Technical Disclosure Commons

Defensive Publications Series

March 31, 2016

APPLICATION OF PATTERNS TO IMAGE FEATURES

Alex Powell

PUBLICATION

APPLICATION OF PATTERNS TO IMAGE FEATURES

Author: Alexander P. Powell

ABSTRACT

Features are described to apply visual patterns or designs to image features of an image. A pattern image depicts a pattern or design created by a user which is intended to be applied to an image feature such as a face. The pattern image is to be applied to modify an input image having detected features such as faces. For example, a face mask is determined that indicates which pixels in the input image are facial skin pixels that are suitable to receive application of the pattern. The pattern image is warped to fit a detected face, and blended pixel values are created for the facial region and provided in an output image by applying pattern image pixel values to the facial region. For example, pixel values derived from the pattern image can be blended, based on the associated face mask, with corresponding pixel values of the input image to create the output image having the blended pixel values. The resulting blended face provides a realistic match to the face in the original image. Using described techniques, patterns can be automatically, flexibly, and accurately applied to image features such as faces in target images without obscuring details of the modified image features, and preserving image noise, sharpness/focus, and other original properties of the image features.

KEYWORDS

- image or photo modification
- face modification
- blend images
- warp pattern
- face paint

DESCRIPTION

This publication describes features related to application of patterns to image features. In some examples, a system can obtain a pattern image, e.g., depicting a pattern or design created by a user and intended to be applied to an image feature such as a face depicted in an image. The system can receive a selection of an input image that is to have the pattern image applied to modify the input image. The system can detect face regions in the input image, e.g., based on facial landmark features such as eyes, nose, and mouth. For each detected face, the system can determine a face mask that indicates which pixels in the input image are facial skin pixels that are suitable to receive application of the pattern in the pattern image. The system can warp the pattern image to fit a detected face, and can then create blended pixel values for the facial region and provided in the output image by applying pattern image pixel values to the facial region. For example, the system can blend pixel values derived from the pattern image with corresponding pixel values of the input image to create the output image having the blended pixel values, where the blending can be applied based at least in part on the associated face mask.

A variety of features can be included. For example, the pixel value that is blended with the corresponding input image pixel value can be based on a color value of a corresponding pixel of the pattern image, and this blended pixel value can also be based on a luminance of a corresponding pixel of the input image. This allows the blended color value to have a similar luminance to the input image pixel is it being blended with, allowing a more natural and blended appearance. Some implementations can adjust the luminance used in the determination of the blended pixel value based on a luminance of one or more skin pixels sampled from the input image, e.g., to increase the luminance value for darker color values of skin pixels. Some implementations can determine a coverage value used to control the blending, where the coverage value can be based on a face mask value of the face mask (allowing blending to occur if the input image pixel is a facial skin pixel), an opacity value or channel associated with the pattern pixel value (e.g., allowing the creator of the pattern image to control locations and amounts of pattern image colors to be applied to an input image), and/or an average luminance based on sampled skin pixels of the input image (allowing adjustment of pattern image colors based on darkness or lightness of skin tone). For example, the face mask and the opacity channel can control where pattern color is applied on the output image.

Various described features can allow a system to warp the pattern image to align with a face in an image, e.g., by aligning facial landmark points between the pattern image and the target face in the input image, and warping the pattern image to fit the size and shape of the face. Such features can cause accurate application of pattern color values to a face. In some implementations, the pattern image can be blurred to match an amount of blur detected in the input image. In some examples, the face mask can be created by sampling one or more color values from one or more face areas in a face (e.g., cheek areas detected via facial landmark features), and designating pixels of the mask to indicate face pixels or non-face pixels based on color distances between corresponding input image pixels and the sampled color values.

Multiple faces in a single image can be modified with patterns. Each face can be masked by a soft, accurate facial skin mask, which can be created for any skin color, and is able to mask out occluding features such as facial hair, bangs, or other objects in front of the face so that such features will not be modified by the pattern. Faces need not be oriented towards the camera, since the paint image is warped to match the detected features. The pattern image can be defined by a user with reference to a reference face template, and can include the opacity channel (e.g., value) to artistically control where the pattern color is applied on a target face and how much of the pattern color is applied to the target face.

These and other described features can allow patterns to be automatically, flexibly, and accurately applied to image features such as faces in target images without obscuring details of the modified image features, and avoiding tedious manual editing of images to apply desired patterns to image features. For example, described features allow a pattern defined in a pattern image to be applied to a face in a target image in a manner that preserves the original lighting, skin texture detail (such as wrinkles and pores), hair color and detail, and skin tone of the face. Original properties of the input image such as image noise and sharpness/focus of the face can be preserved as well. The resulting blended face provides a realistic match to the face in the original image. Features allow faces and other image areas to be "painted" with desired image patterns.

In implementations or situations in which the techniques discussed herein may collect personal information about users, or may make use of personal information (e.g., user data), users are provided with one or more opportunities to control how information is collected about the user and used in one or more described features. A user is provided with control over whether programs or features collect user data (e.g., information about a user’s characteristics (age, gender, profession, etc.), social network, actions or activities, preferences, content created or submitted by a user, a user's current location, etc.). A user is provided with control over whether programs or features collect user information about that particular user or other users relevant to the program or feature. Each user for which personal information is to be collected is presented with one or more options to allow control over the information collection relevant to that user, to provide permission or authorization as to whether the information is collected and as to which portions of the information are to be collected. For example, users can be provided with one or more control options over a communication network.

Fig. 1 illustrates a block diagram of an example network environment 100, which may be used for some features described in this disclosure. In some implementations, network environment 100 includes one or more server systems, e.g., server system 102 in the example of Fig. 1. Server system 102 can communicate with a network 130 and include a server device 104 and a database 106 or other storage device. Network environment 100 also can include one or more client devices, e.g., client devices 120, 122, 124, and 126, which may communicate with each other and/or with server system 102 via network 130. Network 130 can be any type of communication network.

Fig. 1 shows one block for server system 102, server device 104, and database 106, and shows four blocks for client devices 120, 122, 124, and 126. Server system 102 can communicate with other server systems via the network 130. Database 106 and/or other storage devices can be provided in server system block(s) that are included in and/or separate from server device 104. There may be any number of client devices. Each client device can be any type of electronic device, e.g., desktop computer, laptop computer, portable or mobile device, cell phone, smart phone, tablet computer, television, TV set top box or entertainment device, wearable devices (e.g., display glasses or goggles, wristwatch, headset, armband, jewelry, etc.), media player, game device, etc. In other implementations, network environment 100 may not have all of the components shown.

In various implementations, end-users U1, U2, U3, and U4 may communicate with server system 102 and/or each other using respective client devices 120, 122, 124, and 126. In some examples, users U1, U2, U3, and U4 may interact with each other via applications running on respective client devices and/or server system 102, and/or via a network service. In some examples, server system 102 and/or one or more client devices 120-126 can provide an image editing program and/or an image display program. The image editing program may allow a system (e.g., client device or server system) to provide options for editing and displaying an image, examples of which are described herein.

Some implementations can provide features described herein using a client device or server device disconnected from or intermittently connected to computer networks. In some examples, a client device can examine and display images stored on storage devices local to the client device and can provide described features viewable to a user on a connected display device.

Fig. 2 is a flow diagram illustrating one example of a method 200 to apply patterns to image features, e.g., to faces. Method 200 can be implemented, for example, on a server system 102, and/or on one or more client devices 120, 122, 124, or 126 as shown in Fig. 1. Described implementations can be used with single images or with one or more images from one or more series or video sequences of images.

Some implementations can initiate method 200 based on user input. A user may, for example, have selected the initiation of the method 200 from a displayed user interface. In various implementations, method 200 or portions thereof can be performed with guidance by the user and/or can be initiated automatically by a system. For example, the method (or portions thereof) can be periodically performed, or performed based on one or more particular events or conditions, e.g., an application being opened by a user, receiving one or more images that have been newly uploaded to or accessible by the system, a predetermined time period having expired since the last performance of method 200, and/or one or more other specified conditions. The method 200 can be performed for uploaded or captured pattern images or pre-designated pattern images, or for a large collection of accessible input images and/or pattern images.

In some implementations, a library of available pattern images can be accessible to the system for selection and retrieval, e.g., from public and/or private data sources. For example, pattern images intended to be applied to faces can be stored and organized based on categories such as sporting events (e.g., patterns showing team logos), holidays (e.g., thematic face patterns for Halloween, St. Patrick's Day, or other holidays), particular locations (e.g., thematic patterns for a particular country), etc. Similarly, input images to receive patterns can be organized by a user in particular categories or subjects.

In block 201, it is checked whether user consent (e.g., user permission) has been obtained to use user data in the implementation of method 200. For example, user data can include user preferences, information about a user's social network and contacts, user characteristics (identity, name, age, gender, profession, etc.), social and other types of actions and activities, content, ratings, and opinions created or submitted by a user, a user's current location, historical user data, etc. One or more blocks of the methods described herein may use such user data in some implementations. If user consent has been obtained from the relevant users for which user data may be used in the method 200, then the blocks of the methods herein can be implemented with possible use of user data as described for those blocks. If user consent has not been obtained, then blocks of the methods herein are implemented without use of user data. In some implementations, if user consent has not been obtained, the remainder of method 200 is not performed.

