

Study and Benchmarking on Image Matting state-of-the-art methods

Mauricio Aravena Cifuentes

Estudiante de Ingeniería

Cívil Electrónica

Universidad Técnica Federico Santa María

Valparaíso, Chile

maurio.aravena@sansano.usm.cl

Diego Badillo San Juan

Estudiante de Ingeniería

Civil Electrónica

Universidad Técnica Federico Santa María

Valparaíso, Chile

diego.badillo@sansano.usm.cl

Resumen—En este informe estudiamos el problema de “Alpha Matting”, que consiste en la creación de una máscara no binaria para la segmentación de una imagen en un fondo (background), primer plano (foreground) y matices intermedios de transparencia. En particular, se evaluará el trabajo planteado en dos artículos. El primero de ellos, un algoritmo basado en afinidad (*affinity-based*) para la creación de canales alfa de segmentación a partir de la imagen y un trimapa: [1] (Y. Aksoy, T. O. Aydin, and M. Pollefeys, “Designing effective inter-pixel information flow for natural image matting”). El segundo ocupa un enfoque basado en Deep Learning para poder obtener el canal alfa de segmentación a partir de una imagen, que contenga un primer plano y un fondo, y una imagen que contenga un fondo lo suficientemente similar al de la imagen: [2] (S. Sengupta, V. Jayaram, B. Curless, S. Seitz, and I. Kemelmacher-Shlizerman, “Background matting: The world is your green screen”)

Evaluaremos los resultados de ambos artículos con error cuadrático medio (MSE) y error absoluto medio (MAE) usando un dataset creado a partir de imágenes de personas con sus respectivos canales alfa, obtenido de Adobe® [3], y otro creado en el programa de simulación 3D Blender. Ambos datasets serán evaluados por separado para poder comparar el desempeño de ambos en distintos entornos.

I. INTRODUCCIÓN

Extraer el fondo (background) de una imagen que contenga un primer plano de interés a segmentar, es una tarea útil y esencial que puede usarse en varias aplicaciones. Obtener una máscara binaria de un objeto o persona de interés en una imagen puede servirnos para saber a grandes rasgos su ubicación espacial, su forma, su área, etc.

En la realidad, es poco común que un objeto de interés tenga un límite bien definido con el fondo, al capturarlo en una imagen o vídeo. También puede darse el caso de que, incluso dentro de los contornos del objeto, haya información del fondo debido a la transparencia que pueda presentar el primer plano (objeto de interés).

Alpha Matting se refiere al problema de separar el fondo y el primer plano considerando distintos niveles de opacidad (256 para un canal alfa de una imagen codificada en 8 bits). Para esto, cada píxel de la máscara a crear representará una combinación convexa ((1), en donde $0 < \alpha < 1$) entre lo que se considera fondo (Background) lo que se considera primer plano (Foreground).

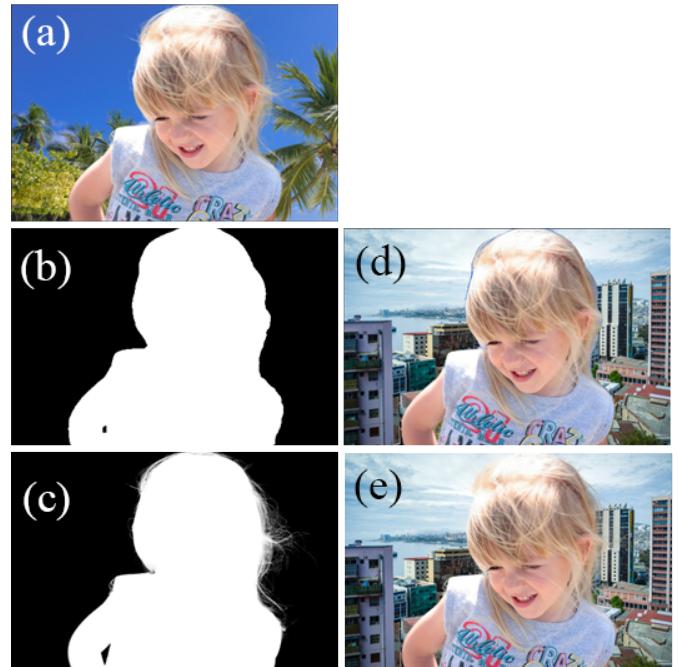


Figura 1. Extracción del fondo para la imagen (a) usando una máscara binaria (b) y una máscara con 256 niveles (c) para obtener las composiciones (d) y (e) respectivamente.

$$I = \alpha F + (1 - \alpha)B \quad (1)$$

En la figura 1 se muestra cómo funcionaría una segmentación binaria de una imagen para el fin de colocar al sujeto en otro fondo, versus a cómo funcionaría usar un canal alfa con más de dos niveles de discretización.

Este problema no es fácil de solucionar, al menos con buenos estándares de calidad, cuando el fondo no es monótono o contiene colores, saturaciones y/o luminosidades similares a las del primer plano. Lo más común, y que ha sido usado por varias décadas en el cine, por ejemplo, es usar una pantalla verde en el fondo para poder obtener la máscara del primer plano. Sin embargo no siempre se dispondrá de una pantalla verde, y una buena iluminación para segmentar la imagen con este método, por lo que el estudio del Alpha Matting resulta interesante para diversas aplicaciones en la vida cotidiana.

