estudios del área. Sin embargo, queda como trabajo futuro utilizar modelos más sofisticados, ya que la gaussiana es simplemente una aproximación al caso promedio de la visibilidad en todas las ubicaciones y no tiene en cuenta los diferentes niveles de contraste entre el target y la imagen de fondo. Estos modelos son computacionalmente costosos y eso impidió su inclusión en este trabajo (en especial el de Bradley et al. [BAG14], con el que estuvimos realizando pruebas).

Como conclusión general del trabajo podemos decir que dimos un paso adelante en la predicción de las regiones de las fijaciones y la secuencia más probable de las fijaciones en los humanos para esta tarea de búsqueda visual, y por ende en la descripción de los modelos de búsqueda visual a nivel algorítmico. Por supuesto, aún resta mucha investigación para poder entender por completo este mecanismo cognitivo, y aquí hemos presentado algunos caminos posibles para continuar avanzando. Además, consideramos que la creación de un conjunto de datos abierto para una tarea de búsqueda visual podrá ayudar a que otros investigadores además de nosotros puedan continuar con esta labor.

Bibliografía

[BAG14] Chris Bradley, Jared Abrams, and Wilson S Geisler. Retina-v1 model of detectability across the visual field. *Journal of vision*, 14(12):22–22, 2014.

[BCJL15] Ali Borji, Ming-Ming Cheng, Huaizu Jiang, and Jia Li. Salient object detection: A benchmark. *IEEE Transactions on Image Processing*, 24(12):5706–5722, 2015.

[BGJM16] Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. Enriching word vectors with subword information. *arXiv preprint arXiv:1607.04606*, 2016.

[BI15] Ali Borji and Laurent Itti. Cat2000: A large scale fixation dataset for boosting saliency research. *arXiv preprint arXiv:1505.03581*, 2015.

[BJB+] Zoya Bylinskii, Tilke Judd, Ali Borji, Laurent Itti, Frédo Durand, Aude Oliva, and Antonio Torralba. Mit saliency benchmark.

[BS97] Stephan A Brandt and Lawrence W Stark. Spontaneous eye movements during visual imagery reflect the content of the visual scene. *Journal of cognitive neuroscience*, 9(1):27–38, 1997.

[CBN05] Hannah Faye Chua, Julie E Boland, and Richard E Nisbett. Cultural variation in eye movements during scene perception. *Proceedings of the National Academy of Sciences of the United States of America*, 102(35):12629–12633, 2005.

[CBSC16a] Marcella Cornia, Lorenzo Baraldi, Giuseppe Serra, and Rita Cucchiara. A Deep Multi-Level Network for Saliency Prediction. 2016.

[CBSC16b] Marcella Cornia, Lorenzo Baraldi, Giuseppe Serra, and Rita Cucchiara. Predicting human eye fixations via an lstm-based saliency attentive model. *arXiv preprint arXiv:1611.09571*, 2016.

[CGDLB08] Myriam Chanceaux, Anne Guérin-Dugué, Benoît Lemaire, and Thierry Baccino. Towards a model of information seeking by integrating visual, semantic and memory maps. *ICVW*, 2008:65–78, 2008.

[CHEK08] Moran Cerf, Jonathan Harel, Wolfgang Einhäuser, and Christof Koch. Predicting human gaze using low-level saliency combined with face detection. In *Advances in neural information processing systems*, pages 241–248, 2008.

[CMH09] Monica S Castelhano, Michael L Mack, and John M Henderson. Viewing task influences eye movement control during active scene perception. *Journal of vision*, 9(3):6–6, 2009.

[DD95] Robert Desimone and John Duncan. Neural mechanisms of selective visual attention. *Annual review of neuroscience*, 18(1):193–222, 1995.

[Dow07] J. Dowling. Retina. *Scholarpedia*, 2(12):3487, 2007. revision #91715.

[FW12] J. Findlay and R. Walker. Human saccadic eye movements. *Scholarpedia*, 7(7):5095, 2012. revision #122018.

[GKT08] Thomas L Griffiths, Charles Kemp, and Joshua B Tenenbaum. Bayesian models of cognition. 2008.

[HBCM07] John M Henderson, James R Brockmole, Monica S Castelhano, and Michael Mack. Visual saliency does not account for eye movements during visual search in real-world scenes. *Eye movements: A window on mind and brain*, pages 537–562, 2007.

[Hoi07] Derek Hoiem. *Seeing the world behind the image: spatial layout for three-dimensional scene understanding.* Carnegie Mellon University, 2007.

[IK00] Laurent Itti and Christof Koch. A saliency-based search mechanism for overt and covert shifts of visual attention. *Vision research*, 40(10):1489–1506, 2000.

[Itt07] L. Itti. Visual salience. *Scholarpedia*, 2(9):3327, 2007. revision #72776.

[Jac86] Arthur M Jacobs. Eye-movement control in visual search: how direct is visual span control? *Attention, Perception, & Psychophysics*, 39(1):47–58, 1986.

[JDT12] Tilke Judd, Frédo Durand, and Antonio Torralba. A benchmark of computational models of saliency to predict human fixations. 2012.

[JEDT09] Tilke Judd, Krista Ehinger, Frédo Durand, and Antonio Torralba. Learning to predict where humans look. pages 2106–2113, 2009.

[JH02] Sheree Josephson and Michael E Holmes. Attention to repeated images on the world-wide web: Another look at scanpath theory. *Behavior Research Methods, Instruments, & Computers*, 34(4):539–548, 2002.