In block 202, the method obtains an input image and one or more pattern images for processing. These images can be digital images composed of multiple pixels, for example, and can be stored on one or more storage devices of the system or otherwise accessible to the system, e.g., a connected storage device such as a local storage device, storage device connected to or in communication with a network accessible to the system, etc. For example, the input image can be a photo captured by a camera, an image frame extracted from a captured video stream or other video data, or an image derived from a different source.

The input image is designated to be modified with one or more patterns from the pattern image(s). Each pattern image can be an image created by a user or device for use in modifying other images such as the input image. For example, each pattern image can include a pattern of color values (or other pixel values) intended to be blended with or otherwise modify one or more image features (e.g., faces) depicted in an input image. For example, the pattern image may have been created to apply to the particular input image, or may have been designed to be generically used with any of multiple input images. In some implementations, multiple pattern images can be obtained, e.g., where different pattern images are intended to be applied to different faces of multiple faces depicted in the input image.

In some implementations, the input image and pattern image(s) can be automatically obtained by the method, e.g., as an image from a stored collection of multiple images, e.g., from a user's album, a pool of stored images submitted by users, a collection of pattern images created by one or more users, etc. The collections can be locally stored and accessible by the device performing method 200, and/or can be remotely stored on a server or client device, e.g., as one or more albums provided in account(s) of user(s) of a network service. In some implementations, the system can determine which input image and/or pattern image to select based on evaluating one or more characteristics of accessible images, e.g., timestamps and other metadata of images, the color distributions of images, the recognized content or labels describing content in images, user data such as user preferences, etc. In some implementations, a user can provide, select, or designate one or more input images and/or pattern images to obtain for processing.

For example, a system can automatically (e.g., without human intervention) select a particular input image to be modified by one or more particular pattern images. Such selection may be determined based on user data, including stored user preferences, a user history of previous modifications made by the user to other images, social data indicating user preferences (e.g., previous comments, ratings, etc. made by the user), recent or upcoming events based on the user's stored calendar or planner, locations visited or to be visited by the user (e.g. as detected by GPS sensors on a device carried by the user), activities of the user (e.g., sensed or inferred by locations visited by the user), etc. For example, the system may automatically determine that the user is going to attend a soccer game tomorrow, and can automatically select a pattern image showing the logo of the user's favorite team for application to the face of the user depicted in an input image showing the user at a previous soccer game. In other examples, the system can determine that the user attended or will attend a holiday party, travel to a particular country, or perform some other activity related to a particular pattern in a pattern image, and can select such a pattern and a related input image depicting the user's face to receive the pattern.

In block 204, one or more faces are detected in the input image. For example, face regions can be detected by image recognition techniques, e.g., facial recognition techniques. The recognizer detects faces in the image anonymously, without determining or requiring identity information for the persons who are depicted. In some implementations, method 200 can implement facial recognition in block 204 to identify the positions and features of one or more faces in the image. For example, method 200 can send the obtained image to a facial recognizer and receive facial recognition results from the recognizer. The facial recognizer can recognize a face using any of multiple techniques. In some examples, the recognizer can identify an image feature as a human face by identifying facial landmark features (and/or other features) having a known general shape or profile and spatially located in the image, such as eyes, nose, and mouth, and having a known spatial relationship with each other (e.g., mouth being below eyes, etc.) and/or with other identifiable features (such as other head features, body features, etc.). Some facial recognizers can provide face location data such as coordinates of a face region, e.g., a face polygon, based on identified facial landmark features.

Furthermore, other facial information besides face location data (e.g., face polygon) can be provided by the facial recognizer or other face detection process. For example, the facial recognizer can provide, as output, points or other locations of the facial landmark features found in the face for the facial identification process. Spatial coordinates of points on one or more of the facial landmark features can be provided, such as the mid-point of each eye, left and right endpoints of the eyes, points indicating the spatial extent of the mouth, a mid-point of the nose, etc. (more examples of which are described below). Information can be provided from a recognizer or other analysis which indicates an identified angle of orientation of the face relative to the plane of the image, such as whether the face is being depicted directly facing the plane of the image or is angled and looking to one side.

If user consent has been obtained from the users depicted in the image, some implementations can provide characteristic information about a face or person that was detected by the recognizer, such as estimated age, gender, and other characteristics of the person whose face is being depicted. Such information can be used in determining whether to apply a pattern image to the input image, which pattern image to select to apply to the input image from a set of multiple available pattern images, etc. Access to personal information including user characteristics such as age, gender, identity, etc., may be restricted based on users' control and preferences governing such information.

In block 206, the system selects a face in the image. For example, if multiple faces were detected in block 204, the system selects one of the detected faces. In block 208, a face mask is determined for the selected face. The face mask is a pixel map having mask pixels corresponding to pixels of the input image, where each mask pixel indicates whether the corresponding input image pixel is a face pixel. The face mask can indicate a face region, e.g., in which pixels of the input image are to be modified based on the pattern image. In some implementations, each mask pixel can indicate a degree of modification for the corresponding pixel of the input image. Any of multiple different techniques can be used to determine the face mask in particular implementations. For example, the determined face mask can be a facial skin mask which indicates which pixels of the image belong to the skin of depicted persons' faces, such that the facial skin pixels do not include pixels depicting other facial features such as eyes and mouth. This allows adjustment to the facial skin pixels to be performed without modifying eyes, mouth, and other facial features. Some implementations for determining a facial skin mask are described below with respect to Fig. 3. Some implementations of block 208 can provide an image mask that indicates skin pixels, including pixels depicting non-facial portions of a person's exposed skin, e.g., neck, bare arms or legs of the person, etc.

In block 210, the system may refine the face mask determined in block 208. For example, the determined face mask may have undesired artifacts, such as pixel irregularities, resulting from the mask creation process. In one example, the face mask may include small pixel "holes" (e.g., discontinuous or sharply changing areas surrounded by different-valued pixels), pixel protrusions extending out from sides of mask features, or other irregularities resulting from noise or other processing artifacts. Such imperfections can be reduced by using image restructuring techniques. For example, morphological dilation can be used on the mask to fill in undesired holes in the mask, e.g., by examining pixels neighboring the holes and extending the neighboring pixel values to the hole pixels. In some implementations, morphological erosion can be used on the mask after the morphological dilation to reduce portions of the mask that may have been increased in area resulting from the morphological dilation. Such refining allows sharp discontinuities to be reduced in the mask, thus, reducing sharp falloffs in the application of the pattern image pixel values to the input image as described below.

In some implementations, the face mask can be refined in block 210 by using a blur operation on the mask. For example, the face mask can be blurred after the morphological operations are applied to the mask as described above. The blurring can reduce the number or extent of hard or sharp edges in the mask, e.g., provide a smooth boundary to the face region indicated in the mask. This smoothing can allow the blending of the pattern image and the input image to be less noticeable and smoothly transition between pattern image pixel values and original input image pixel values.

In block 212, the system warps the pattern image to align with the face region. If multiple pattern images were obtained in block 202, a pattern image associated with (e.g., intended to be applied to) the selected face is selected and warped in this block. The pattern image can indicate pixel values (e.g., color values) that are to be applied to the selected face in the input image in particular locations of the face. In some implementations, the pattern image has a standard shape, e.g., has been fitted to a reference face or template used to define pattern images for use with the method 200. For example, the pattern image can indicate color values to be applied within an area conforming to the reference face, and can include regions having no color values that are intended to map to the eyes and mouth of a face.

The pattern image can also include facial landmark points that indicate particular positions of facial landmark features on a face to which the pattern image is to be applied. Facial landmark points can include points in the face that define a boundary of facial features, e.g., eyes, lips, nose, and other main landmark features of a face. In some examples, the pattern image can include landmark points at the centers and/or corners of eyes, left and right ends of eyebrows, centers of upper and lower lips, left and right mouth corners, tip of nose (ventral region of the septal cartilage), center base of nose (caudal region of the septal cartilage), and/or left and right base of the nose (left and right alar bases). Some examples of a pattern image are shown with respect to Figs. 6B and 7.

The pattern image can be aligned to the selected face in the input image by aligning one or more of the facial landmark points in the pattern image to corresponding facial landmark points of the selected face. For example, the detected face region in block 204 can include landmark information corresponding to one or more of the landmark points that are defined for the pattern image. In some implementations, landmark points of the pattern image that are occluded from visibility in the face of the input image can be discarded (e.g., ignored). For example, some landmark points of the selected face may not be visible in the input image, e.g., due to the orientation of the face. In some examples, a left cheek of the face may not be visible if the face is facing to the right side of the image, or the bottom of the nose may not be visible if the face is oriented toward the bottom of the image. Landmark points based on such occluded features can be discarded, e.g., removed from or ignored in the pattern image so that the pattern image will not be folded on itself when applied to the face.

Once corresponding landmark points of the pattern image and selected face are aligned, the system can warp the pattern image to match the spatial layout or configuration of the selected face. In some implementations, a spline-based warping technique can be used to apply the warp, e.g., thin-plate splines. The warping technique ignores the discarded landmark points of the pattern image (e.g., removed due to occluded corresponding landmark points on the face) such that the portions of the pattern image surrounding the discarded landmark points are not warped onto the face (e.g., applied onto the face region without warping). For example, the use of warping techniques, discards of occluded landmark points of the pattern image, and particular facial landmark points (e.g., nose landmark points for tip of nose, center base of nose, and left/right base of nose) can effectively warp the pattern image onto the face to simulate a 3D application or effect of the pattern on the face. The system need not determine and use a 3D model of the face for this warping technique. For example, since a nose protrudes from a face, the warping of the pattern image based on nose landmark points enhances the 3D effect of the warping of the pattern image.

