(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
22 May 2020 (22.05.2020)

WIPO | PCT

(10) International Publication Number
WO 2020/101246 A1

(51) International Patent Classification:
G06T 7/10 (2017.01) *G06T 5/20* (2006.01)
G06T 5/00 (2006.01) *G06N 3/02* (2006.01)

(21) International Application Number:
PCT/KR2019/014916

(22) International Filing Date:
05 November 2019 (05.11.2019)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
2018139928 13 November 2018 (13.11.2018) RU
2019104710 20 February 2019 (20.02.2019) RU

(71) Applicant: **SAMSUNG ELECTRONICS CO., LTD.** [KR/KR]; 129, Samsung-ro, Yeongtong-gu, Suwon-si, Gyeonggi-do 16677 (KR).

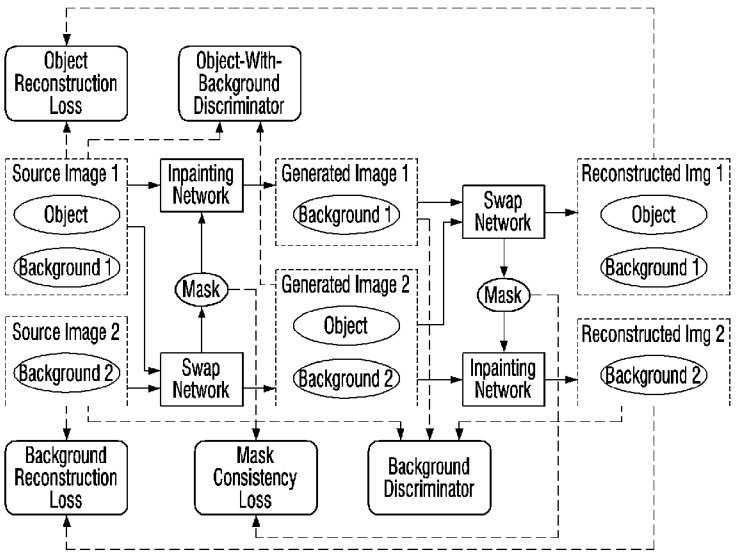
(72) Inventors: **OSTYAKOV, Pavel Aleksandrovich**; office #1500, 12 Dvintsev street, korpus 1, Moscow, 127018 (RU). **SUVOROV, Roman Evgenievich**; office #1500, 12 Dvintsev street, korpus 1, Moscow, 127018 (RU). **LOGACHEVA, Elizaveta Mikhailovna**; office #1500, 12 Dvintsev street, korpus 1, Moscow, 127018 (RU). **KHOMENKO, Oleg Igorevich**; office #1500, 12 Dvintsev street, korpus 1, Moscow, 127018 (RU). **NIKOLENKO, Sergey Igorevich**; office #1500, 12 Dvintsev street, korpus 1, Moscow, 127018 (RU).

(74) Agent: **KIM, Tae-hun** et al.; 9th Floor, Shinduk Bldg., 343, Gangnam-daero, Seocho-gu, Seoul 06626 (KR).

(54) Title: JOINT UNSUPERVISED OBJECT SEGMENTATION AND INPAINTING

$\hat{y}$ (I) $\hat{x}$ (II)

[Fig. 1]



(57) Abstract: The invention relates to implementation of image processing functions associated with finding the boundaries of objects, removing objects from an image, inserting objects into an image, creating new images from a combination of existing images. Proposed is a method for automated image processing and a computing system for performing automated image processing, comprising: first neural network for forming a coarse image z by segmenting an object O from an original image x containing the object O and background B_x by a segmentation mask, and, using the mask, cutting off the segmented object O from the image x and pasting it onto an image y containing only background B_y, second neural network for constructing an enhanced version of an image (Image I) with pasted segmented object O by enhancing coarse image z based on the original images x and y and the mask m; third

[Continued on next page]

WO 2020/101246 A1

(81) **Designated States** *(unless otherwise indicated, for every kind of national protection available)*: AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BN, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DJ, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IR, IS, JO, JP, KE, KG, KH, KN, KP, KR, KW, KZ, LA, LC, LK, LR, LS, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PA, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SA, SC, SD, SE, SG, SK, SL, SM, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW.

(84) **Designated States** *(unless otherwise indicated, for every kind of regional protection available)*: ARIPO (BW, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SD, SL, ST, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, RU, TJ, TM), European (AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK, SM, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, KM, ML, MR, NE, SN, TD, TG).

Published:

— *with international search report (Art. 21(3))*

neural network, for restoring the background-only image (Image II) without removed segmented object O by inpainting image obtained by zeroing out pixels of image x using the mask m; wherein the first, second and third neural networks are combined into common architecture of neural network for sequential performing segmentation, enhancing and inpainting and for simultaneously learning, wherein the common architecture of neural network accepts the images and outputs processed images of the same dimensions.

Description

Title of Invention: JOINT UNSUPERVISED OBJECT SEGMENTATION AND INPAINTING

Technical Field

[1] An invention relates to implementation of image processing functions associated with finding the boundaries of objects, removing objects from an image, inserting objects into an image, creating new images from a combination of existing.

Background Art

[2] Unsupervised and weakly supervised object segmentation. In [18] authors propose a GAN-based [30] technique to generate object segmentation masks from bounding boxes. Their training pipeline consists of taking two crops of the same image: one with object and one without any object. Objects are detected using Faster R-CNN [19]. Then they train a GAN to produce a segmentation mask so that these two crops merged with that mask result into a plausible image. Authors use a combination of adversarial loss, existence loss (which verifies that an object is present on an image) and cut loss (which verifies that no object part left after an object has been cut). They experiment with only some classes from Cityscapes [5] and all classes from MS COCO [14] datasets. Authors report that their approach achieves higher mean intersection-over-union values than classic GrabCut [21] algorithm and recent Simple-Does-It [12]. That approach requires a pretrained Faster R-CNN and a special policy for foreground and background patch selection. It also experiences difficulties with properly segmenting some object classes (e.g. kite, giraffe etc). Their approach also works well only with small resolution images (28 X 28).

[3] In [23] authors propose an annotation-free framework to learn segmentation network for homogeneous objects. They use an adaptive synthetic data generation process to create a training dataset.

[4] While being traditionally tackled with superpixel clustering, unsupervised image segmentation recently has been addressed with deep learning [9]. In the latter paper authors propose to maximize information between two clustered vectors obtained by fully convolutional network from nearby patches of the same image. A similar technique, but constrained with reconstruction loss, has been proposed in [24]. Authors describe W-Net (autoencoder with U-Netlike encoder and decoder), which tries to cluster pixels at inner layer and then reconstruct image from pixel clusters. Their segmentation result is unaware of object classes.

[5] Visual grounding. Methods for visual grounding aim on unsupervised or weakly supervised matching of freeform text queries and regions of images. Usually super-vision

takes form of pairs of (Image; Caption). Model performance is usually measured as intersection-over-union against ground truth labels. The most popular datasets are Visual Genome [13], Flickr30k [17], Refer-It-Game [11] and MS COCO [14]. General approach to grounding consists in predicting if the given caption and image corresponds to each other. Negative samples are obtained by shuffling captions and images independently. Text-image attention is the core feature of most models for visual grounding. [28]. Obviously, using more fine-grained supervision (e.g. region-level annotations instead of image-level) allows to achieve higher scores [29].

[6] Trimap generation. Trimap generation is a problem of producing a segmentation of an image into three classes: foreground, background and unknown (transparent foreground). Most algorithms require human intervention to propose trimap, but recently superpixel and clustering based approaches have been proposed for automatic trimap generation [7]. However, their approach requires executing multiple optimization steps for each image. Deep learning is used to produce alpha matting mask given image and a trimap [26]. There is also some work on video matting and background substitution in video [8]. They use per-frame superpixel segmentation and then optimize energy in conditional random field of Gaussian mixture models to separate foreground and background frame-by-frame.