II. TRABAJO RELACIONADO

Natural image matting se refiere al proceso de obtener una máscara alfa que separe el primer plano (foreground) y fondo (background) considerando los distintos niveles de opacidad. Existen dos métodos generales “tradicionales” propuestos para abordar este problema: métodos basados en afinidad (affinity-based) [4] y métodos basados en muestreo (sampling-based) [5], [6]. También existen métodos híbridos entre ambos métodos anteriormente mencionados. Esto, sin mencionar métodos que aborden un enfoque basado en aprendizaje (deep learning), como es uno de los métodos que se tratará en este informe [7]. Los métodos de afinidad, son también conocidos como métodos de propagación, puesto que “propagan” información de opacidad desde regiones con opacidad conocida. Estos se basan en cercanías espaciales o de colores entre los colores de los píxeles. Para esto, este tipo de algoritmos debe recibir como entrada un *trimapa*, que consiste en una máscara de pre-segmentación de la imagen que contiene tres valores (ver figura 2). La región **F** consiste en píxeles que inequívocamente pertenecen al *foreground*, mientras que **B** consiste en píxeles que inequívocamente pertenecen al *background*. La región **U** es aquella que contendrá distintos niveles de opacidad en el canal alfa final. Es importante notar que el resultado del algoritmo de afinidad dependerá considerablemente del trimapa inicial entregado a la entrada. En este informe, estudiaremos y evaluaremos los resultados del método basado en afinidad denominado *Information Flow Matting* [1].

En cuanto a métodos basados en aprendizaje, estos podrían o no necesitar un trimapa a la entrada. [8], [9], [10], [2].

En [2], cuyos resultados evaluaremos en este informe, ocupan aprendizaje de máquinas para predecir el canal alfa que describa el *foreground* y *background* y sus respectivos niveles de opacidad. El enfoque utilizado en este artículo busca lograr encontrar el canal alfa de segmentación sin necesidad de recibir como entrada un trimapa. Esto se reemplaza por la necesidad de ingresar como entrada el fondo de la imagen, que no necesita estar exactamente sincronizado con el fondo de la imagen a segmentar, como se verá en más detalle próximamente.

II-A. Information Flow Matting [1]

En esta sección explicaremos el enfoque del algoritmo a grandes rasgos, para mayor detalle, pedimos al lector referirse al artículo [1].

Hemos escogido este algoritmo como ejemplo de método tradicional para encontrar el canal alfa puesto que es el preferido que ha sido implementado en la biblioteca *OpenCV* para problemas de *alpha matting*.

A grandes rasgos, este algoritmo se basa en la minimización de la energía

$$E_1 = E_{CM} + \sigma_{KU} E_{KU} + \sigma_{UU} E_{UU} + \sigma_L E_L + \lambda E_T \quad (2)$$

definida como la suma ponderada de las energías E_{CM} , E_{KU} , E_{UU} , E_L y E_T . No entraremos en detalle a la definición matemática de cada una de estas energías, pero sí a qué se refieren.

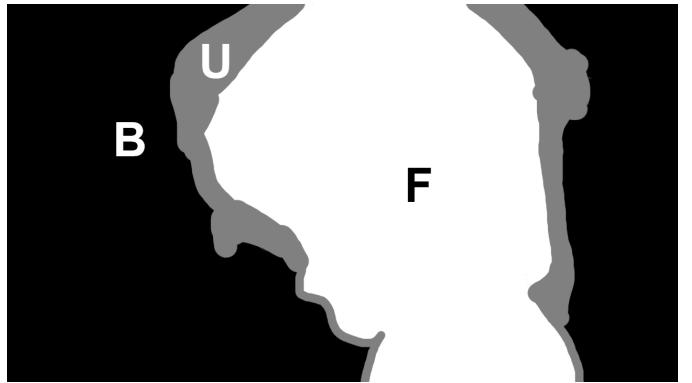


Figura 2. Trimapa. U: Unknown. B: Background. F: Foreground.

II-A1. Color-mixture information flow: Este algoritmo de afinidad consiste en buscar $K_{CM} = 20$ píxeles q en **toda la imagen**, por cada píxel p en la región \mathcal{U} , cuyos valores de color sean lo más cercanos al píxel en cuestión de la región desconocida \mathcal{U} . Es decir, se buscan 20 *K-nearest-neighbors* en color (\mathcal{N}_p^{CM}) para cada píxel p en \mathcal{U} , sin considerar la distancia espacial. El color del píxel p en la región \mathcal{U} es representado como una suma ponderada (pesos $w_{p,q}^{CM}$) entre los colores de los 20 píxeles q encontrados.

La función de energía E_{CM} queda representada como:

$$E_{CM} = \sum_{p \in \mathcal{U}} \left(\alpha_p - \sum_{q \in \mathcal{N}_p^{CM}} w_{p,q}^{CM} \alpha_q \right)^2 \quad (3)$$

en donde α_p representa la opacidad del píxel p en \mathcal{U} .

II-A2. \mathcal{K} -to- \mathcal{U} information flow: Análogamente al caso anterior, ahora se buscarán $K_{KU} = 7$ píxeles q de colores similares para cada píxel p en \mathcal{U} , pero ahora su búsqueda no será en toda la imagen, sino solo en la zona conocida (*known*), es decir, en **B** y **F**. Además, se considerará la distancia espacial también, a diferencia del algoritmo anterior.

$$E_{KU} = \sum_{p \in \mathcal{U}} \left(\alpha_p - \sum_{q \in \mathcal{N}_p^{\mathcal{F}}} w_{p,q}^{\mathcal{F}} \alpha_q - \sum_{q \in \mathcal{N}_p^{\mathcal{B}}} w_{p,q}^{\mathcal{B}} \alpha_q \right)^2 \quad (4)$$

La suma de todos los pesos $w_{p,q}^{\mathcal{B}}$ y $w_{p,q}^{\mathcal{F}}$ es la unidad. Al considerar que $\alpha_q = 1$ para $q \in \mathcal{F}$ y $\alpha_q = 0$ para $q \in \mathcal{B}$, podemos reescribir (4) como

$$E_{KU} = \sum_{p \in \mathcal{U}} \left(\alpha_p - \sum_{q \in \mathcal{N}_p^{\mathcal{F}}} w_{p,q}^{\mathcal{F}} \right)^2 \quad (5)$$

Es decir, solo interesarán los píxeles del foreground.