In some implementations, a 3D model of the face can be determined, e.g., using associated depth information (e.g., associated with the input image from a 3D camera or other source). In some cases, a 3D model can be estimated for the face based on aligning or warping facial landmark points to a stored generic 3D face model. The pattern image can be modified to conform to the 3D model, rotated to an appropriate angle based on detected face angle in the input image, and aligned onto the face in the image.

In some implementations, the system can provide an amount of blur in the pattern image that matches a detected amount of blur present in the input image. For example, a sharpness estimate or focus estimate for the selected face can be determined using known techniques, and the pattern image can be blurred by the same amount and/or to obtain the same amount of blurriness. This can allow, for example, the pattern image to match an out-of-focus face, e.g., a face that may be in a background of an input image.

In block 214, the system applies the pattern image to the selected face in the input image to determine a blended face in an output image. The pattern image has been aligned with the selected face in the input image and one or more pixels of the pattern image can be used to modify corresponding pixels of the input image. For example, a blending of values derived from the pattern image and corresponding values of the input image can be performed. The face mask determined in block 208 can be used to apply the color values of the pattern image to pixels of the selected face. Other factors can also be used to adjust the color values, including opacity values associated with the pattern image, luminance of the pixels of the input image, etc. Some examples of applying the pattern image are described below with respect to Fig. 4.

In block 216, the system may determine whether there is another face in the input image to modify with a pattern image. For example, each of multiple pattern images obtained in block 202 may be assigned to be applied to a particular face in the input image. If there is another face to process, the method returns to block 206 to select the face. In some implementations, multiple pattern images can be applied to a single face in an input image. For example, each pattern image can be applied to a face in succession, e.g., using the blocks of method 200 as described above. In some cases, it may be more difficult to determine an accurate face mask for a face that has been previously modified with a pattern image, e.g., due to differences in color values over the area of the face as applied by the previous pattern image.

If there are no more faces in the image to process with a pattern image, the method may continue to block 218, where the system may provide the output image for display. For example, the system can provide the output image to a display device of a server device or client device, e.g., a display screen. In some implementations, the system can send the output image, e.g., over one or more networks, to be displayed by display devices of one or more server devices and/or client devices. Some implementations can display the output image in a user interface allowing a user to view the output image as well as select a different pattern image to apply to the original input image, or to apply to the output image. For example, after such a selection, the method 200 can be performed for the different pattern image and provided for display.

Fig. 3 is a flow diagram illustrating an example method 300 to determine a facial skin mask for a face in an input image, e.g., for block 208 of Fig. 2. The processing of method 300 (and/or the other methods described herein) can be performed in any of a variety of different color spaces, e.g., HSV (hue-saturation-value), HSL (hue-saturation-lightness), RGB (red-green-blue), YCbCr, etc. In some example implementations, the HSV color space is used.

In block 302, the system samples particular face areas of the selected face for pixel values to represent skin tones. For example, designated sample areas of a face can be used to sample pixel values representing skin tones. The pixel values can be color values, for example, or other pixel values, e.g., brightness values if the input image is grayscale or black and white (different pixel value types are referred to herein as "color values" for simplicity). The designated sample areas can be selected based on a known likelihood of the areas having one or more skin tones and/or a variety of different samples of skin tones. For example, some implementations can statistically sample areas such as one or more cheek areas of the face as determined based on facial landmark points indicating facial features and borders of facial features such as eyes, nose, and mouth. In some examples, such cheek sample areas can be defined by triangles defined on each side of the face between the eye, the mouth corner, and the nose on the respective side of the face. If the face orientation or angle is such that the view of one cheek area is occluded (e.g., occluded by more than a threshold amount as compared to a view of the cheek when the face is directly facing the plane of the image), samples are not taken from that occluded cheek area. The cheek triangular area may include different shades of skin tones, e.g., due to shadowed and unshadowed skin surfaces (e.g., shadows from the nearby nose), and therefore can be a suitable sampling area. Other facial areas can be sampled. In some implementations, a predefined number of samples can be taken from each sample area.

The samples of pixel values from the input image can be used to form a color estimate for the color of skin of the selected face. In some cases, the samples may include not only actual skin colors, but other colors, e.g., from facial features different from skin. For example, the samples may include color values of glasses, sunglasses, hair, skin marks (freckles, moles, pimples, etc.) if such features happen to be in the sample area.

Since the sampled skin tones are sampled based on their location in particular areas of the face, and not based on their actual color values, the skin tones sampled may not actually be the color of human skin. For example, black and white photos, painted faces, and the like can be used with the present process to sample face pixel values that do not have a skin color. These sampled pixels can still be used with the features described herein to determine a face mask and apply a pattern image to a face.

In block 304, the system sorts the sampled color values based on a color distance to an average of the samples, and outliers are removed. For example, using the triangular cheek areas as sample areas as described above, all the color value samples from both cheek areas can be averaged to obtain an average color value representative of the skin tone. The sampled color values are then compared to the average color value, and sampled color values having the greatest color distance (e.g., least color similarity) to the average color value are discarded as outliers. For example, a predetermined number or percentage of sampled color values can be discarded, e.g., 25% of the sampled color values. This block can cause non-skin samples (e.g., sampled from facial features such as glasses, hair, etc.) to be discarded, where such samples are likely to be outliers that are not close in color to the average skin color.

In some implementations, the system can also discard samples that are too close in color to another one of the samples, e.g., if there is a color distance between these samples that is under a threshold. For example, a wide sampling of colors may be desired, where samples that are different from each other are useful for comparisons to pixels of the input image in determining the skin mask in block 306. Samples that are close in value to another sample may not be as useful for comparisons and may increase processing time and/or the use of processing resources, and therefore can be discarded.

In block 306, the system selects a pixel of the input image for which to determine a mask value. In some implementations, the pixel can be selected from the entire input image. Some implementations can select a pixel from a particular area of the input image, e.g., an approximate face area that includes the selected face region and a portion of the image surrounding the selected face region. Mask pixels outside this particular area can be assumed to indicate non-facial pixels.

In block 308, the system determines a mask value corresponding to the selected pixel based on a color distance between the selected pixel and the closest sample value of the samples remaining after block 306. For example, the color value of the selected pixel is compared to the sample color value that is closest to the selected pixel color value of the remaining samples. The mask value is determined based on the color distance between these values. In some implementations, the mask value can be chosen from a continuous value range to indicate a degree to which the mask pixel is a face pixel. In one example, a mask value of 1 can be considered a 100% indication of a face pixel, and mask value of 0 can be considered a 0% indication of a face pixel, and a value in between 0 and 1 can be considered a partial indication of a face pixel as indicated by the value. For example, the mask value can be set closer to 1 if the color distance between selected pixel and closest sample is smaller, while the mask value can be set closer to 0 if that color distance is large. In one example, an upper threshold and a lower threshold can be used, where a color distance above the upper threshold provides a mask pixel indicating a 100% face pixel, a color distance below the lower threshold provides a mask pixel indicating a 0% face pixel, and a color distance between the thresholds indicates a partial face pixel that has a value between 0 and 1 based on the amount of color distance.

In some examples, in a black and white visual representation of the mask pixels, the mask value of 1 can be displayed white, 0 can be displayed black, and in-between values can be displayed a corresponding shade of grey (e.g., lighter grey for values closer to 1, and darker grey for values closer to 0). In other implementations, the face mask can provide only 0 and 1 values indicating whether each mask pixel is a face pixel or a non-face pixel.

In some examples, weights can be applied to particular characteristics or channels of color to emphasize or deemphasize those characteristics or channels in the comparison to the closest sample value. For example, the color distance between the selected pixel and the closest sample value can be determined in the HSV color space, and a different weight can be applied to each color channel (e.g., hue, saturation, and value) of the HSV color space. In one example, the hue channel can be weighted with a value of 20, the saturation channel weighted with a value of 4, and the value channel weighted with a value of 3. Such a weighting distribution can emphasize the hue characteristic of the comparison over the other channels, e.g., because hue typically cannot change significantly and still be considered a skin tone, while saturation and value can vary to a greater degree and still be considered a skin tone.

In block 310, the system adjusts the mask value determined in block 308 based on a falloff region surrounding the face. For example, a falloff region can be defined based on a distance from a center of the selected face. For example, for large distances away from the center of the face, a falloff function can be used to gradually change the mask value to a non-face value (e.g., towards 0) in a direction away from the center. In some implementations, the gradual change can start at distances more than a predefined distance from the face center, e.g., in a circular shaped falloff region. Some implementations can use an ellipse-shaped falloff region that is centered on the face and approximately surrounds the face, e.g., as determined by standard distances based on the known facial landmark features of the face. For example, an inner radius of the ellipse can fully enclose the detected landmark points on the selected face, and an outer radius can fall off smoothly.