[7] Generative adversarial networks. In the latest years, GANs [6] are probably the most frequently used approach to train a generative model. Yet powerful, they prone to unstable training process and inconsistent performance on higher resolution images. A more recently proposed approach, CycleGAN [30] trains two GANs together to establish bidirectional mapping between two domains. Their approach offers much greater stability and consistency. On contrary, it requires the dataset to visualize a kind of invertible operation. A plenty of modifications and applications to CycleGAN have been published, including semantic image manipulation [22], domain adaptation [2], unsupervised image-to-image translation [15], multi-domain translation [3] and many others. There is also a problem that such a mapping between domains may be ambiguous. BicycleGAN [31] and augmented CycleGAN [1] address that problem by requiring that mapping must preserve latent representations.

[8] In that paper we base on ideas of Cut&Paste [18] and CycleGAN [6] and propose a novel architecture and pipeline, which addresses a different problem (background swapping) and achieve better results on unsupervised object segmentation, inpainting and image blending.

Disclosure of Invention

Technical Problem

[9] -

Solution to Problem

[10] The present invention presents a novel approach to visual understanding by simultaneously learning to segment object masks and remove objects from background (aka cut and paste).

[11] Proposed is computing system for performing automated image processing, comprising: first neural network for forming a coarse image z by segmenting an object O from an original image x containing the object O and background B_x by a segmentation mask, and, using the mask, cutting off the segmented object O from the image x and pasting it onto an image y containing only background B_y, second neural network for constructing an enhanced version of an image $\hat{y}$ with pasted segmented object O by enhancing coarse image z based on the original images x and y and the mask m; third neural network, for restoring the background-only image $\hat{x}$ without removed segmented object O by inpainting image obtained by zeroing out pixels of image x using the mask m; wherein the first, second and third neural networks are combined into common architecture of neural network for sequential performing segmentation, enhancing and inpainting and for simultaneously learning, wherein the common architecture of neural network accepts the images and outputs processed images of the same dimensions. At that the first, second and third neural networks are generators which create the images $\hat{x}$ and $\hat{y}$ and convert them. The system further comprising two neural networks configured as discriminators, which estimate plausibility of the images. At that, the first discriminator is a background discriminator that attempts to distinguish between a reference real background image and inpainted background image; a second discriminator is an object discriminator that attempts to distinguish between a reference real object O image and enhanced object O image. At that, the first and second neural networks constitute a swap network. At that, the swap network is configured to train end-to-end with loss functions for constructing enhanced version of the image $\hat{y}$ with pasted the segmented object O. At that, one of loss functions is an object reconstruction function for ensuring consistency and training stability, and is implemented as the mean absolute difference between the image x and image $\hat{x}$. At that one of loss functions is an adversarial object function for increasing the plausibility of the image $\hat{y}$, and is implemented with a dedicated dis-

criminator network. At that one of loss functions is a mask consistency function for making the first network being invariant against the background, and is implemented as the mean absolute distance between the mask extracted from image x and the mask extracted from image $\hat{y}$. One of loss functions is an object enhancement identity function for forcing the second network to produce images closer to real images, and is the mean absolute distance between $Genh(x)$ and $x\ self$. At that, one of loss functions is a background identity function for ensuring that the common architecture does not do anything to an image that does not contain objects. At that one of loss functions is an overall loss function that is a linear combination of an object reconstruction function, an adversarial object function, a mask consistency function, an object enhancement identity function, a background identity function. At that, the segmentation mask is predicted by the first network in view of image x.

[12] Proposed is method for automated image processing by the following steps: using first neural network for forming a coarse image z by segmenting an object O from an original image x containing the object O and background B_x by a segmentation mask, and, using the mask, cutting off the segmented object O from the image x and pasting it onto an image y containing only background B_y, using second neural network for constructing an enhanced version of an image $\hat{y}$ with pasted segmented object O by enhancing coarse image z based on the original images x and y and the mask m; using third neural network for restoring the background-only image $\hat{x}$ without removed segmented object O by inpainting image obtained by zeroing out pixels of image x using the mask m; outputting the images $\hat{x}$ and $\hat{y}$ of the same dimensions. At that, first, second and third neural networks are generators which create the images $\hat{x}$ and $\hat{y}$ and convert their. The method further comprising two neural networks configured as discriminators, which estimate plausibility of the images. At that the first discriminator is a background discriminator that attempts to distinguish between a reference real background image and inpainted background image; a second discriminator is an object discriminator that attempts to distinguish between a reference real object O image and enhanced object O image. At that, first and second neural networks constitute swap network. At that, the swap network is configured to train end-to-end with loss functions for constructing enhanced version of the image $\hat{y}$

with pasted the segmented object O. At that, one of loss functions is an object reconstruction function for ensuring consistency and training stability, and is implemented as the mean absolute difference between the image x and image $\hat{x}$. At that, one of loss functions is an adversarial object function for increasing the plausibility of the image, and is implemented with a dedicated discriminator network. At that one of loss functions is a mask consistency function for making the first network being invariant against the background, and is implemented as the mean absolute distance between the mask extracted from image x and the mask extracted from image $\hat{y}$. At that, one of loss functions is an object enhancement identity function for enhancing the second network to produce images closer to real images, and is the mean absolute distance between $Genh(x)$ and x *self*. At that, one of loss functions is a background identity function for ensuring that the common architecture does not do anything to an image that does not contain objects. At that, one of loss functions is an overall loss function that is a linear combination of an object reconstruction function, an adversarial object function, a mask consistency function, an object enhancement identity function, a background identity function. At that, the segmentation mask is predicted by the first network in view of image x.

Advantageous Effects of Invention

[13] -

Brief Description of Drawings

[14] The above and/or other aspects will be more apparent by describing exemplary embodiments with reference to the accompanying drawings, in which:

[15] Figure 1 an architecture of the neural network, data preparation scheme and setting its parameters. A high-level overview of the SEIGAN (Segment-Enhance-Inpainting) pipeline for joint segmentation and inpainting: the swap operation is executed twice and optimized to reproduce original images. Ellipses denote objects and data; solid rectangles, neural networks; rounded rectangles, loss functions; solid lines show the data flows, and dashed lines indicate the flow of values to loss functions.

[16] Figure 2. Architecture of the swap network (from figure 1) that cuts the object from one image and pastes it onto another.

[17] Figure 3. Examples of images and masks generated by our model.

[18] Figure 4. Architecture of residual network used for inpainting and/or segmentation networks.

[19] Figure 5. Architecture of U-Network used for segmentation and refinement networks.

Best Mode for Carrying out the Invention

[20] -

Mode for the Invention

[21] The proposed invention can be useful hardware comprising software products and devices that perform automatic or automated image processing, including:

[22] - graphic editor;

[23] - creative applications for creating graphic content;

[24] - hardware systems (wearable devices, smartphones, robots) for which you want to find objects in images;

[25] - augmented reality modeling (virtual / augmented reality);

[26] - to prepare data for setting up methods of machine learning (any industry).

[27] The symbols used in the application materials are explained below.

[28] O - an object, depicted in an image.

[29] B_x - background, depicted in an image x.

[30] B_y - background, depicted in an image y.

[31] $x=<O,B_x>$ - an image, containing object O and background B_x.

[32] $y=<\varnothing,B_y>$ - an image, containing only background B_y (and no object in foreground).

[33] x - a set of all images x.

[34] y - a set of all images y.

[35] $\hat{x}=<\varnothing,\hat{B}_x>$ - an image x with object O removed (so the image contains only background B_x).

[36] $\hat{y}=<\hat{O},\hat{B}_y>$ - an image y with object O pasted.

[37] $\hat{B}_x\approx\hat{B}$, $\hat{B}_y\approx\hat{B}$, and $\hat{O}\approx O$ - transformed (approximate) variants of backgrounds B_x and B_y and object O.

[38] m = Mask(x) - segmentation mask for image x.