II-A3. Intra- \mathcal{U} information flow: Para este caso se busca afinidad entre píxeles dentro de la zona “desconocida” (*unknown*) solamente. Esto es, para cada píxel p en \mathcal{U} buscamos $K_{\mathcal{U}} = 5$ píxeles q en la misma zona desconocida que contengan la mayor cercanía, tanto de colores como espacial.

$$E_{UU} = \sum_{p \in \mathcal{U}} \sum_{q \in \mathcal{N}_p^{\mathcal{U}}} w_{p,q}^{\mathcal{U}} (\alpha_p - \alpha_q)^2 \quad (6)$$

II-A4. Local information flow: Para este paso, se considera para cada píxel p en \mathcal{U} , sus píxeles inmediatamente adyacentes usando conectividad-8. La función de energía para este caso es:

$$E_L = \sum_{p \in \mathcal{U}} \sum_{q \in \mathcal{N}_p^L} w_{p,q}^L (\alpha_p - \alpha_q)^2 \quad (7)$$

II-A5. Energía de corrección: En (2) se agrega una energía adicional E_T . Su propósito es mantener los píxeles de *foreground* y *background* como tales. Es decir, si algo ha sido marcado como *foreground* o *background* en el trimapa, se espera que en el canal alfa final esta información no haya cambiado mucho. Por esto, a esta energía se le asigna el mayor peso $\lambda = 100$. Los otros pesos en (2) son $\sigma_{KU} = 0,05$, $\sigma_{UU} = 0,01$ y $\sigma_L = 1$.

$$E_T = \sum_{p \in \mathcal{F}} (\alpha_p - 1)^2 + \sum_{p \in \mathcal{B}} (\alpha_p - 0)^2 \quad (8)$$

II-B. Deep Learning: The World is Your Green Screen [2]

Como se mencionó anteriormente, este método no necesita un *trimapa* a la entrada, sino una imagen adicional que contenga solo el *background*. Analizando el problema general, ecuación 1, la dificultad radica en que generalmente sólo se conoce la imagen - I , lo que no permite obtener información acerca de los valores de *foreground*, *background* o el valor de la constante α . El enfoque propuesto, se basa en considerar el caso, que para cierta imagen I , se conoce B - *background*. Si se quisiera determinar el valor de la constante α y por conseciente, el valor del *foreground* - F , se podría plantear el siguiente sistema de ecuaciones:

$$I_r = F_r \alpha + (1 - \alpha)B_r$$

$$I_g = F_g \alpha + (1 - \alpha)B_g$$

$$I_b = F_b \alpha + (1 - \alpha)B_b$$

Se asumen una imagen de tres canales, se tiene un sistema de ecuaciones de cuatro incógnitas (Los tres canales de F y el valor de α), pero tres ecuaciones, por lo que queda subdeterminado. Pese a esto, si se considera la relación que

existe entre el *background* y *foreground*, la diferencia entre estos permitiría ayudar (en algunos casos, donde la diferencia entre los colores del B y F lo permitan) determinar las incógnitas del problema. Es necesario recalcar, que en general sólo se tendrá acceso a la imagen I y la obtención de B será en conjunto parte del problema, sin embargo, los autores proponen dos soluciones para atacar la problemática.

II-B1. Reducción del alcance de la problemática: En vez de atacar el problema general, visto desde todas las aristas posibles, concentrarse únicamente en una aplicación concreta:

La generación de un canal α de una persona. Esta reducción permite considerar, que al momento de capturar/obtener la imagen de un sujeto en un *background* específico, se puede tomar una segunda captura, pidiendo al sujeto que se retire de la escena, para obtener el “*background*” de la escena. Note que si bien, se habla de *background*, es necesario recalcar con comillas que este sería una aproximación del mismo, dado que este no puede ser capturado al mismo instante y posición exacta que el sujeto, diversos factores del mismo podrán cambiar en el intervalo entre que el sujeto abandona la escena y se toma la captura de la escena. Si en la escena hay movimiento, por ejemplo de hojas en el viento, personas que se cruzan, el movimiento del fotógrafo, etc. Por lo que podemos considerar que esta captura de la escena sin el sujeto, es en realidad una **aproximación** del *background* original ($B \approx \tilde{B}$). Redefiniendo la ecuación 1, para este nuevo fondo \tilde{B} :

$$I \approx \alpha F + (1 - \alpha)\tilde{B} \quad (9)$$

Si bien, esta idea de la aproximación del *background* trae consigo sus propios problemas, estos se verán en el siguiente punto, por el momento considere que pueden ser solventados. La importancia de esta reducción del problema, viene de la mano de los desarrollos presentes en el estado del arte, técnicas de *soft-segmentation* [11], las cuales pueden ser implementadas mediante redes de *deep learning* [12] entrenadas para segmentar personas en una imagen, por lo que se puede aprovechar el trabajo ya implementado, para poder generar una máscara binaria de la persona, la cual servirá como guía, siendo una aproximación a lo que sería el canal α de la imagen, la que se denominará S , de esta