The falloff region of block 310 can be used to disregard features of the input image that might be considered part of the face due to similarity to skin color, but which are not part of the face. For example, a hand of a person that may be positioned close to the face may be detected as having face pixels in block 308, and block 310 can set the hand pixels as non-facial mask pixels because the hand is positioned beyond the falloff region borders.

In block 312, the system can determine whether there is another pixel of the mask to select for processing. If so, the method returns to block 306 to select another pixel and determine its mask value in the following blocks. If there are no more pixels of the mask to process, the method returns to block 210 of Fig. 2.

The resulting face mask of method 300 can reduce the application of pattern image color to highlights and shadows on facial skin, which would not be considered to be full facial skin due to their color distance from the samples of skin color. For example, some implementations apply a pattern image to a face to generally affect the mid-tone colors of the face, and not to greatly affect highlights and shadows. In one example, specular highlights on skin may not be desired to be affected by the pattern image since these highlights primarily show a color of the light that is reflecting off the skin, and not a color obtained from the diffuse properties of the skin. Such highlights will not be considered skin in the face mask due to their color distance from the sampled skin tones. Similarly, shadows can retain their dark lighting after application of the pattern image since shadows are not strongly associated with facial pixels in the face mask.

Fig. 4 is a flow diagram illustrating an example method 400 to apply a pattern image to the selected face in the input image to provide a modified face, e.g., for block 214 of Fig. 2. In this example, method 400 can determine a modifier image that includes pixels based on the color values of the pattern image, and then blend the input image with the modifier image to determine an output image.

In block 402, the system selects a pixel of the input image. For example, the pixel can be a pixel selected from an area outside an approximated face area determined from the facial landmarks of the selected face. In other implementations, the pixel can be selected from the entire area of the input image.

In block 404, the system determines the luminance of the selected pixel. For example, the luminance can be determined using a standard formula. In one example, the luminance can be calculated using a formula from the standard Rec. 601 (CCIR 601) that utilizes RGB channels of the pixel value, e.g., $Y' = 0.299 R' + 0.587 G' + 0.114 B'$.

In block 406, the system determines a skin tone compensation for the luminance determined in block 404. The skin tone compensation adjusts the luminance value based on whether the skin tone is darker or lighter. For example, if the luminance value is on the darker end of the scale used, the luminance value can be adjusted higher. If the luminance value is on the lighter end of the scale, it can retain its value from the input image. In one example, two thresholds can be used. If the luminance value is below the lower threshold (e.g., 0.4, 0.38 etc.), the luminance value can be adjusted higher by a full amount, e.g., raised to a particular power (e.g., 0.6). A luminance value above the higher threshold (e.g., 0.5, 0.53, etc.) can remain unadjusted. A luminance value between the thresholds can be increased by an amount based on its magnitude between the thresholds, e.g., based on a continuous function between the thresholds.

In block 408, the system determines a color value of a modifier image pixel corresponding to the selected pixel, where the color value is based on the luminance of block 406 and the color value of a corresponding pixel of the pattern image ("pattern pixel color"). The modifier image is an image that will be blended with the input image to apply the pattern of the pattern image to the input image, as described below. The corresponding pixel of the pattern image is the pixel of the pattern image that has been matched to a particular input image pixel after the warping of the pattern image to fit the selected face.

In some implementations, the color value of the modifier image pixel can be determined by blending between the corresponding pattern pixel color, black, and white. The blending can be performed according to a blend factor that is based on the luminance of the pixel from block 406. For example, the blend factor can indicate which pattern color value to select along a gradient range of values from black at one extreme, to the pattern pixel color at a mid-point, to white at the other extreme. In one example, the gradient range of values can range from a value indicating 0% luminance of the pattern pixel color (e.g., black), to increasing percentages of luminance of the pattern pixel color (e.g., 1% to 99% of the pattern color), to a value indicating 100% luminance of the pattern pixel color (e.g., white).

Fig. 5 is a graphical illustration of one example of a range of pattern color values for the modifier image pixel are shown as a gradient range 500. A gradation of the color value for the corresponding pixel of the pattern image is shown, between a color value of black at one end of the range and a color value of white at the other end of the range. The color value that is selected within the range 500 for the color value of the modifier image pixel is determined based on the luminance from block 406. The blend factor (e.g., luminance value) is indicated by the position of the slider 502 along the gradient range 500, where a larger luminance positions the slider 502 toward the right of the range 500 and a smaller luminance positions the slider 502 toward the left side of the range 500. A color value in the range 500 is selected for the modifier image based on the luminance value of the corresponding input pixel.

Referring back to Fig. 4, in block 410, the system determines whether there is another pixel to select from the input image for processing. For example, the system can process all of the pixels in the approximate face area determined for the selected face, or can process all of the pixels of the image. If there is another pixel to select for processing, the method returns to block 402. If there are no more pixels of the input image to select for processing, the method continues to block 412.

In block 412, the system may store the color values determined in the iterations of block 408 in a modifier image. The modifier image can be the same size and resolution as the input image, for example. For example, the modifier image pixels determined in block 408 can include pixels in the approximate face area of the selected face, which have been assigned color values based on the corresponding pixels of the pattern that has been fitted to the face as described for block 408. The resulting modified image has a selected face portion having colors based on the pattern image.

Some pixels of the modifier image that are outside the pattern image can be any value. Many of these pixels will be masked out in the blending process to determine the output image as described below. In some implementations, in the modifier image, pixels outside the approximate face area can be clamped to the color values of the nearest pixel provided by the pattern image. This may allow a smoother appearance in the output image if the mask does not accurately mask out some of the pixels outside the warped pattern in the modifier image.

In block 414, the system selects a pixel of the input image. For example, the pixel can be a pixel selected from the approximate face area of the selected face, or alternately can be selected from the entire area of the input image.

In block 416, the system determines a coverage value for the selected pixel based on the face mask, the luminance of the selected pixel, and a corresponding opacity value. The coverage value indicates how much of the pattern color of the corresponding pixel in the modifier image will be applied to the selected pixel in the blending operation (described below in block 418). The coverage value is based at least partially on the mask value of the face mask determined in block 208 of Fig. 2, where the mask value corresponds to the selected pixel. For example, the corresponding mask value indicates a degree to which the selected pixel can be considered a face pixel, and the coverage value incorporates this mask value. If the mask value indicates that the selected pixel is a non-facial pixel, then the coverage value will block all blending of the color of the corresponding modifier image pixel in the blending operation of block 418.

In some implementations, the coverage value can be adjusted based on a luminance of one or more pixels of the input image. For example, some implementations may produce an undesired output image effect if a color value of the modifier image is blended with a skin pixel that is darker in tone. In some implementations, the coverage value can be adjusted so that a smaller amount of the pattern color of the modifier image is blended with the selected pixel of the input image in block 418 due to a lower luminance detected in or derived from the input image. In some implementations, the coverage value can be adjusted based on an average luminance value of the luminance values of sampled pixels of the input image.

In some examples, the luminance-related adjustment can reduce the coverage value by multiplying it by a factor of less than 1. For example, the factor can be determined based on an average luminance of the luminance values of the sampled facial skin pixels of block 304 of Fig. 3. In some implementations, the factor can be the average luminance plus a modifier empirically determined, e.g., 0.3. The factor can also be clamped between predefined limits that are less than 1. For example, the factor can be clamped between 50% and 90% such that the full coverage value is never used (and thus the full pattern color value is never blended with the selected input image pixel). In some implementations, the full coverage value is not used because of an unnatural effect resulting if the full pattern values are blended with the input image pixels. In some implementations, if the luminance of the selected pixel is less than a luminance threshold, the coverage value is reduced, e.g., similarly as described above.

The coverage value can also be adjusted based on a corresponding opacity value. An opacity value can be associated with each pixel (or some pixels) of the pattern image. For example, a user that created the pattern image may have assigned the opacity values to the pixels of the pattern image, where some opacity values allow the modifier image color value to be fully blended with the color value of the input image (e.g., without additional reduction from the opacity value), and some opacity values can reduce the strength or amount of the corresponding color to be blended with the input image pixel value. An opacity value reduction can cause further reduction to the coverage value that may have already been reduced by luminance and/or mask value as described above.

For example, a pattern can be defined to be applied to (e.g., blended with) an entire face or only a portion of a face. The pattern in a pattern image may be applied to only a portion of a face, and the pattern may only cover a portion of a standard-sized pattern image. In some implementations, portions of the pattern image that are not to be applied can be assigned an opacity value that blocks the pattern image color values of those portions from being blended with the corresponding pixels of the input image. Some opacity values can reduce application of the corresponding pattern color values by a percentage amount, e.g., 50%, etc. In some implementations, some values of opacity values can cause an increase in the amount of the corresponding pattern image color values that are applied to (e.g., blended with) the corresponding input image pixels.

In block 418, the system blends the color value of the selected pixel with the color value of the corresponding pixel of the modifier image based on the coverage value determined in block 416, to produce a blended pixel value for an output pixel of an output image. For example, the coverage value for the selected pixel can be used to linearly interpolate between the selected pixel of the input image and the corresponding pixel of the modifier image. In some implementations, an alpha blending of the selected pixel and modifier image pixel is performed.

The coverage value has been determined in block 416 such that if the selected pixel is a facial skin pixel, an amount of the corresponding pattern image color value as provided in the modifier image may be blended with the selected pixel value (if allowed by the corresponding opacity value). If the selected pixel is a non-facial skin pixel, the coverage value will prevent the blending of the modifier color value with the selected pixel value of the input image.