[39] $z=m\odot x+(1-m)\odot y$ - a coarse image constructed by blending images x and y with blending mask m.

[40] Gseg, Ginp, Genh - neural networks used as generators for segmentation, inpainting and enhancement.

[41] Dbg, Dobj - neural networks used as discriminators (Dbg classifies images with real backgrounds from inpainting ones, Dobj classifiers images with real objects from images with pasted ones).

[42] Gram(i) - a Gram matrix constructed from a 3D tensor representing features of image

pixels.

[43] VGG(i) - a function to calculate a 3D tensor, which represents features of image pixels.

[44] L, L_{bg}^{disc}, L_{bg}^{rec}, L_{obj}^{rec}, L_{obj}^{disc}, L_{mask}, L_{obj}^{id}, L_{bg}^{id} - optimization criteria used to tune parameters of neural networks.

[45] $\lambda_1,\ldots,\lambda_7$ - non-negative real coefficients used to balance importance of different optimization criteria.

[46] The proposed image processing functions require less detailed control on the part of the person, compared to the existing analogues at the moment.

[47] The proposed solution can be implemented in software, which in turn can be run on any devices with sufficient computing power.

[48] Throughout the paper, we denote images as object background tuples, e.g. $x=<O,B_x>$ means that image x contains object O and background B_x, and $y=<\varnothing,B_y>$ means that image y contains background B_y and no objects.

[49] The main problem that we address in this work can be formulated as follows. Given a dataset of background images $Y=\{<\varnothing,B_y>\}_{y\in Y}$ and a dataset of objects on different backgrounds $X=\{<O,B_x>\}_{x\in X}$ (unpaired, i.e., with no mapping between X and Y), train a model to take an object from an image $x\in X$ and paste it onto a new background defined by an image $y\in Y$, while at the same time deleting it from the original background. In other words, the problem is to transform a pair of images $x=<O,B_x>$ and $y=<\varnothing,B_y>$ into a new pair $\hat{x}=<\varnothing,\hat{B}_x>$ and $\hat{y}=<\hat{O},\hat{B}_y>$, where $\hat{B}_x\approx B$, $\hat{B}_y\approx B$, and $\hat{O}\approx O$, but $\hat{O}\not\approx O$ the object and both backgrounds are changed so that the new images look natural.

[50] This general problem can be decomposed into three subtasks:

[51] - Segmentation: segment the object O from an original image $x=<O,B_x>$ by predicting the segmentation m = Mask(x); given the mask, we can make a coarse blend that simply cuts off the segmented object from x and pastes it onto y: $z=m\odot x+(1-m)\odot y$, where $\odot$ denotes componentwise multiplication. In the process of learning, the parameters of the neural network are adjusted in such a way that, when the image with the object is input, this neural network gives the correct mask on which the object is selected. The user does not participate in this process.

[52] - Enhancement: given the original images x and y, coarse image z, and segmentation

$\hat{y} = \langle \hat{O}, \hat{B}_y \rangle$. -Inpainting: given a segmentation mask m and an image $(1-m)\odot x$ obtained by zeroing out pixels of x according to m, restore the background-only image $\hat{x} = \langle \emptyset, \hat{B}_x \rangle$. Instead of removed segmented object O, a part of the image is filled with the third neural network based on the remaining part of the image and a random signal. During training, the parameters of the third neural network are configured in such a way that, on the basis of this fragmentary information, it produces a plausible background fill. The result is two images $\hat{x}$ and $\hat{y}$. However, the focus is on the image $\hat{y}$, while the image with a blank background is an intermediate result of this algorithm, although it can also be used.

[53] For each of these tasks we can construct a separate neural network that accepts an image or a pair of images and outputs new image or images of the same dimensions. However, our main hypothesis that we explore in this work is that in the absence of large paired and labeled datasets (which is the normal state of affairs in most applications), it is highly beneficial to train all these neural networks together.

[54] Thus, we present our SEIGAN (Segment-Enhance-Inpaint) architecture that combines all three components in a novel and previously unexplored way. In the fig. 1 boxes with dotted outline denote data (images); ellipses denote objects contained in the data; boxes with sharp corners denote subprograms implementing neural networks; boxes with rounded corners denote subprograms which control the process of tuning neural network parameters during the training procedure; lines denote flows of data during training procedure (the fact that an arrow points from one box to another means that the results of the first box are passed as input to the second).We outline the general flow of our architecture on Figure 1; the “swap network” module there combines segmentation and enhancement. Since cut-and-paste is a partially reversible operation, it is natural to organize the training procedure in a way similar to CycleGAN [30]: the swap and inpainting networks are applied twice in order to complete the cycle and be able to use the idempotency property for the loss functions. We denote by $\hat{x}$ and $\hat{y}$ and y the results of the first application, and by $\hat{\hat{x}}$ and $\hat{\hat{y}}$ the results of the second application, moving the object back from $\hat{\hat{x}}$ and $\hat{\hat{y}}$ (see Fig. 1).

[55] The architecture, showed in figure 1, combines five different neural networks, three used as generators, which create an image and convert it, and two as discriminators, which estimate plausibility of the image:

[56] ● G_{seg} is solving the segmentation task: given an image x, it predicts Mask(x), the segmentation mask of the object on the image;

[57] ● G_{inp} is solving the inpainting problem: given m and (1-m)⊙x,predict $\hat{x} = <\emptyset, \hat{B}_x>$;

[58] ● G_{enh} does enhancement: given x, y, and z = m⊙x+(1-m), predict $\hat{y} = <\hat{O}, \hat{B}_y>$;

[59] ● D_{bg} is the background discriminator that attempts to distinguish between real and fake (inpainted) background-only images; its output $D_{bg}(x)$ should be close to 1 if x is real and close to 0 if x is fake;

[60] ● D_{obj} is the object discriminator that does the same for object-on-background images; its output $D_{obj}(x)$ should be close to 1 if x is real and close to 0 if x is fake.

[61] Generators G_{seg} and G_{enh} constitute the so-called "swap network" depicted as a single unit on Fig. 1 and explained in detail on Fig. 2. This figure depicts the architecture of the “swap network” (a box named “Swap network” on Fig. 1) along with a minimal set of other entities needed to describe how the “swap network” is used. Boxes with dotted outline denote data (images); ellipses denote objects contained in the data; boxes with sharp corners denote subprograms implementing neural networks; boxes with rounded corners denote subprograms which control the process of tuning neural network parameters during the training procedure; lines denote flows of data during training procedure (the fact that an arrow points from one box to another means that the results of the first box are passed as input to the second). Segmentation network is a neural network which takes an image and outputs a segmentation mask of the same size. Refinement network takes an image and outputs an improved its version (i.e. with more realistic colors, with artifacts removed, etc.) of the same size.

[62] Compared to [18], the training procedure in SEIGAN has proven to be more stable and able to work in higher resolutions. Furthermore, our architecture allows to address more tasks (inpainting and blending) simultaneously rather than only predicting segmentation masks. As usual in GAN design, the secret sauce of the architecture lies in a good combination of different loss functions. In SEIGAN, we use a combination of adversarial, reconstruction, and regularization losses.

[63] The inpainting network Ginp aims to produce a plausible background $\hat{B}_x$ given a

source image (1-m)⊙x, which represents the original image x with the object subtracted according to segmentation mask m obtained by applying the segmentation network, m=$G_{seg}(x)$; in practice, we fill the pixels of m⊙x with white. Parameters of inpainting networks are optimized during the end-to-end training according to the following loss functions (shown by rounded rectangles on Fig. 1).