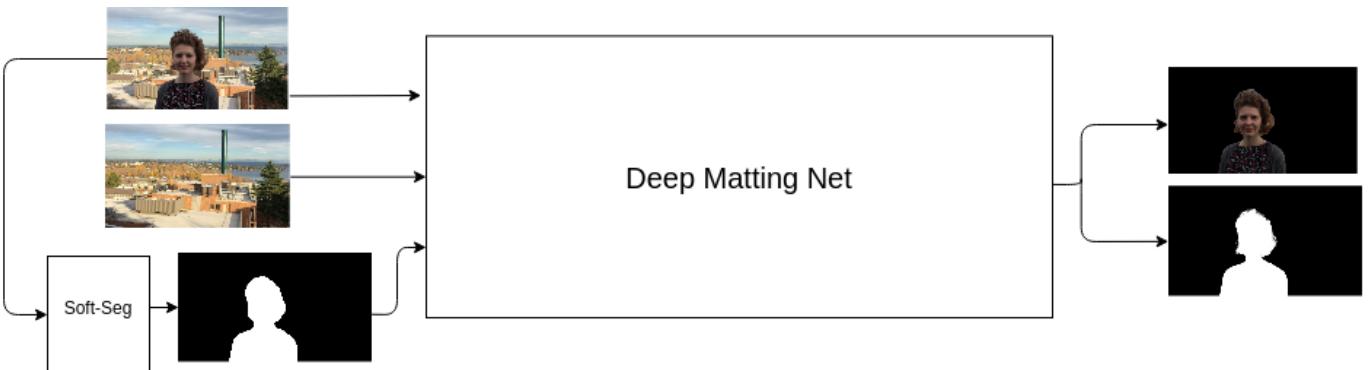


Figura 3. Diagrama de bloques para la implementación de [2]. **Izquierda:** Entradas del sistema, la imagen compuesta se hace pasar por una etapa de *soft segmentation*, para obtener el canal α a priori. **Derecha:** Se tienen las salidas del algoritmo, obteniéndose el canal α procesado y una imagen de *foreground*

manera pudiendo explotar la relación entre las diferencias de *foreground* y *background*.

II-B2. Uso de deep learning para la resolución de 9:

Ahora el problema queda definido por la expresión en la ecuación 9, y poseemos \tilde{B} . En primera instancia es de interés hacer que este *background* sea lo más parecido al presente en la imagen original. Dado que habrán diferencias que serán inherentes a los cambios en la escena, por el momento sólo se realizará la corrección de los cambios asociados a la posición de la cámara al momento de la captura. Para esto se utiliza el acercamiento de utilizar descriptores locales, de esta forma utilizando las características presentes tanto en la imagen original I y en \tilde{B} , hallar la homografía que realice la corrección deseada, en la implementación utilizada en este documento, se utilizó AKAZE [13]. A partir de esta operación se obtendrá una imagen de \tilde{B} la cual es lo más cercana, en términos de orientación y posición, con respecto al *background* original. El siguiente paso de la solución propuesta es utilizar los resultados obtenidos hasta el momento S , \tilde{B} y la imagen de entrada I ¹ a una red *Deep-Matting*, la cual podrá a partir de las entradas, obtener un F y un canal α . Se define la red como:

$$(F, \alpha) = G(X; \theta) \quad (10)$$

Dada la complejidad de obtener imágenes cuyos canales alfa sean conocidos, los autores optan por utilizar el dataset de Adobe® [3]. Este contiene, la imagen de canal α , F y un trimapa para cada imagen del dataset, tanto para entrenamiento como para pruebas. Es posible, entonces, la expansión del dataset utilizando la información contenida tanto en el *foreground* y el canal α , añadiendo un *background* a la composición. De esta forma, se podría tener un dataset lo suficientemente amplio, para poder realizar el entrenamiento. Si bien, esta extensión del dataset, simplifica la tarea de diseño y entrenamiento de la red G , durante el desarrollo de la primer iteración de la red G , los autores se dieron cuenta que los enfoques tradicionales [14] generaban problemas en los resultados. En los casos donde donde \tilde{B} presenta colores similares a F , la red tenía a confiar demasiado en \tilde{B} generando agujeros en el resultado final. Esto se traducía a que los resultados, sobre imágenes compuestas reales, tendían a ser poco precisos. Este *overfitting* de la red hacia el dataset, motivó a cambiar el paradigma de la implementación.

Se proponen dos soluciones a esta situación. La primera, es la implementación de un bloque *Context Switching Block*, figura 4, el cual combina de forma más efectiva la información proveniente de las entradas: I , \tilde{B} , S . Aplicando convolución 1x1, *BatchNorm* y un bloque selector *ReLU*, de I sobre cada una de las otras entradas, generando características de 64 canales para cada par. Estas características luego son combinadas con la imagen I , mediante convolución 1x1, *BatchNorm* y *ReLU* (bloque *comb* en la figura 4). Este bloque permite, utilizando las composiciones sintéticas obtenidas del dataset de Adobe, llegar a una generalización de la información, que

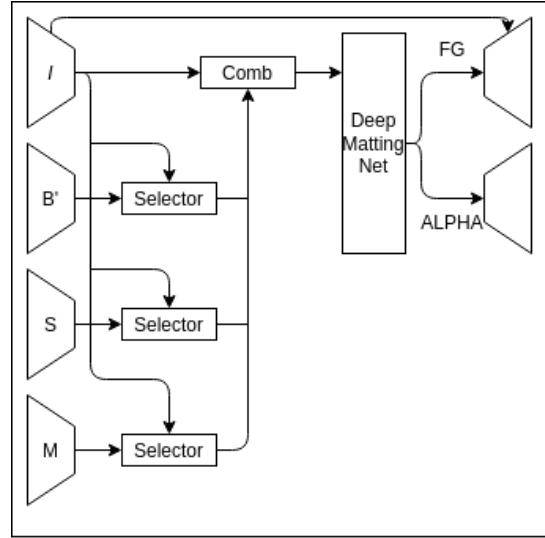


Figura 4. Bloque *Context Switch* propuesto

permite añadir robustez al sistema en la replicación de los resultados sobre composiciones reales. Si bien, esta bloque mejora los resultados obtenidos, no permite cerrar del todo la brecha entre el entrenamiento y los casos reales, a este nivel los errores ahora provienen por sobre todo de los detalles finos de la imagen, posición de los brazos, dedos y cabellos no son correctamente segmentados, el problema de la similitud de colores en el *background* y *foreground* aún persiste. Se propone la siguiente implementación y entrenamiento de la red G :