In some implementations, a modifier image need not be determined and stored in block 412, and the color value obtained for a selected pixel in block 408 (based on the pattern image) can be blended directly with the corresponding input image pixel in block 418 based on the coverage value determined in block 416.

In block 420, the system determines whether there is another pixel of the input image to process for blending. If so, the method returns to block 414 to select another pixel of the input image.

If there are no further pixels to process, the method can return to block 216 of Fig. 2, e.g., to check if additional faces in the input image need to be processed. At such a point in the process, the system has determined the output pixels for the selected face. If there are additional faces, each face can be processed similarly to provide blended output pixels for that face in the output image. If there are no other faces in the image to process, then other pixels of the input image that were not processed by blocks 206-216 can, for example, be copied to the output image. In some implementations, an output image having a first face modified with a pattern image will become the input to a next iteration of method 200. In some cases, some areas of the input image may be modified twice, e.g., if two faces are very close to each other and their skin masks bleed into one another. However, this is not likely due to the accuracy of the face masks.

Luminance noise that may be present in the input image may also be present in the blended output image, e.g., since the blending preserves most of the luminance values of the input image.

Various blocks and operations of these methods can be performed in a different order than shown and/or at least partially simultaneously, or for multiple or different times than shown, etc. In one example, blocks 410, 412, and 414 can be omitted, and a coverage value in block 416 and blending in block 418 can be determined for the same pixel selected in block 402.

Various applications can use one or more features described herein. For example, the pattern image can be a design that is "painted" onto a face depicted in the input image as if it were face paint. The pattern can be any of a variety of designs, including a national flag or other flag design or team logo (e.g., for a sports game), a set of colors, a motto, symbol, picture, etc. In some applications, the pattern image can include features to make the target face appear as if it has a mask on, e.g., like Halloween face paint.

Some implementations can use one or more described features to apply pattern image adjustments to faces in other contexts. For example, the pattern image can provide an appearance of facial makeup or beauty products (e.g., eye shadow, blush, lipstick, etc.) applied to a face in an image. The pattern image can cover or reduce the appearance of skin marks or other facial features or objects on a person's face. For example, the pattern image can be defined to be similar to the skin tones it is intended to blend so as to reduce the visibility of particular skin marks or features such as facial hair, glasses, etc.

One or more features described herein can be used to display face patterns on faces appearing in video content. For example, faces in each video frame of a video sequence can have pattern image(s) applied as described herein. Video applications can include a real-time video conference. Social networking services and other network services can allow users to apply patterns to faces of themselves and other users depicted in image content displayed by the services. One or more features described herein can be used to apply patterns to faces of avatars, e.g., faces or characters in simulated environments, computer games, or other environments.

Some implementations can apply a pattern image to a face in an image, and then that face can be cropped and placed in a different image. For example, face paint similar to face paint worn by a character in a movie or other entertainment content can be applied to a face in a pattern image and then that face can be placed in a poster having the theme of the movie or other entertainment content.

Some implementations can use one or more described features to apply patterns to other recognized or detected image features besides faces. For example, other skin portions of a body can be detected and have a pattern applied similarly as faces described herein, e.g., to simulate a tattoo applied on an arm, etc. A pattern image can also be applied to other objects that can be recognized in an image using image recognition techniques, including monuments, landscape features (buildings, roads, trees, lakes, etc.). For example, a pattern image can be applied to a side of a building, a road surface, a cloud, an animal, or other recognized image features. In some cases, an image feature mask can be used to indicate which pixels of the input image are included in the desired image feature to modify with the pattern image.

Figs. 6A-6I are diagrammatic illustrations related to examples of applying a pattern image to an input image using one or more features described herein. These figures show line drawings of faces and patterns that represent images having pixels, e.g., photographs and similar types of images. In some implementations, the images shown in Fig. 6A-6I can be displayed in a user interface that in turn can be displayed on a display device. In other implementations, images can be processed as described herein without being displayed.

Fig. 6A shows an example image of a reference face image 600 that can be used to define a pattern image for a face. For example, reference face 602 can provide reference dimensions for a face on which to create a pattern image that will be fit to any detected face in an input image. Reference face 602 includes facial landmark points 603 associated with the facial landmark features of the reference face 602.

In this example, a user has created a star pattern 604 on or over the reference face 602. For example, the user can use functions of an image editing program to draw or paint the star pattern over the reference face 602. The user can select, copy, and/or paste the star pattern 604 from a different image or other source data. Some implementations can automatically create a pattern in a pattern file such as the star pattern 604 without human intervention, e.g., based on user preferences or other user data. For example, the system can copy the star pattern from another image determined to be rated highly by a user. A system can select a pattern file from multiple available stored pattern files. In some examples, the star pattern includes a bold outline of a particular color with the inside of the star being empty. In some implementations, the star pattern can be filled with a color.

Fig. 6B shows an example of a pattern image 606 including the star pattern 604 created on the reference face 602 shown in Fig. 6A. Pattern image 606 includes the star pattern 604 without the reference face 602. In this figure, a face outline 605 derived from the reference face 602 indicates the placement of the star pattern 604 within the pattern image for explanatory purposes. The pattern image 606 can include the facial landmark points 603 from the reference image 600.

Fig. 6C shows an example of an input image 616 selected to be processed by the pattern image 606 of Fig. 6B. Image 616 depicts a face 618 to be applied with the pattern image 606 determined in Fig. 6B. For example, a user may have selected the input image 616 from a variety of available images.

Fig. 6D shows an example of facial landmark points 620 detected for the face 618 of the input image 616. The facial landmark points 620 can be determined based on the detected landmark features of the face, including eyes, eyebrows, nose, and mouth. In this example, landmark points 620 are determined for the endpoints of the eyebrows, the center of the eyes, the left and right base of the nose, the tip of nose, and center base of nose.

Fig. 6E shows an example of sample areas 624 determined for the face 618 of input image 616. In this example, the sample areas 624 are areas of the cheeks of the face 618, defined by triangles drawn between the eye, the outside mouth corner, and the tip of the nose on each side of the face. Sample color values can be obtained from the sample areas 624, can be sorted and filtered of outliers, and can used to determine a face mask as described herein.

Fig. 6F shows an example of a falloff region 630 that can be used in the determination of a face mask for using applying the pattern image to the face 618. As described above with respect to block 310, the falloff region can be used to designate non-face pixels at particular distances away from the center of the face. In this example, an elliptically-shaped falloff region 630 is used, where the region falls off gradually at approximately the boundaries of the face.

Fig. 6G shows an example of a face mask 640 resulting from utilizing color distances to one or more of the color samples taken from face areas 624 of Fig. 6E and the falloff region 630 of Fig. 6F. In this visualization of the face mask 640, lighter regions are designated as face skin pixels and darker regions are designated as non-face skin pixels (e.g., disregarding the white lines of the drawing demonstrating the position of the face, hair, neck, etc. of the subject of the image). Grey pixels are designated as partial facial skin pixels. A specular highlight 642 on the forehead of the face 618 is darker in and thus not designated as facial skin, since the method desires to maintain this highlight in its approximate original color of the input image and to not modify it greatly with the pattern image colors. A gradual falloff of facial skin pixels is shown as a fade from lighter shades to darker shades at the neck and ear of the subject, e.g., due to the elliptical falloff region 630 described above.

Fig. 6H shows a diagrammatic example of a mapping of facial landmark points 603 of the pattern image 606 to the facial landmark points 620 of the face 618 in the input image 616. The reference face 602 is not included in the pattern image 606 but is shown in Fig. 6H in dotted lines 605 to indicate the facial landmark points and the star pattern 604 of the pattern image with reference to these landmark points. Occluded landmark points in the input image 616 are correspondingly discarded or ignored in the alignment of pattern image 606 to the face 618. For example, the head in input image 616 is slightly facing downwards, such that the landmark point 644 at the bottom of the nose is occluded. Thus, the corresponding landmark point 648 in the pattern image is discarded and not used in the alignment (the other nose landmark points are used). The point-to-point correspondence between these facial landmark points and warping of the pattern image as described above allows the pattern image to be fit to the face 618.

Fig. 6I shows an example of a final output image 650 after the pattern image 606 has been warped to the face 618 of the input image 616 and color values derived from the pattern image 606 have been blended with pixel values of the input image 616 at particular areas of the face 618. In this example, the star pattern 604 has been blended with the input pixels around the right eye of the face 618. The other portions of the input image 618 have not been blended with any pattern color values since the pattern image does not provide color values in those areas, and the skin mask 640 prevents application of pattern colors outside the detected facial skin pixels for this face 618.

Fig. 7 shows another example of a pattern image 710. Pattern image 710 is designed to cover an entire face (in frontal view) with a pattern derived from a national flag. The pattern image 710 has borders roughly corresponding to the reference face image on which it was based (not shown). In this example, the pattern image 710 includes blank areas 712 corresponding to the eyes and mouth of a face where the pattern is not intended to be applied. Pattern image 710 also includes facial landmark points (not shown) defining the extents of facial landmark features such as the eyes, mouth, and nose similarly as pattern image 606.