[64] The adversarial background loss aims to improve the plausibility of the resulting image. It is implemented with a dedicated discriminator network Dbg. For Dbg, we use the same architecture as in the original CycleGAN [30] except for the number of layers; our experiments have shown that a deeper discriminator works better in our setup. As the loss function Dbg uses the MSE adversarial loss suggested in Least Squares GAN (LSGAN) [16], as in practice it is byfar more stable than other types of GAN loss functions:

[65] $$L_{bg}^{disc}=(1-D_{bg}(y))^2+\frac{1}{2}D_{bg}(\hat{x})^2+\frac{1}{2}D_{bg}(\hat{y})^2,$$

[66] where $y=<\varnothing,B_y>$ is the original background image,

[67] $\hat{x}=<\varnothing,\hat{B}_x>$ is the background image resulting from x after the first swap, and $\hat{\hat{y}}=<\varnothing,\hat{\hat{B}}_y>$ is the background image resulting from $\hat{y}$ after the second swap.

[68] The background reconstruction loss aims to preserve information about the original background Bx. It is implemented using texture loss [25], the mean absolute difference between Gram matrices of feature maps after the first 5 layers of VGG-16 networks:

[69] $$L_{bg}^{rec}=\left|G_{ram}(VGG(y))-G_{ram}(VGG(\hat{\hat{y}}))\right|,$$

[70] where VGG(y) denotes the matrix of features of a pretrained image classification neural network (e.g. VGG but not limited to), and $G_{ram}(A)_{ij}=\sum_k A_{ik}A_{jk}$ is the Gram matrix.

[71] Our choice of loss functions is motivated by the fact that there are plenty of possible plausible reconstructions of the background, so the loss functions must allow for a certain degree of freedom that mean absolute error or mean squared error would not permit but which texture loss does. In our experiments, optimizing MAE or MSE has usually led to the generated image being filled with median or mean pixel values, with no objects or texture. Note that the background reconstruction loss is applied only to y because we do not have the ground truth background for x (see Fig. 1).

[72] Another important remark is that before feeding the image to the inpainting network Ginp, we subtract a part of image according to segmentation mask m, and we do it in a differentiable way, without any thresholding applied to m. Thus, gradients can propagate back through the segmentation mask to the segmentation network Gseg. Joint training of inpainting and segmentation has a regularization effect. First, the inpainting network Ginp wants the mask to be as accurate as possible: if it is too small then Ginp will have to erase the remaining parts of the objects, which is a much order problem, and if it is too large then Ginp will have more empty area to inpainting. Second, Ginp wants the segmentation mask m to be high-contrast (with values close to 0 and 1) even without thresholding: if much of m is low-contrast (close to 0.5) then Ginp will have to learn to remove the "ghost" of the object (again, much harder than just inpainting on empty space), and it will most probably be much easier for the discriminator Dbg to tell that the resulting picture is fake.

[73] Showed in Fig. 3 is an example of data consumed and produced by the proposed method. The meanings of the images, from left to right, top-down:

[74] 1)The leftmost image in the topmost row is a real input image with an object (an example of “Source image 1” on Fig. 1);

[75] 2)2nd image in the topmost row is a real input image without objects (an example of “Source image 2” on Fig. 1);

[76] 3)the mask predicted by the segmentation network given image 1;

[77] 4)a real input image with an object (another example of “Source image 1” on Fig. 1);

[78] 5)a real input image without objects (another example of “Source image 2” on Fig. 1);

[79] 6)The leftmost image in the bottom row is the output of inpainting network with object from image 1 removed by mask on image 3 (an example of “Generated image 1” on Fig. 1);

[80] 7)output of refinement network with object from image 1 pasted onto background from image 2 (an example of “Generated image 2” on Fig. 1);

[81] 8)the mask predicted by the segmentation network given image 4;

[82] 9)output of inpainting network with object from image 4 removed by mask on image 8 (another example of “Generated image 1” on Fig. 1);

[83] 10)output of refinement network with object from image 4 pasted onto background from image 5 (another example of “Generated image 2” on Fig. 1).

[84] For Ginp, we use a neural network consisting of two residual blocks connected sequentially (see Figure 4). We also experimented with ShiftNet [27]. Fig. 4. depicts architecture of ResNet neural network used as “inpainting network” and “segmentation network”. Ellipses denote data; rectangles - layers of neural networks. The overall architecture is present in the left part of the Figure. The right part of the figure contains a

more detailed description of blocks used in the left part. Arrows denote data flow (i.e. output of one block is fed as input to another block). Conv2d denote convolutional layer; BatchNorm2d denote batch normalization layer; ReLU - linear rectification unit; ReflectionPad - padding of pixels with reflection; ConvTranspose2d - deconvolutional layer.

[85] The swap network aims to generate a new image $\hat{y} = <\hat{O}, \hat{B}_y>$; from two original images, $x = <O, B_x>$ with an object O and $y = <\varnothing, B_y>$ with a different background B_y.

[86] The swap network consists of two major steps: segmentation G_{seg} and enhancement G_{enh} (see Fig. 2).

[87] The segmentation network Gseg produces a soft segmentation mask $m = G_{seg}(x)$ from x. With the mask m, we can extract the object O from its source image x and paste it on By to produce a "coarse" version of the target image $z = m \odot x + (1 - m) \odot y$; z is not the end result, though: it lacks anti-aliasing, color or lightning correction, and other improvements. Note that in the ideal case, pasting an object in a natural way might also require a more involved understanding of the target background; e.g., if we want to paste a dog onto a grass field then we should probably put some of the background grass in front of the dog, hiding its paws as they would not be seen behind the grass in reality.

[88] To address this, we introduce the so-called enhancement neural network G_{enh} whose purpose is to generate a "smoother", more natural image

[89] $\hat{y} = <\hat{O}, \hat{B}_y>$ given original images x and y, and segmentation mask m, which lead to the coarse result $z = m \odot x + (1 - m) \odot y = <O, B_y>$. We have experimented with the enhancement network implemented in four different ways:

[90] ● black-box enhancement: $G_{enh}(x,y,m)$ outputs the final improved image;

[91] ● mask enhancement: $G_{enh}(x,y,m)$ outputs a new segmentation mask m' that better fits object O and new back ground B_y together;

[92] ● color enhancement: $G_{enh}(x,y,m)$ outputs per-pixel per-channel multipliers $\gamma \odot z$; the weights γ are regularized to be close to 1 with an additional MSE loss;

[93] ● hybrid enhancement: $G_{enh}(x,y,m)$ outputs both a new mask m' and multipliers γ

[94] In any case, we denote by $G_{enh}(x,y,m)$ the final improved image after all outputs of Genh have been applied to z accordingly.

[95] We train the swap network end-to-end with the following loss functions (shown by rounded rectangles on Fig. 1).

[96] The object reconstruction loss L_{obj}^{rec} aims to ensure consistency and training stability. It is implemented as the mean absolute difference between the source image x = (O, Bx) and $\hat{x} = G_{enh}(\hat{y}, \hat{x}, G_{seg}(\hat{y}))$:

[97] $$L_{obj}^{rec} = \left| x - \hat{\hat{x}} \right|,$$

[98] where $\hat{y} = G_{enh}(x, y, G_{seg}(x))$ and

[99] where $\hat{y} = G_{enh}(x, y, G_{seg}(x))$ and $\hat{x} = G_{inp}((1 - G_{seg}(x)) \odot x$, i.e.

[100] $\hat{\hat{x}}$ is the result of applying the swap network to x and y twice.