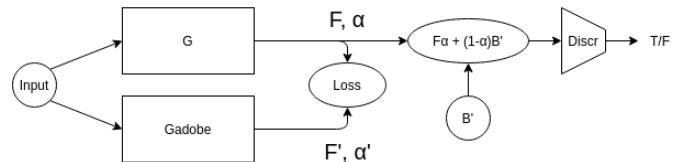


Figura 5. Esquema de entrenamiento de la red G

El esquema consiste, en entrenar la red G mediante aprendizaje en información real (*background* + *foreground*), sin etiquetado con *self-supervision*. La red puede ser mejorada, considerando que defectos en la determinación de las incógnitas *foreground* y canal α llevaran a realizar un proceso de composición, ecuación 1, en donde se podrá distinguir que el resultado de la “suma de dos imágenes”. Para explotar esta idea, se añade un bloque, *discriminador* (*discr* en la figura 5), el cual es entrenado para poder detectar *fake-composites*, de esta forma, la red G puede ser refinada, al comparar el resultado que saldría de realizar una composición con las salidas de G , sobre un *background* dado \tilde{B} , verificando que puedan engañar al *discriminador*. De esta forma entrenando G en un esquema de *adversarial loss*. Sin embargo, note que esto podría generar un caso dónde la red G se estancaría en un “óptimo” $\alpha = 1$, dado que esto daría como resultado que la composición quedara completamente definida por su *foreground*:

$$I = F \cdot 1 + (1 - 1) \cdot B = F$$

¹Existe además una posible cuarta entrada, que corresponde a fotogramas adyacentes, para el caso en que se desee procesar un video.



Figura 6. Imágenes obtenidas directamente del renderizado en Blender, usando un fondo transparente. Su trimapa es el de la figura 2.

Este resultado, lograría engañar al bloque discriminador, dado que esta composición de por sí es real. Para evitar esto, se realiza la implementación de una segunda red, entrenada únicamente con el dataset de Adobe, en una relación *teacher-student*, donde los resultados de la red G_{adobe} actúan como un *pseudo-groundtruth* [15], añadiendo una etapa de entrenamiento con *loss* en la salida. De esta forma G se entrenaría, castigando con un peso mayor su resultado frente al discriminador y con un peso menor, su resultado frente a la red G_{adobe} , reduciendo, entre épocas este peso.

III. NUESTRO ENFOQUE

Evaluamos el desempeño de ambos algoritmos para imágenes de entrada que contengan personas, en particular, sus rostros. Para ello generamos un dataset y ocupamos métricas de evaluación que se explican a continuación.

III-A. Dataset

Hemos usado dos conjuntos de imágenes para la evaluación de ambos métodos. El primero de ellos, corresponde a un dataset de Adobe®, el cual contiene, entre otras, 11 imágenes de personas y sus respectivos canales alfa y trimaps, ambos segmentados a mano. Solo nos interesarán las imágenes dentro del dataset que contengan personas, puesto que uno de los métodos solo funciona con personas, y solo tiene sentido hacer una comparación mutua de ambos métodos [1] y [2] con entradas que ambos sean capaces de procesar. Expandir el dataset de evaluación no es una tarea fácil, puesto que debería generarse la máscara alfa a mano para cada imagen, o de lo contrario, se debería disponer de buena iluminación y una

pantalla verde (o de algún color constante y no presente en el *foreground*) para capturar fotos o videos en los que se pueda obtener una segmentación de calidad. Nuestra propuesta para expandir el dataset consiste en utilizar Blender, un software de simulación 3D, entre otros, para renderizar modelos de personas con un fondo transparente y así obtener intrínsecamente el canal alfa.

En particular, hemos decidido trabajar con un modelo de la cabeza de David de Michelangelo, al cual le hemos agregado cabello utilizando el método de partículas para evaluar de mejor manera las zonas de transición suave en las máscaras obtenidas en ambos algoritmos. Hemos modelado tres peinados y cuatro posiciones de cámara. Esto para cada color de piel mostrado en la figura 6. En total, disponemos de 36 *foregrounds* para David.

Para la generación del dataset total de evaluación, obtuvimos *backgrounds* de licencia libre de cuatro categorías:

- Interiores
- Ciudad
- Naturaleza
- Misceláneo

En total, para el dataset de Adobe, considerando 11 *foregrounds* y 28 *backgrounds* (7 de cada categoría), obtenemos un dataset de evaluación de 308 imágenes compuestas. Para el caso de David usamos 9 *backgrounds* por imagen, obteniendo un total de 324 imágenes compuestas para el dataset de David. En total nuestro dataset de evaluación consiste de 632 imágenes.

III-B. Métricas de evaluación

Para evaluar el desempeño de ambos métodos, utilizamos las métricas de MSE y MAE, error cuadrático medio y error



Figura 7. Algunos ejemplos de las imágenes compuestas a partir de los fondos.

absoluto medio respectivamente.

$$MSE = \frac{1}{n} \sum_{i=1}^n (\alpha_i - \hat{\alpha}_i)^2 \quad (11)$$

$$MAE = \frac{1}{n} \sum_{i=1}^n |\alpha_i - \hat{\alpha}_i| \quad (12)$$

Cada método arroja a la salida una imagen de un canal que representa el canal alfa del *foreground*. Para la evaluación encontraremos un MSE y MAE para cada imagen de salida, comparando con el respectivo canal alfa del *groundtruth*. En las ecuaciones (11) y (12), cada α_i representan el valor de un píxel en el *groundtruth*, y cada $\hat{\alpha}_i$ representa el valor de un píxel en el resultado práctico.

III-C. Problema de sombras

Evaluamos además, en una prueba cualitativa, el desempeño de ambos algoritmos cuando el fondo, en la imagen compuesta, presenta sombras proyectadas por la persona (*foreground*). Esto pues sabemos que el segundo método propuesto para la evaluación usa la información de un fondo que se captura antes o después sin el sujeto de interés en el *foreground*, es decir, la sombra proyectada por la persona no estaría proyectada en este.