[KRK10] Andreas Kotowicz, Ueli Rutishauser, and Christof Koch. Time course of target recognition in visual search. *Frontiers in Human Neuroscience*, 4, 2010.

[KTZC09] Christopher Kanan, Mathew H Tong, Lingyun Zhang, and Garrison W Cottrell. Sun: Top-down saliency using natural statistics. *Visual cognition*, 17(6-7):979–1003, 2009.

[LMB13] Olivier Le Meur and Thierry Baccino. Methods for comparing scanpaths and saliency maps: strengths and weaknesses. *Behavior research methods*, 45(1):251–266, 2013.

[MSM15] Florent Meyniel, Mariano Sigman, and Zachary F Mainen. Confidence as bayesian probability: from neural origins to behavior. *Neuron*, 88(1):78–92, 2015.

[NG05] Jiri Najemnik and Wilson S Geisler. Optimal eye movement strategies in visual search. *Nature*, 434(7031):387, 2005.

[NG08] Jiri Najemnik and Wilson S Geisler. Eye movement statistics in humans are consistent with an optimal search strategy. *Journal of Vision*, 8(3):4–4, 2008.

[OHVE07] EAB Over, ITC Hooge, BNS Vlaskamp, and CJ Erkelens. Coarse-to-fine eye movement strategy in visual search. *Vision Research*, 47(17):2272–2280, 2007.

[PAF01] D Purves, GJ Augustine, and D Fitzpatrick. *Neuroscience*. Sinauer Associates, 2 edition, 2001.

[PS00] Claudio M. Privitera and Lawrence W. Stark. Algorithms for defining visual regions-of-interest: Comparison with eye fixations. *IEEE Transactions on pattern analysis and machine intelligence*, 22(9):970–982, 2000.

[QT09] Ariadna Quattoni and Antonio Torralba. Recognizing indoor scenes. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 413–420. IEEE, 2009.

[RC07] K. Rayner and M. Castelhano. Eye movements. *Scholarpedia*, 2(10):3649, 2007. revision #126973.

[RCY09] Keith Rayner, Monica S Castelhano, and Jinmian Yang. Eye movements when looking at unusual/weird scenes: Are there cultural differences? *Journal of Experimental Psychology: Learning, Memory, and Cognition*, 35(1):254, 2009.

[Rol15] Martin Rolfs. Attention in active vision: A perspective on perceptual continuity across saccades. *Perception*, 44(8-9):900–919, 2015.

[Ros99] Ruth Rosenholtz. A simple saliency model predicts a number of motion popout phenomena. *Vision research*, 39(19):3157–3163, 1999.

[RTMF08] Bryan C Russell, Antonio Torralba, Kevin P Murphy, and William T Freeman. Labelme: a database and web-based tool for image annotation. *International journal of computer vision*, 77(1):157–173, 2008.

[SF95] Eero P Simoncelli and William T Freeman. The steerable pyramid: A flexible architecture for multi-scale derivative computation. In *Image Processing, 1995. Proceedings., International Conference on*, volume 3, pages 444–447. IEEE, 1995.

[SKT+08] Josef Sivic, Biliana Kaneva, Antonio Torralba, Shai Avidan, and William T Freeman. Creating and exploring a large photorealistic virtual space. In *Computer Vision and Pattern Recognition Workshops, 2008. CVPRW'08. IEEE Computer Society Conference on*, pages 1–8. IEEE, 2008.

[Tat07] Benjamin W Tatler. The central fixation bias in scene viewing: Selecting an optimal viewing position independently of motor biases and image feature distributions. *Journal of vision*, 7(14):4–4, 2007.

[TBG05] Benjamin W Tatler, Roland J Baddeley, and Iain D Gilchrist. Visual correlates of fixation selection: Effects of scale and time. *Vision research*, 45(5):643–659, 2005.

[TGK06] Joshua B Tenenbaum, Thomas L Griffiths, and Charles Kemp. Theory-based bayesian models of inductive learning and reasoning. *Trends in cognitive sciences*, 10(7):309–318, 2006.

[TOCH06] Antonio Torralba, Aude Oliva, Monica S Castelhano, and John M Henderson. Contextual guidance of eye movements and attention

in real-world scenes: the role of global features in object search. *Psychological review*, 113(4):766, 2006.

[TS01] Antonio Torralba and Pawan Sinha. Statistical context priming for object detection. In *Computer Vision, 2001. ICCV 2001. Proceedings. Eighth IEEE International Conference on*, volume 1, pages 763–770. IEEE, 2001.

[TWK+10] Benjamin W Tatler, Nicholas J Wade, Hoi Kwan, John M Findlay, and Boris M Velichkovsky. Yarbus, eye movements, and vision. *i-Perception*, 1(1):7–27, 2010.

[VTBE15] Oriol Vinyals, Alexander Toshev, Samy Bengio, and Dumitru Erhan. Show and tell: A neural image caption generator. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3156–3164, 2015.

[WH08] J. Wolfe and T. S. Horowitz. Visual search. *Scholarpedia*, 3(7):3325, 2008. revision #145401.

[Wol94] Jeremy M Wolfe. Guided search 2.0 a revised model of visual search. *Psychonomic bulletin & review*, 1(2):202–238, 1994.

[Yar67] Alfred L Yarbus. Eye movements during perception of complex objects. In *Eye movements and vision*, pages 171–211. Springer, 1967.

[ZTM+08] Lingyun Zhang, Matthew H Tong, Tim K Marks, Honghao Shan, and Garrison W Cottrell. Sun: A bayesian framework for saliency using natural statistics. *Journal of vision*, 8(7):32–32, 2008.