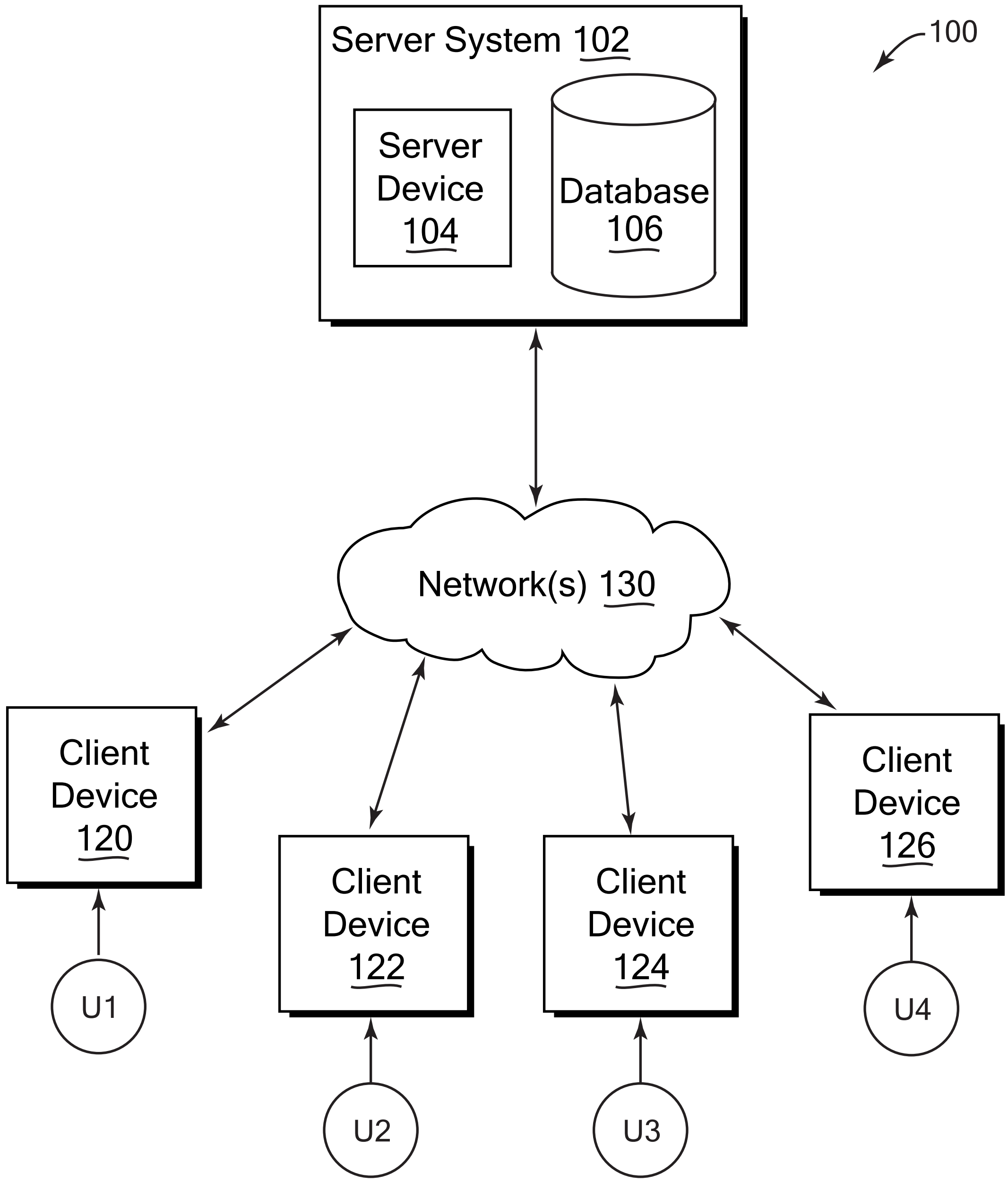
100
Server System 102
Server Device 104
Database 106
Network(s) 130
Client Device 120
Client Device 122
Client Device 124
Client Device 126
U1
U2
U3
U4

FIG. 1

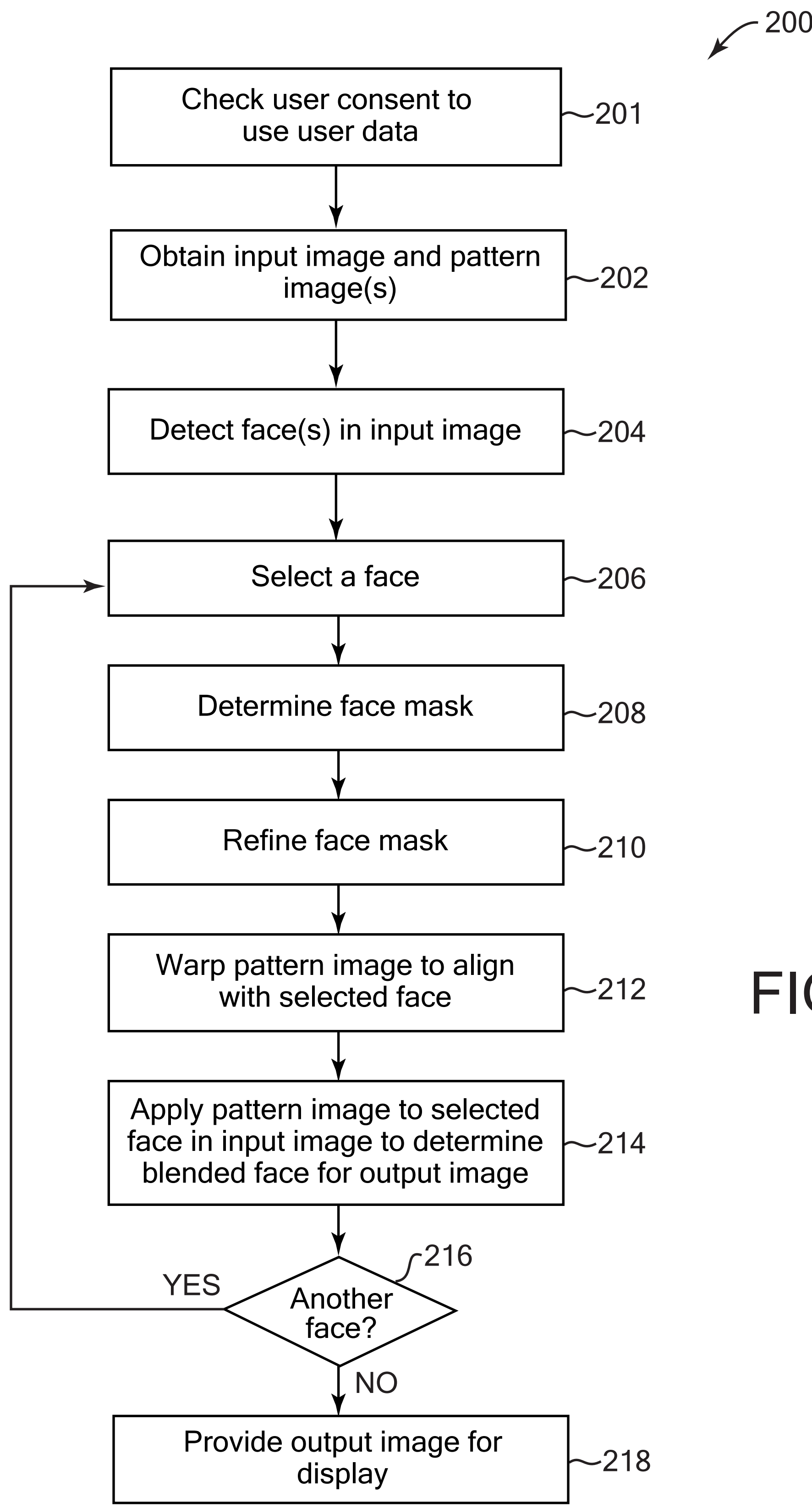
200
Check user consent to use user data
201
Obtain input image and pattern image(s)
202
Detect face(s) in input image
204
Select a face
206
Determine face mask
208
Refine face mask
210
Warp pattern image to align with selected face
212
Apply pattern image to selected face in input image to determine blended face for output image
214
216
YES
Another face?
NO
Provide output image for display
218