IV. RESULTADOS

A continuación se presentan los resultados obtenidos para la evaluación de ambos algoritmos. Uno de los problemas a los que nos enfrentamos al presentar estos datos, es que la interpretación de números sin una correspondencia a lo que representan haría que esta sección fuer algo compleja de entender para el lector. Por razones de extensión no es posible mostrar el resultado y la interpretación de cada imagen, sin embargo, en la figura 14 un ejemplo de lo que representaría un valor de $MSE = 0,24$, además de adjuntar en el apéndice los mejores y peores desempeños para cada dataset y método, con sus respectivas métricas. Para facilitar la lectura de los gráficos, específicamente de los resultados del dataset de Adobe, ver figura 8.

IV-A. Information Flow Matting

A partir de estos resultados, es posible concluir ciertos puntos sobre la respuesta de los algoritmos. En el caso de *Information Flow Matting*, se puede observar en la figura 9, que para el dataset de Adobe, los peores resultados se obtienen para las imágenes etiquetadas como: *girl4*, *pixels* y *wedding*, estos resultados se pueden explicar debido a que en el caso de la primera etiqueta, corresponde a una imagen donde el cabello del sujeto está al viento y es de un color claro, por lo que la complejidad de determinar la opacidad de los píxeles aumenta. En el caso de la segunda etiqueta se un problema similar, la complejidad se encuentra nuevamente en el cabello del sujeto y el plano hace que existe mucho paso del *background* a través del cabello. Finalmente, para la última etiqueta, el problema es similar, el sujeto está cubierto por una tela translúcida, lo que hace que la complejidad se centre



Figura 8. *Labels* para imágenes del dataset de Adobe. (a) Boy (b) girl (c) girl2 (d) girl3 (e) wedding (f) pixels (g) woman (h) girl4 (i) model (j) bathrobe (k) sunny

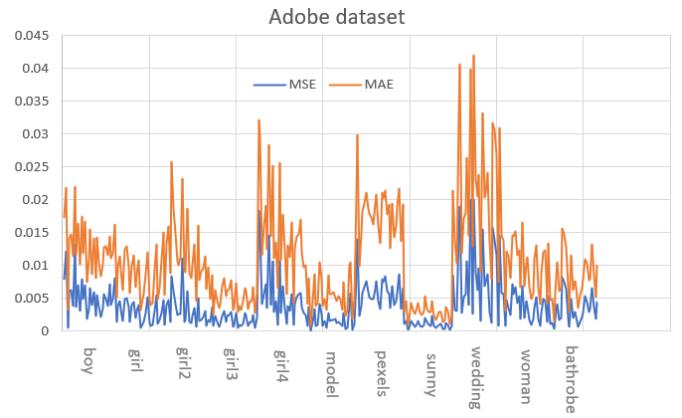


Figura 9. MSE y MAE para dataset de Adobe. Information flow matting.

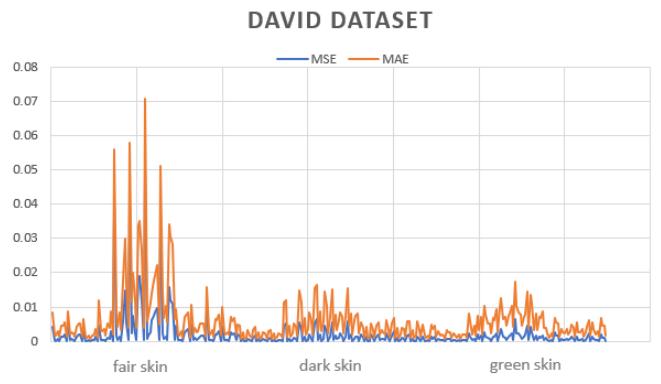


Figura 10. MSE y MAE para dataset de David. Information flow matting.

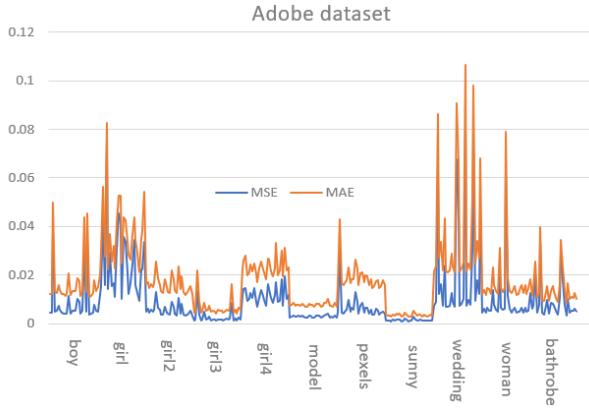


Figura 11. MSE y MAE para dataset de Adobe. Machine learning.

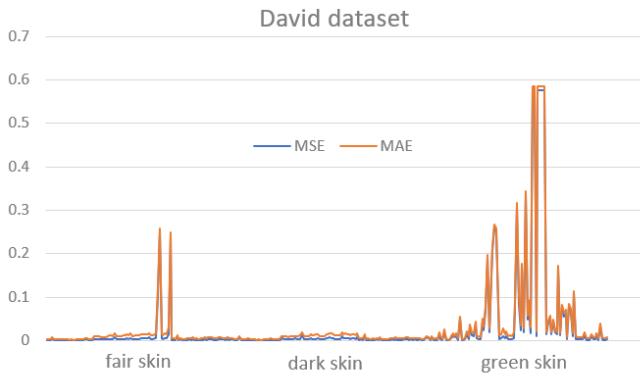


Figura 12. MSE y MAE para dataset de David. Machine learning.

en determinar la opacidad de los píxeles que rodean al sujeto. Estos resultados, concuerdan con lo observado al mirar el *TOP 5*, de los peores resultados del algoritmo para el dataset, en la figura ??, donde analizando el canal alfa, se puede comprobar que los problemas de la máscara se relacionan directamente a la complejidad o transparencia del sujeto original, generando en el canal alfa secciones mal representadas. Para el dataset de David, se puede ver en la figura 10, que los casos donde se centra el mayor error, son donde el sujeto posee *fair-skin*, contrastando con lo obtenido para el *TOP 5* de los peores resultados para el dataset, figura ??, se puede evidenciar que el problema nuevamente tienen que ver con la complejidad y transparencia asociada al cabello del sujeto, este hecho, como se aprecia en los resultados, se potencia cuando el color de éste es similar a alguno de los colores del *background*.