[101] The adversarial object loss L_{obj}^{disc} aims to increase the plausibility of $\hat{y} = <\hat{O}, \hat{B}_y>$. It is implemented with a dedicated discriminator network Dobj. It also has the side effect of maximizing the area covered by segmentation mask $m=G_{seg}(x)$. We apply this loss to all images with objects: real image x and "fake" images $\hat{y}$ and $\hat{x}$. Again, the discriminator has the same architecture as in CycleGAN [30] except for the number of layers, where we have found that a deeper discriminator works better. We again use the MSK loss inspired by LSGAN [16]:

[102] $$L_{obj}^{disc} = (1 - D_{obj}(x))^2 + \frac{1}{2} D_{obj}(\hat{y})^2 + \frac{1}{2} D_{obj}(\hat{\hat{y}})^2$$

[103] The mask consistency loss aims to make the segmentation network invariant against the background. It is implemented as the mean absolute distance between m = Gseg(x), the mask extracted from x = (O, Bx), and m = Gseg(y), the mask extracted from $\hat{y} = <\hat{O}, \hat{B}_y>$:

[104] $$L_{mask} = \left| G_{seg}(x) - G_{seg}(\hat{y}) \right|$$

[105] The mask is essentially black-white picture of the same size as the picture from which this mask was extracted. White pixels on the mask correspond to the selected

areas of the image (pixels in which the object is depicted in this case), black ones - to the background. Mean absolute distance is the modulus of the difference in pixel values, averaged over all pixels. The mask is re-extracted to make sure that the neural network that extracts the mask responds precisely to the shape of the object, and does not respond to the background behind it (in other words, the masks for the same object must always be the same).

[106] Finally, apart from the loss functions defined above we have used the identity loss, an idea put forward in CycleGAN [30]. We introduce two different instances of identity loss:

[107] ● object enhancement identify loss L_{obj}^{id} brings the result of the enhancement network G_{enh} on real images closer to identify: it is the mean average distance between $G_{enh}(x)$ and x itself:

[108] $$L_{obj}^{id}=|G_{enh(x)}-x|\ ;$$

[109] ● background identify loss L_{bg}^{id} tries to ensure that our cutting and inpainting architecture does not do anything to an image that does not contain objects: for an image $y=<\varnothing, B_y>$ we find a segmentation mask $G_{seg}(y)$, subtract it from y to get $(1-G_{seg}(y))\odot y$, apply inpainting G_{inp} and then minimize the mean average distance between the original y and the result:

[110] $$L_{bg}^{id}=|G_{inp}((1-G_{seg}(y))\odot y)-y|\ .$$

[111] The overall SEIGAN loss function is a linear combination of all loss functions defined above:

[112] $$L=\lambda_1 L_{bg}^{disc}+\lambda_2 L_{bg}^{rec}+\lambda_3 L_{obj}^{disc}+\lambda_4 L_{obj}^{rec}+\lambda_5 L_{mask}+\lambda_6 L_{obj}^{id}+\lambda_7 L_{bg}^{id}$$

[113] with coefficients chosen empirically.

[114] During experiments, we have noticed several interesting effects. First, original images $x=<O, B_x>$ and $y=<\varnothing, B_y>$ might have different scale and aspect ratios before merging. Rescaling them to the same shape with bilinear interpolation would introduce significant differences in low-level textures that would be very easy to identify as fake for the discriminator, thus preventing GAN from convergence.

[115] The authors of [18] faced the same problem and addressed it by a special procedure they use to create training samples: they took foreground and background patches only from the same image to ensure the same scale and aspect ratios, which reduces diversity and makes fewer images suitable for the training set. In our setup this problem is addressed by a separate enhancement network, so we have fewer limitations when finding appropriate training data.

[116] Another interesting effect is the low contrast in segmentation masks when inpainting

is optimized against MAE or MSE reconstruction loss. A low-contrast mask (i.e., m with many values around 0.5 rather than close to 0 or 1) allows information about the object from the original image to "leak through" and facilitate reconstruction. A similar effect has been noticed before by other researchers, and in the CycleGAN architecture it has even been used for steganography [4]. We first addressed this issue by converting the soft segmentation mask to a hard mask by simple thresholding. Later we found that optimizing inpainting against the texture loss L_{bg}^{rec} is a more elegant solution that leads to better results than thresholding.

[117] For the segmentation network Gseg, we used the architecture from CycleGAN [30], which itself is an adaptation of the architecture from [10]. For better performance, we replaced ConvTranspose layers with bilinear upsampling. Also, after the final layer of the network we used the logistic sigmoid as the activation function.

[118] For the enhancement network Genh, we used the U-net architecture [20] since it is able both to work with images in high resolution and to make small changes in the source image. This is important for our setup because we do not want to significantly change the image content in the enhancement network but rather just "smooth" the boundaries of the pasted image in a smarter way.

[119] Fig. 5 This figure depicts architecture of U-Net neural network used as “inpainting network” and “refinement network”. Ellipses denote data; rectangles - layers of neural networks. The overall architecture is present in the left part of the Figure. The right part of the figure contains a more detailed description of blocks used in the left part. Arrows denote data flow (i.e. output of one block is fed as input to another block). Conv2d denote convolutional layer; BatchNorm2d denote batch normalization layer; ReLU - linear rectification unit; ReflectionPad - padding of pixels with reflection; ConvTranspose2d - deconvolutional layer.

[120] Data Preparation

[121] Major part of our experiments is carried out on images, publicly available on Flickr under Creative Commons license. We used query "dog" to collect initial image. Then we used a pretrained Faster R-CNN to detect all objects (including dogs) and all regions without any object. Then we constructed two datasets {<O, B_1>} (from regions with dogs) and {(B_2)} (from regions without objects of any class). After data collection, we conducted data filtering procedure in order to get regions of images without any extraneous objects.

[122] The filtering procedure was carried out as follows. First of all, we used a Faster R-CNN [19] (pretrained on MS COCO (14]) to detect all objects on an image. Then, we get crops of the input image according to the following rules:

[123] 1. After rescaling, size of the object is equal to 64x64 and

[124] size of the final crop is equal to 128x128;

[125] 2. The object is located at the center of the crop;

[126] 3. There are no other objects which intersect with the given crop;

[127] 4. The source size of the object on a crop is bigger than 60 (by smallest side) and no bigger than 40 percentage of the whole source image (by longest side).

[128] The foregoing exemplary embodiments are examples and are not to be construed as limiting. In addition, the description of the exemplary embodiments is intended to be illustrative, and not to limit the scope of the claims, and many alternatives, modifications, and variations will be apparent to those skilled in the art.

[129] References

[130] [1] J A. Almahairi. S. Rajeswar, A. Sordoni, R Bachman, and A. Courville. Augmented cyclegan: Learning many-to-many mappings from unpaiied data. arXiv preprint arXiv.l802.10151. 2018.

[131] [2] K. Bousmalis. A. Iipan. P. Wohlhait. Y. Bai. M. Kelcey.

[132] M. Kalakrishnan. L Downs. J. I bar/. P. Pastor. K. Konolige. et al. Using simulation and domain adaptation to improve efficiency of deep robotic grasping. In 2018 IEEE International Conference on Robotics and Automation (ICRA), pages 4243-4250. IEEE, 2018.

[133] [3] Y. Choi. M. Choi. M. Kim. J.-W. Ha. S. Kim. and J. Choo. Stargan: Unified generative adversarial networks for multi-domain image-to-image translation. arXiv preprint. 1711, 2017.

[134] [4] C. Chu. A. Zhmoginov. and M. Sandler. Cyclegan: a master of steganography. arXiv preprint arXiv: 1712.02950,2017.

[135] [5] M. Cordts. M. 6mran. S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson. U. Franke. S. Roth, and B. Schiele. The cityscapes dataset for semantic urban scene understanding. In Proc. of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2016.

[136] [6] I. Goodfellow, J. Pouget-Abadie. M. Miiza, B. Xu, D. Warde-Farley. S. Ozair. A. Courville. and Y. Bengio. Generative adversarial nets. In Advances in neural information processing systems, pages 2672-2680, 2014.

[137] [7] V. Gupta and S. Raman. Automatic trimap generation for image matting. In Signal and Information Processing (ICon-SIP). International Conference on. pages 1-5. IEEE. 2016.

[138] [8] H. Huang. X. Fang. Y. Ye. S. Zhang, and P. L Rosin Prac-tical automatic background substitution for live video. Computational Visual Media, 3(3):273-284.2017.

[139] [9] X. Ji, J. F. Henriques, and A. Vedaldi. Invariant information distillation for unsupervised image segmentation and clustering. arXiv preprint arXiv:1807.06653, 2018.