FIG. 2

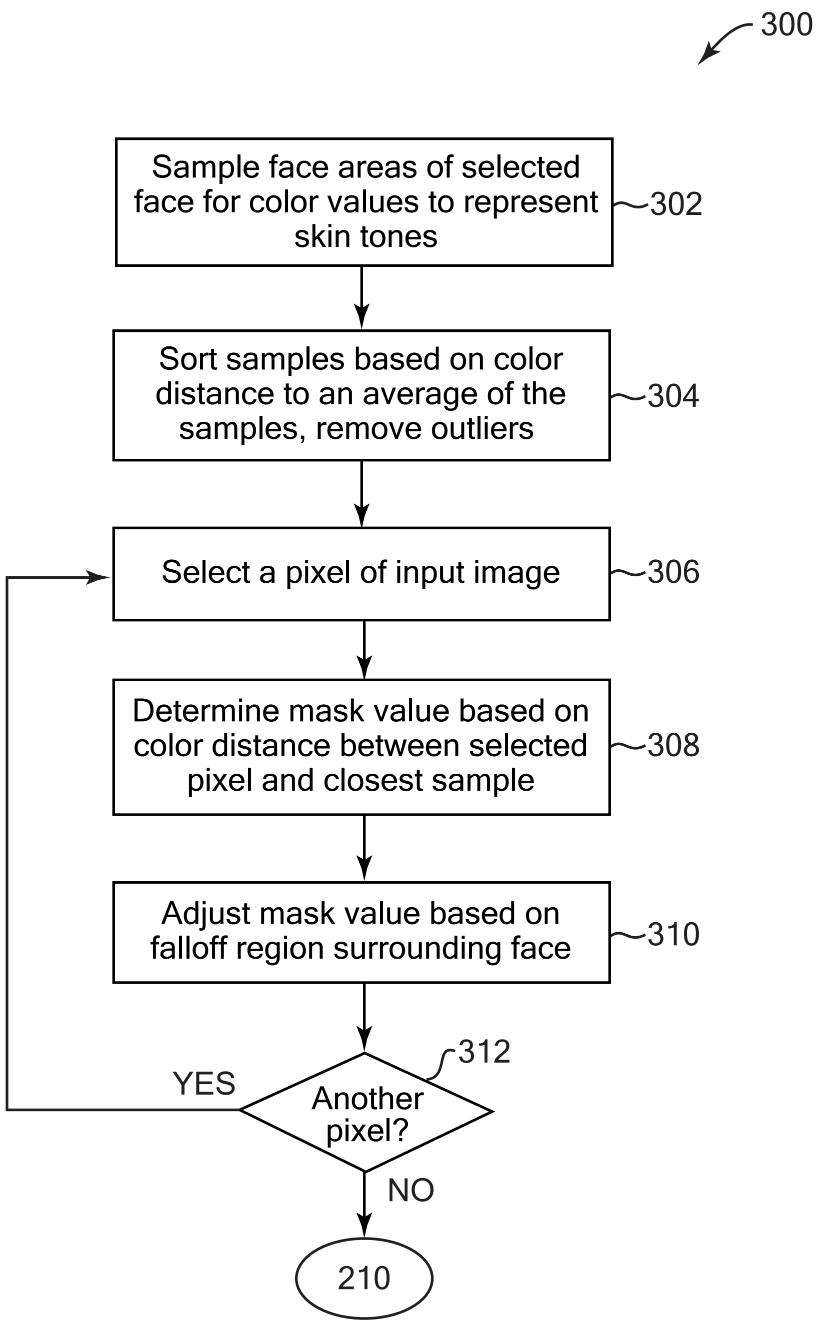
300
Sample face areas of selected face for color values to represent skin tones
302
Sort samples based on color distance to an average of the samples, remove outliers
304
Select a pixel of input image
306
Determine mask value based on color distance between selected pixel and closest sample
308
Adjust mask value based on falloff region surrounding face
310
312
YES
Another pixel?
NO
210

FIG. 3

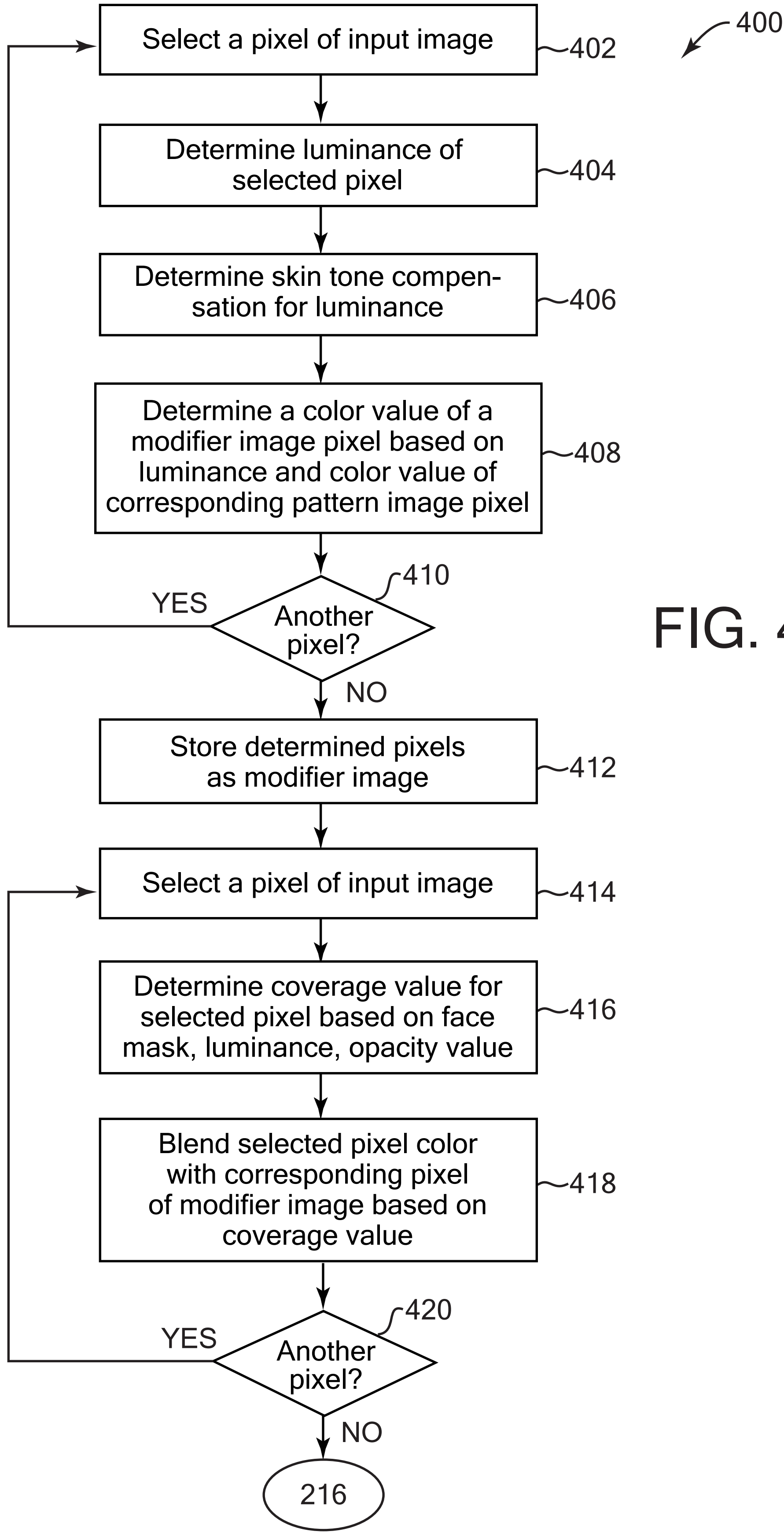
400
Select a pixel of input image
402
Determine luminance of selected pixel
404
Determine skin tone compensation for luminance
406
Determine a color value of a modifier image pixel based on luminance and color value of corresponding pattern image pixel
408
Another pixel?
410
YES
NO
Store determined pixels as modifier image
412
Select a pixel of input image
414
Determine coverage value for selected pixel based on face mask, luminance, opacity value
416
Blend selected pixel color with corresponding pixel of modifier image based on coverage value
418
Another pixel?
420
YES
NO
216