IV-B. The world is your green screen

En el caso de los resultados de *The world is your green screen*, se puede observar en la figura 11, que para el dataset de Adobe, las etiquetas que corresponden a peores resultados se traducen en *girl1*, *wedding* y *woman*. En el caso de la primera etiqueta, el error lo podemos encontrar en la posición del sujeto, la composición de la tela con la mujer hace que la

etapa previa de *soft-segmentation* tenga problemas generando la máscara a priori. Para la segunda etiqueta, el problema es análogo al encontrado para el algoritmo anterior, la presencia de la tela transparente hace que el algoritmo no pueda discernir de forma consistente en la opacidad de los píxeles que rodean al sujeto, generando un canal alfa que presenta agujeros. Finalmente, para la última etiqueta a simple vista analizando al sujeto, no se puede llegar a una cause del problema de fallo, a continuación cuando se analicen los resultados de los peores casos, se evidenciará la razón detrás. Observado el *TOP 5* de los peores resultados del algoritmo para el dataset de Adobe, figura ??, se puede comprobar que los análisis anteriores se confirman, los peores resultados se dan cuando la etapa de *soft-segmentation* no puede detectar de forma correcta al sujeto. En el caso donde el sujeto presenta transparencias, el algoritmos es incapaz de realizar una estimación de la opacidad de los píxeles generando agujeros en el canal alfa. Cabe mencionar, que estos problemas frente a la opacidad son más graves que los hallados para el caso del algoritmo *IFM*. Finalmente, se puede apreciar el porqué de la falla del sujeto asociado a la última etiqueta, cuando el color del *foreground* es similar al color del *background* el algoritmo tiene dificultades para realizar la separación entre ambos, generando agujeros donde el color se interpreta como continuo. Para el dataset de david, se tienen los siguientes resultados 12. Se puede apreciar que la respuesta del algoritmo, sólo falla de forma importante cuando el sujeto tiene *piel-verde*, esto se corrobora al observar el *TOP 5* de los peores resultados del algoritmo, para el dataset ??, donde queda en evidencia que el canal alfa erróneo dado que en la etapa previa de *soft-segmentation* el algoritmo fue incapaz de detectar a una persona, dando como resultado un canal alfa a priori que no puede ser refinado, dado que desde el punto de vista del algoritmo, no hay un sujeto presente. Si bien el caso de la *piel-verde* se podría argumentar, que es un caso irreal, se debe considerar que la piel es reflectante, por lo que al iluminar a un sujeto, con una luz verde, este “aparentaría” tener la piel de este color. Esta falla en el proceso de generación del canal alfa, nos hace entender que el algoritmo sólo puede funcionar de manera adecuada con un sujeto correctamente iluminado.

IV-C. Resultado para Problema de sombras

Probamos con distintos algoritmos basados en afinidad, y que usan un trimapa como entrada, y comparamos sus resultados con los del método que usa *deep learning* y que recibe como entrada el fondo en vez del trimapa. En la figura 13 se adjuntan los resultados obtenidos, en donde se puede apreciar que en los algoritmos de afinidad, se logra obtener una segmentación que, a grandes rasgos, logra captar bien la forma y niveles de transparencia en el cabello del rostro, sin importar las sombras en el fondo. sin embargo, al usar el método *deep learning*, no se logra obtener una segmentación fidedigna, puesto que confunde la sombra proyectada por la cabeza como parte del *foreground*, al no estar presente en el *background* que se ingresa como entrada al algoritmo.

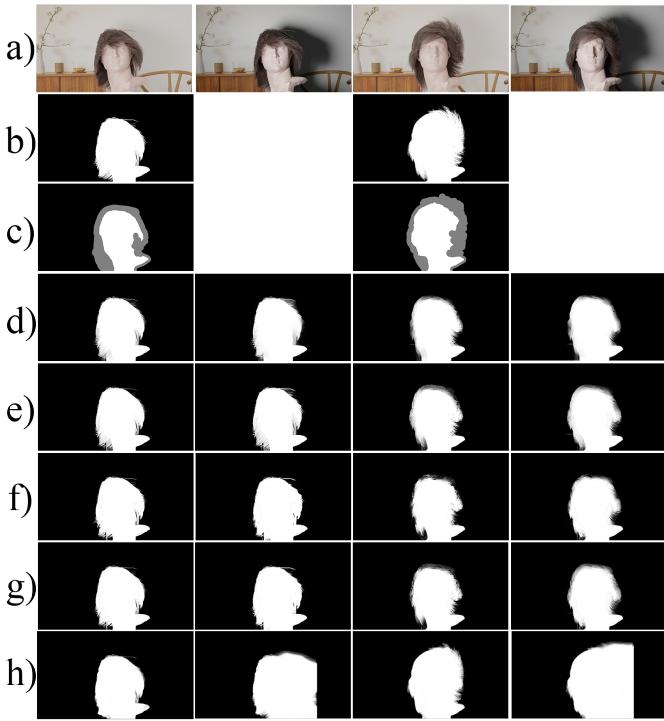


Figura 13. Para cuatro imágenes (a), dos trimapas (c), y sus respectivos *groundtruth* (b) mostramos los resultados obtenidos para distintos algoritmos para la generación del canal alfa: Closed form matting (d), KNN (e), Shared matting (f), Information flow matting (g) y el método basado en aprendizaje que se ha estudiado en este informe (h).