[140] [10] J. Johnson, A. Alahi, and F. Li. Perceptual losses for real-time style transfer and super-resolution. CoRR, abs/1603.08155, 2016.

[141] [11] S. Kazemzadeh, V. Ordonez, M. Matten, and T. Berg. Referitgame: Referring to objects in photographs of natural scenes. In Proceedings of the 2014 conference on empirical methods in natural language processing {EMNLP), pages 787-798,2014.

[142] [12] A. Khoreva, R. Benenson, J. H. Hosang, M. Hein, and B. Schiele. Simple does it: Weakly supervised instance and semantic segmentation. In CVPR, volume 1, page 3,2017.

[143] [13] R. Krishna, Y. Zhu, O. Groth, J. Johnson, K. Hata, J. Kravitz, S. Chen, Y. Kalantidis, L.-J. Li, D. A. Shamma, M. Bernstein, and L Fei-Fei. Visual genome: Connecting language and vision using crowdsourced dense image annotations. 2016.

[144] [14] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollar, and C. L. Zitnick. Microsoft coco: Common objects in context. In European conference on computer vision, pages 740-755. Springer, 2014.

[145] [15] M.-Y. Liu, T. Breuel, and J. Kautz. Unsupervised image-to image translation networks. In Advances in Neural Information Processing Systems, pages 700-708, 2017.

[146] [16] X. Mao, Q. Li, H. Xie, R. Lau, Z. Wang, and S. P. Smol-

[147] ley. Least squares generative adversarial networks, arxiv preprint. arXiv preprint ArXiv:1611.04076, 2(5), 2016.

[148] [17] B. A. Plummer, L. Wang, C. M. Cervantes, J. C Caicedo, J. Hockenmaier, and S. Lazebnik. Flickr30k entities: Collecting region-to-phrase correspondences for richer image-to-sentence models. In Proceedings of the IEEE international conference on computer vision, pages 2641-2649, 2015.

[149] [18] T. Remez, J. Huang, and M. Brown. Learning to segment via

[150] cut-and-paste. arXiv preprint arXiv:1803.06414, 2018.

[151] [19] S. Ren, K. He, R. Girshick, and J. Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In Advances in neural information processing systems, pages 91-99,2015.

[152] [20] O. Ronneberger, P Fischer, and T. Brox. U-net: Convolutional networks for biomedical image segmentation. CoRR, abs/1505.04597,2015.

[153] [21] C. Rother, V. Kolmogorov, and A. Blake. Grabcut: Interactive foreground extraction using iterated graph cuts. In ACM transactions on graphics (TOG), volume 23, pages 309-314. ACM, 2004.

[154] [22] T.-C. Wang, M.-Y. Liu, J.-Y. Zhu, A. Tao, J. Kautz, and B. Catanzaro. High-resolution image synthesis and semantic manipulation with conditional gans. arXiv preprint arXiv:1711.11585,2017.

[155] [23] Z. Wu, R. Chang, J. Ma, C. Lu, and C.-K. Tang. Annotation-free and one-shot

learning for instance segmentation of homogeneous object clusters. arXiv preprint arXiv:1802.00383, 2018.

[156] [24] X. Xia and B. Kulis. W-net: A deep model for fully unsupervised image segmentation. arXiv preprint arXiv:1711.08506, 2017.

[157] [25] W. Xian, P. Sangkloy, J. Lu, C Fang, F. Yu, and J. Hays. Texturegan: Controlling deep image synthesis with texture patches. CoRR, abs/1706.02823, 2017.

[158] [26] N. Xu, B. L. Price, S. Cohen, and T. S. Huang. Deep image matting. In CVPR, volume 2, page 4, 2017.

[159] [27] Z. Yan, X. Li, M. Li, W. Zuo, and S. Shan. Shift-net: Image inpainting via deep feature rearrangement. arXiv preprint arXiv:1801.09392, 2018.

[160] [28] L. Yu, Z. Lin, X. Shen, J. Yang, X. Lu, M. Bansal, and T. L. Berg. Mattnet: Modular attention network for referring expression comprehension. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2018.

[161] [29] Y. Zhang, L. Yuan, Y. Guo, Z. He, I.-A. Huang, and H. Lee. Discriminative bimodal networks for visual localization and detection with natural language queries. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017.

[162] [30] J. Zhu, T. Park, P. Isola, and A. A. Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. CoRR, abs/1703.10593, 2017.

[163] [31] J.-Y. Zhu, R. Zhang, D. Pathak, T. Darnell, A. A. Efros, O. Wang, and E. Shechtman. Toward multimodal image-to-image translation. In Advances in Neural Information Processing Systems, pages 465-176, 2017.

[164]

Claims

[Claim 1] A computing system for performing automated image processing, comprising:
first neural network
for forming a coarse image z by segmenting an object O from an original image x containing the object O and background B_x by a segmentation mask, and, using the mask, cutting off the segmented object O from the image x and pasting it onto an image y containing only background B_y,
second neural network
for constructing an enhanced version of an image $\hat{x}$ with pasted segmented object O by enhancing coarse image z based on the original images x and y and the mask m;
third neural network,
for restoring the background-only image $\hat{y}$ without removed segmented object O by inpainting image obtained by zeroing out pixels of image x using the mask m;
wherein the first, second and third neural networks are combined into common architecture of neural network for sequential performing segmentation, enhancing and inpainting and for simultaneously learning, wherein the common architecture of neural network accepts the images and outputs processed images of the same dimensions.

[Claim 2] System of claim 1, in which the first, second and third neural networks are generators which create the images $\hat{x}$ and $\hat{y}$ and convert them.

[Claim 3] System of claim 2, further comprising two neural networks configured as discriminators, which estimate plausibility of the images.

[Claim 4] System of claim 3, in which the first discriminator is a background discriminator that attempts to distinguish between a reference real background image and inpainted background image; a second discriminator is an object discriminator that attempts to distinguish between a reference real object O image and enhanced object O image.

[Claim 5] System of any of claims 2-4, in which the first and second neural networks constitute a swap network.

[Claim 6] System of claim 5, in which the swap network is configured to train end-to-end with loss functions for constructing enhanced version of the image $\hat{y}$ with pasted the segmented object O.

[Claim 7] System of claim 6, in which one of loss functions is an object reconstruction function for ensuring consistency and training stability, and is implemented as the mean absolute difference between the image x and image $\hat{x}$.

[Claim 8] System of claim 6, in which one of loss functions is an adversarial object function for increasing the plausibility of the image $\hat{y}$, and is implemented with a dedicated discriminator network.

[Claim 9] System of claim 6, in which one of loss functions is a mask consistency function for making the first network being invariant against the background, and is implemented as the mean absolute distance between the mask extracted from image x and the mask extracted from image $\hat{y}$.

[Claim 10] System of claim 6, in which one of loss functions is an object enhancement identity function for forcing the second network to produce images closer to real images, and is the mean absolute distance between $G_{enh}(x)$ and x itself.

[Claim 11] System of claim 6, in which one of loss functions is a background identity function for ensuring that the common architecture does not do anything to an image that does not contain objects.

[Claim 12] System of claim 6, in which one of loss functions is an overall loss function that is a linear combination of an object reconstruction function, a adversarial object function, a mask consistency function, an object enhancement identity function, a background identity function.

[Claim 13] System of claim 1, in which the segmentation mask is predicted by the first network in view of image x.