FIG. 4

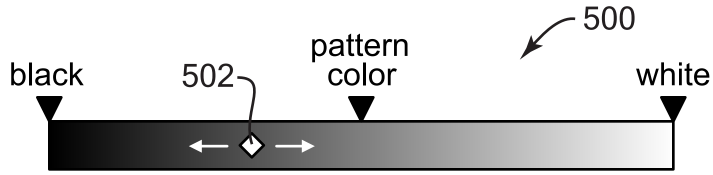

FIG. 5

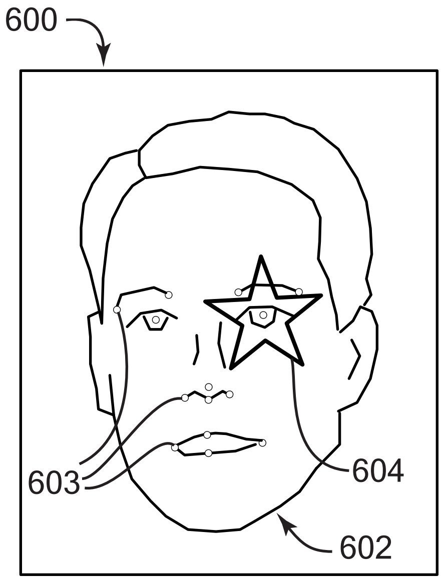

FIG. 6A

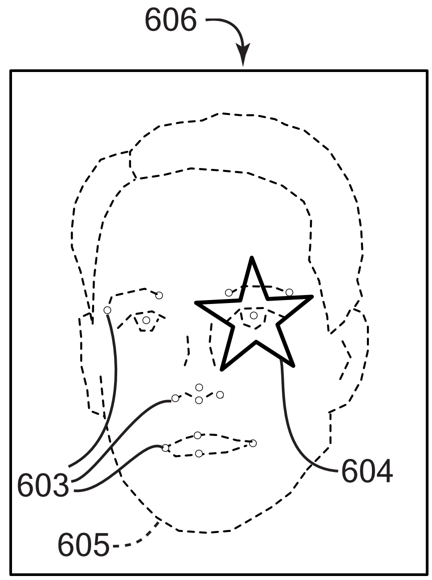

FIG. 6B

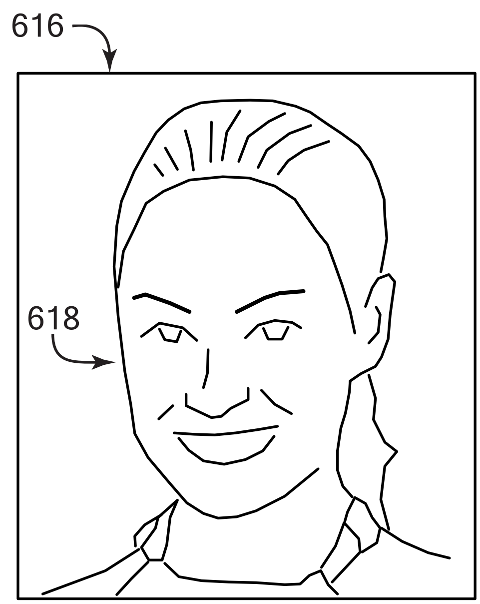

FIG. 6C

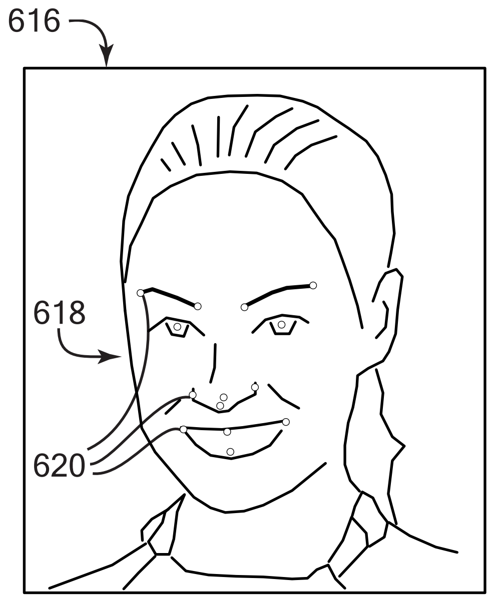

FIG. 6D

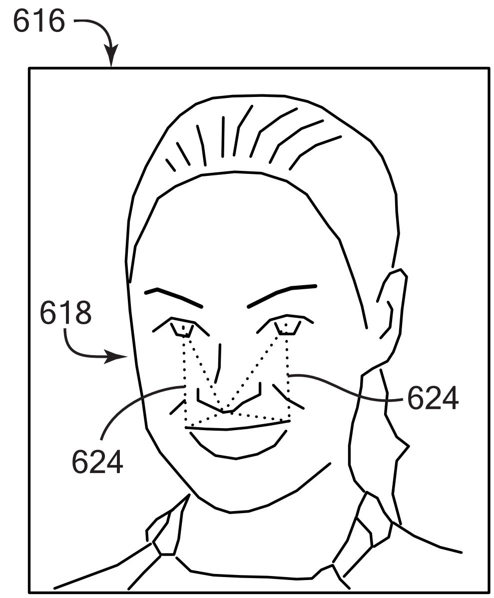

FIG. 6E

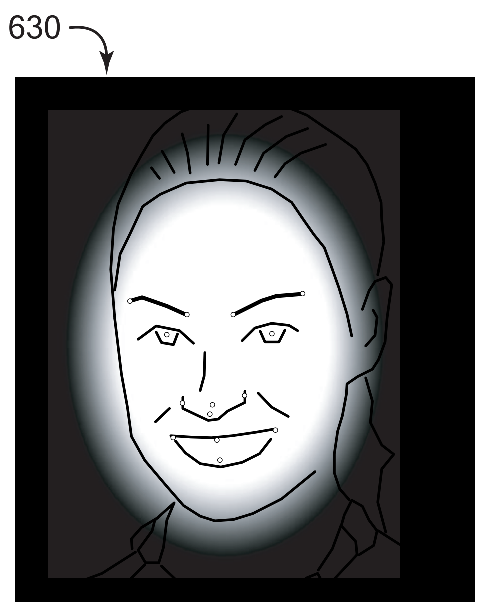

FIG. 6F

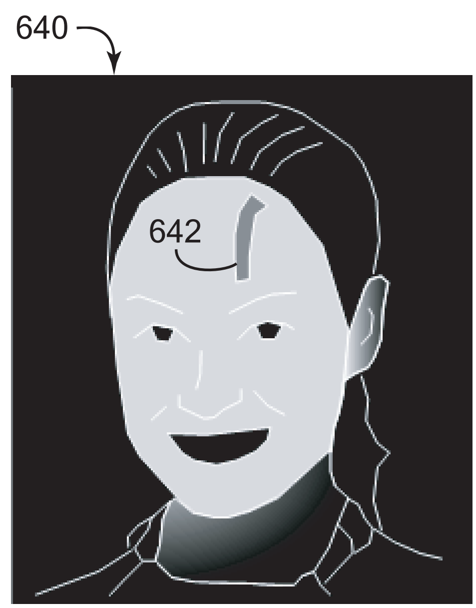

FIG. 6G

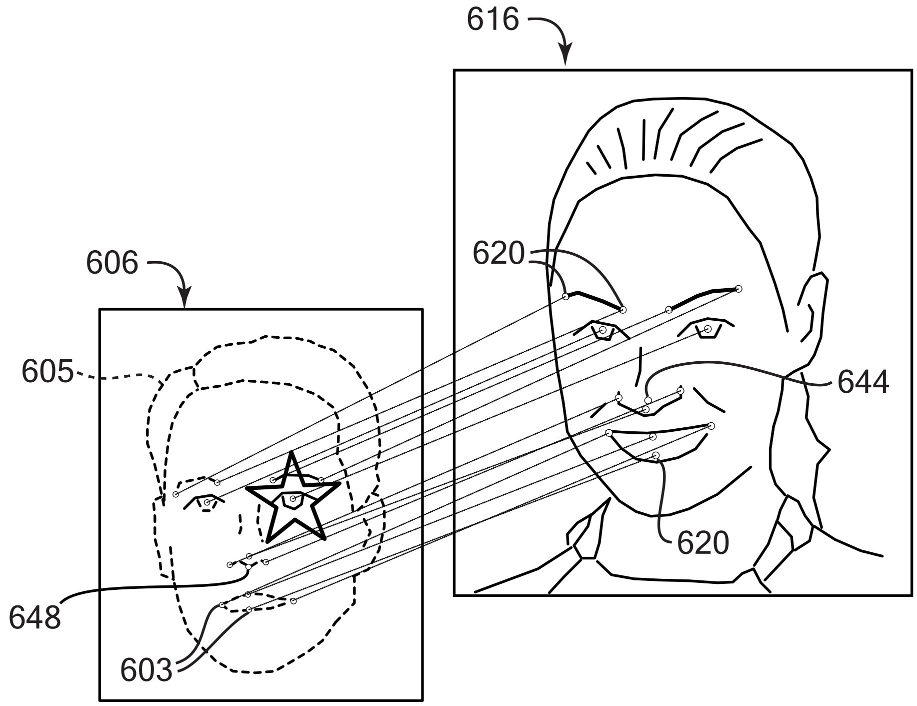

FIG. 6H

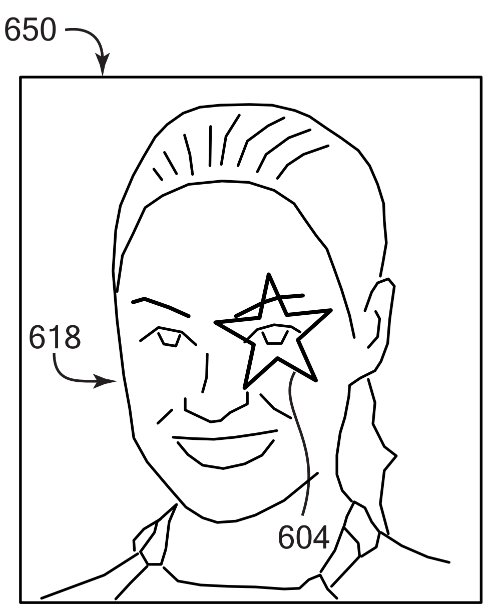

FIG. 6I

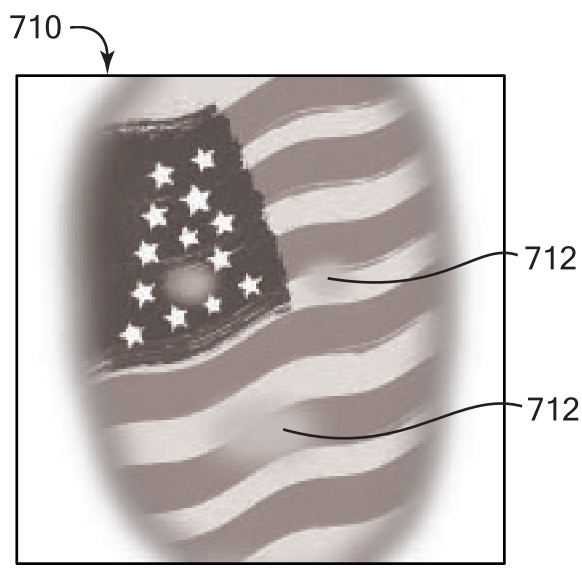

FIG. 7