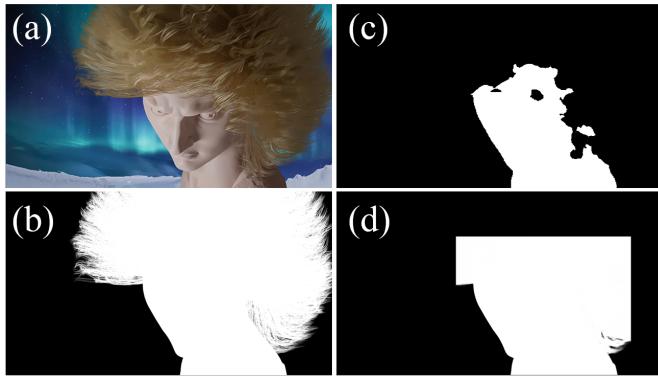


Figura 14. Ejemplo de resultado para un $MSE = 0.24$ para una entrada (a), con el canal alfa *groundtruth* (b), pre-segmentación (c) y salida (d)

V. CONCLUSIÓN

Podemos concluir que, en general, el algoritmo Information Flow Matting, basado en afinidad, sin aprendizaje, tiene un desempeño más robusto. En ningún caso falla tanto al punto de entregar una máscara alfa compuesta de solamente *background*, como se puede observar en la figura ??, en la que la pre-segmentación no es ni siquiera capaz de detectar la ubicación de la persona. De hecho, en los peores casos de Information Flow Matting, se obtiene un $MSE = 0,04$ para David y $MSE = 0,04$ para Adobe, lo que es menor a los *peaks* que se observan en las figuras 12 y 11. De todas formas

al hacer comparación, debemos tener en cuenta que ambos métodos trabajan con distintas entradas.

Para futuras mejoras, proponemos trabajar con un método mixto. En particular proponemos entrenar una red de *deep-matting* que sea capaz de entregar un resultado intermedio, sin necesidad de incluir como entrada un trimapa, y que luego mediante un método, como podría ser *Information-Flow-Matting*, refinar este resultado, para llevárselo a un canal alfa final. Esto dado que este estudio nos permitió darnos cuenta que el método de *Information-Flow-Matting* permite obtener un resultado bastante robusto, con gran detalle y manejo adecuado de transparencias en el sujeto, sin embargo requiere tener como entrada un trimapa. Por otro lado, el camino mediante *deep-learning* permite obtener resultados que en general son buenos, pero dependen mucho de la posición e iluminación del sujeto.

AGRADECIMIENTOS

A los autores les gustaría agradecer a Brian Price por su facilitación del dataset de Adobe; y a Blender Tutor en YouTube por su tutorial para el renderizado de cabello usando partículas, que fue de utilidad para la creación del dataset de David.

REFERENCIAS

- [1] Y. Aksoy, T. O. Aydin, and M. Pollefeys, “Designing effective inter-pixel information flow for natural image matting,” in *Proc. CVPR*, 2017.
- [2] S. Sengupta, V. Jayaram, B. Curless, S. Seitz, and I. Kemelmacher-Shlizerman, “Background matting: The world is your green screen,” in *Computer Vision and Pattern Recognition (CVPR)*, 2020.
- [3] N. Xu, B. Price, S. Cohen, and T. Huang, “Deep image matting,” 2017.
- [4] A. Levin, D. Lischinski, and Y. Weiss, “A closed-form solution to natural image matting,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 30, no. 2, pp. 228–242, 2008.
- [5] X. Feng, X. Liang, and Z. Zhang, “A cluster sampling method for image matting via sparse coding,” in *Computer Vision – ECCV 2016*, B. Leibe, J. Matas, N. Sebe, and M. Welling, Eds. Cham: Springer International Publishing, 2016, pp. 204–219.
- [6] L. Karacan, A. Erdem, and E. Erdem, “Image matting with kl-divergence based sparse sampling,” in *2015 IEEE International Conference on Computer Vision (ICCV)*, 2015, pp. 424–432.
- [7] X. Li, J. Li, and H. Lu, “A survey on natural image matting with closed-form solutions,” *IEEE Access*, vol. 7, pp. 136 658–136 675, 2019.
- [8] S. Cai, X. Zhang, H. Fan, H. Huang, J. Liu, J. Liu, J. Liu, J. Wang, and J. Sun, “Disentangled image matting,” 2019.
- [9] Y. Zhang, L. Gong, L. Fan, P. Ren, Q. Huang, H. Bao, and W. Xu, “A late fusion cnn for digital matting,” in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019, pp. 7461–7470.
- [10] Q. Chen, T. Ge, Y. Xu, Z. Zhang, X. Yang, and K. Gai, “Semantic human matting,” 2018.
- [11] Y. Aksoy, T. H. Oh, S. Paris, M. Pollefeys, and W. Matusik, “Semantic soft segmentation,” *ACM Transactions on Graphics*, vol. 37, pp. 1–13, 07 2018.
- [12] L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff, and H. Adam, “Encoder-decoder with atrous separable convolution for semantic image segmentation,” 2018.
- [13] A. B. Pablo Alcantarilla (Georgia Institute of Technology), Jesus Nuevo (TrueVision Solutions AU), “Fast explicit diffusion for accelerated features in nonlinear scale spaces,” in *Proceedings of the British Machine Vision Conference*. BMVA Press, 2013.
- [14] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, “Unpaired image-to-image translation using cycle-consistent adversarial networks,” 2020.
- [15] S. Sengupta, A. Kanazawa, C. D. Castillo, and D. Jacobs, “Sfsnet: Learning shape, reflectance and illuminance of faces in the wild,” 2018.