[Claim 14] Method for automated image processing by the following steps:
using first neural network
- forming a coarse image z by segmenting an object O from an original image x containing the object O and background B_x by a segmentation mask, and, using the mask, cutting off the segmented object O from the

image x and pasting it onto an image y containing only background $B_{y,}$ using second neural network

- constructing an enhanced version of an image $\hat{y}$ with pasted segmented object O by enhancing coarse image z based on the original images x and y and the mask m;

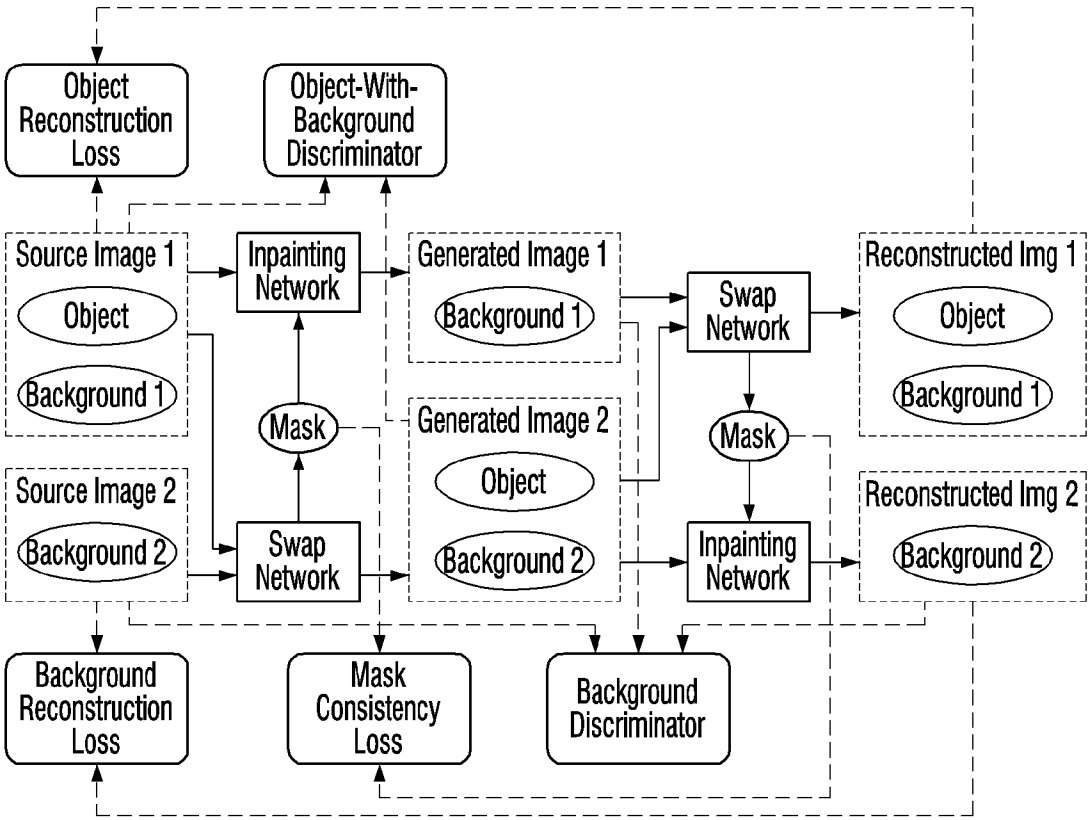
using third neural network

- restoring the background-only image $\hat{x}$ without removed segmented object O by inpainting image obtained by zeroing out pixels of image x using the mask m;

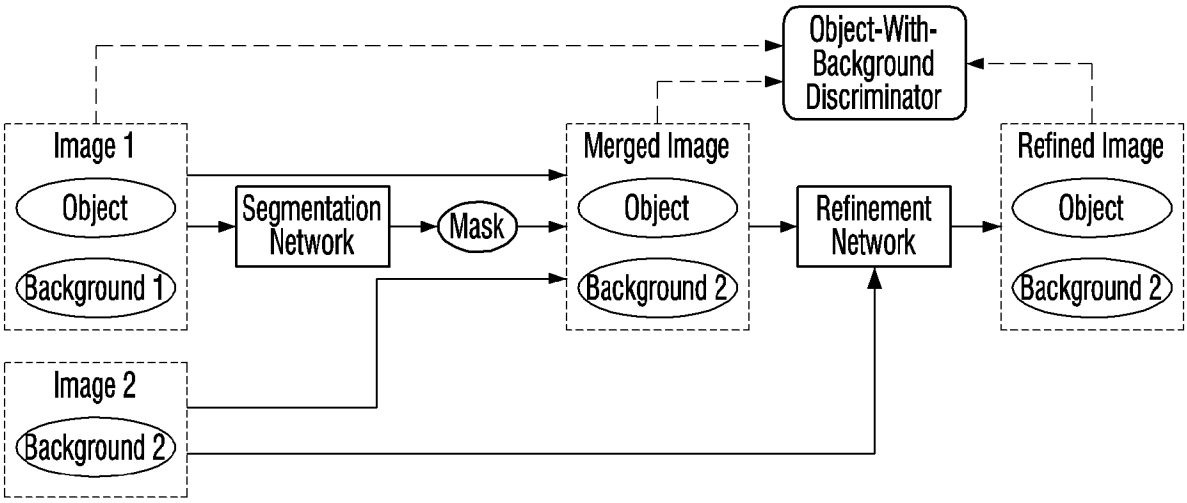
outputting the images $\hat{x}$ and $\hat{y}$ of the same dimensions.

[Claim 15] Method of claim 14, in which first, second and third neural networks are generators which create the images $\hat{x}$ and $\hat{y}$ and convert their.

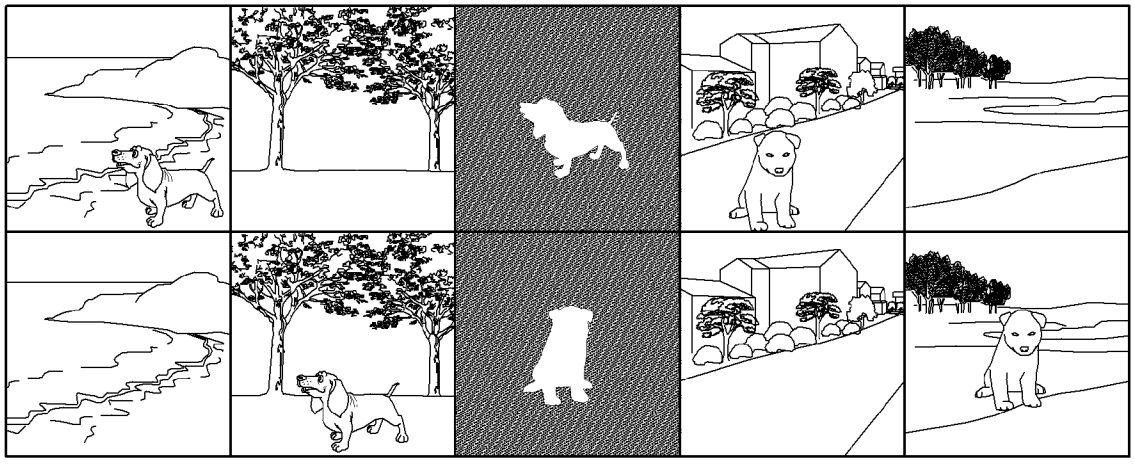
[Fig. 1]



[Fig. 2]



[Fig. 3]



[Fig. 4]

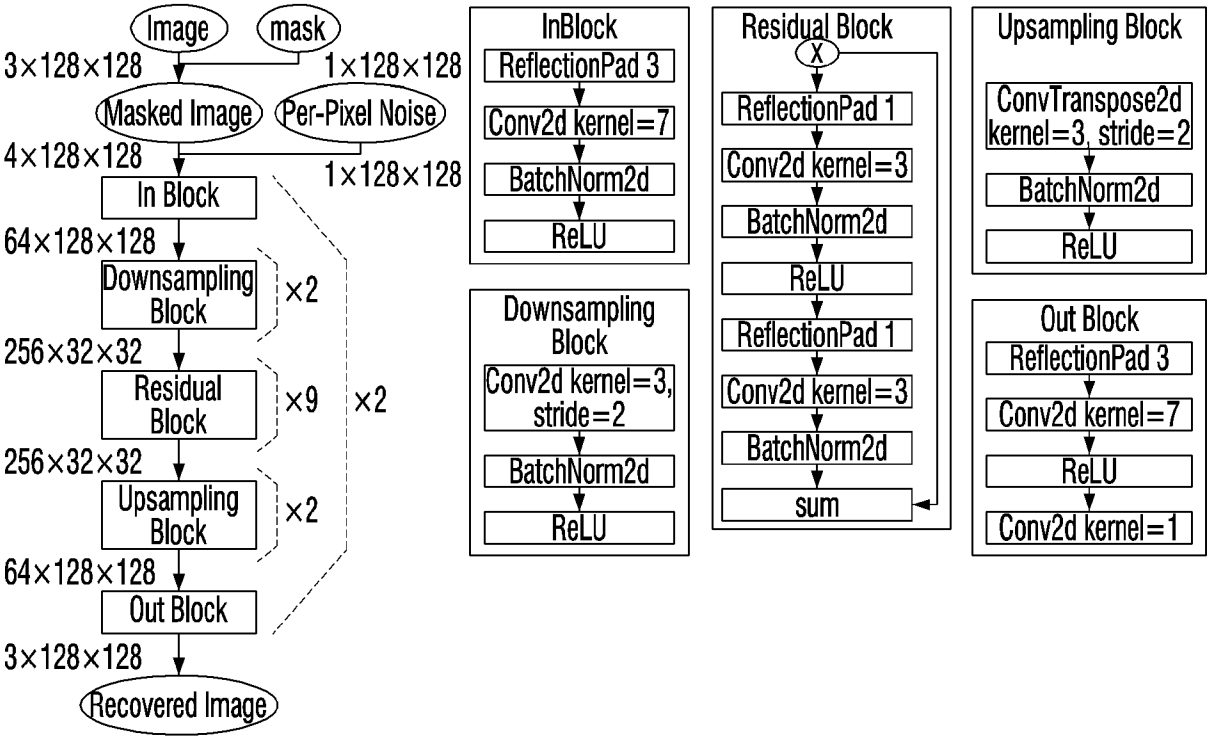
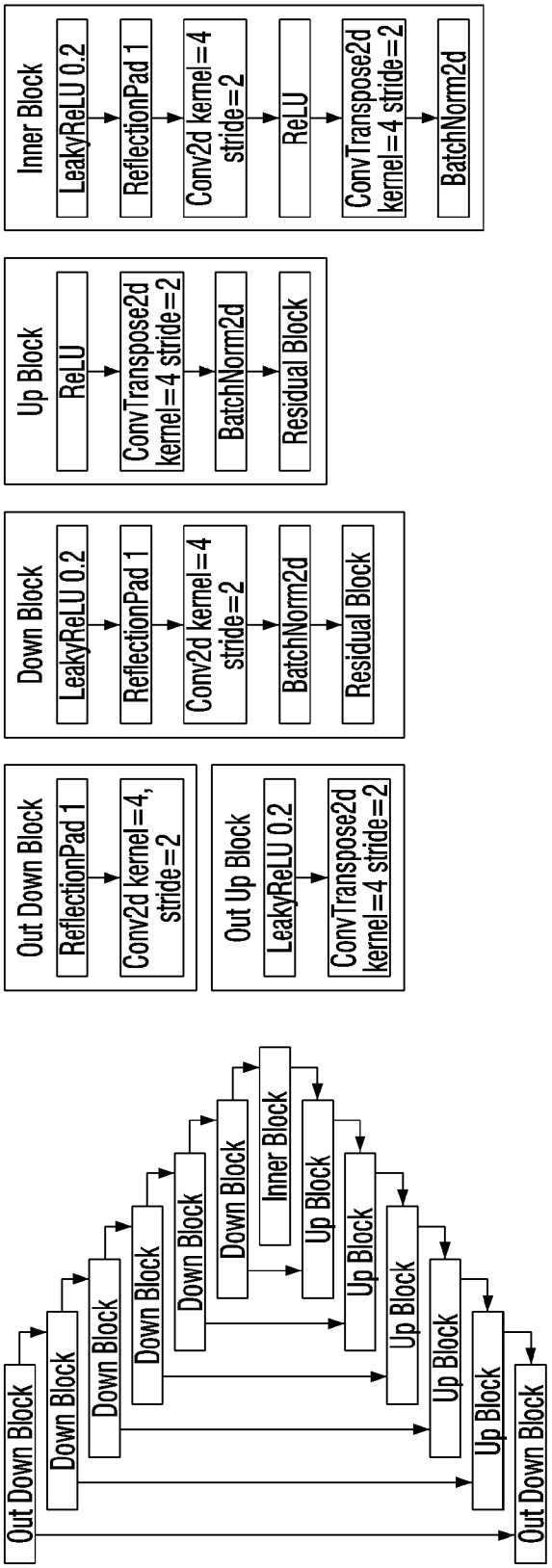


Image
mask
3×128×128
1×128×128
Masked Image
Per-Pixel Noise
4×128×128
1×128×128
In Block
64×128×128
Downsampling Block
×2
256×32×32
Residual Block
×9
×2
256×32×32
Upsampling Block
×2
64×128×128
Out Block
3×128×128
Recovered Image
InBlock
ReflectionPad 3
Conv2d kernel=7
BatchNorm2d
ReLU
Downsampling Block
Conv2d kernel=3, stride=2
BatchNorm2d
ReLU
Residual Block
X
ReflectionPad 1
Conv2d kernel=3
BatchNorm2d
ReLU
ReflectionPad 1
Conv2d kernel=3
BatchNorm2d
sum
Upsampling Block
ConvTranspose2d kernel=3, stride=2
BatchNorm2d
ReLU
Out Block
ReflectionPad 3
Conv2d kernel=7
ReLU
Conv2d kernel=1

[Fig. 5]


Inner Block
LeakyReLU 0.2
ReflectionPad 1
Conv2d kernel=4 stride=2
ReLU
ConvTranspose2d kernel=4 stride=2
BatchNorm2d
Up Block
ReLU
ConvTranspose2d kernel=4 stride=2
BatchNorm2d
Residual Block
Down Block
LeakyReLU 0.2
ReflectionPad 1
Conv2d kernel=4 stride=2
BatchNorm2d
Residual Block
Out Down Block
ReflectionPad 1
Conv2d kernel=4, stride=2
Out Up Block
LeakyReLU 0.2
ConvTranspose2d kernel=4 stride=2
Out Down Block
Down Block
Down Block
Down Block
Down Block
Down Block
Inner Block
Up Block
Up Block
Up Block
Up Block
Up Block
Out Down Block

INTERNATIONAL SEARCH REPORT

International application No.
PCT/KR2019/014916

A. CLASSIFICATION OF SUBJECT MATTER

G06T 7/10(2017.01)i, G06T 5/00(2006.01)i, G06T 5/20(2006.01)i, G06N 3/02(2006.01)i

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
G06T 7/10; G06K 9/00; G06K 9/66; G06T 5/00; G06T 5/50; H04N 13/00; G06T 5/20; G06N 3/02

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched
Korean utility models and applications for utility models
Japanese utility models and applications for utility models

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)
eKOMPASS(KIPO internal) & keywords: restore, image, object, background, neural network

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	XU ZHAO et al., `JOINT BACKGROUND RECONSTRUCTION AND FOREGROUND SEGMENTATION VIA A TWO-STAGE CONVOLUTIONAL NEURAL NETWORK', arXiv.org, 24 July 2017, pp. 1-6 [retrieved on 30 January 2020]. Retrieved from <https://arxiv.org/abs/1707.07584> See pages 1-6.	1-15
A	US 9396523 B2 (MICROSOFT CORPORATION) 19 July 2016 See claims 1-12; and figures 3-6.	1-15
A	US 9760978 B1 (ADOBE SYSTEMS INCORPORATED) 12 September 2017 See claims 1-8; and figures 2-4.	1-15
A	US 2018-0232887 A1 (ADOBE SYSTEMS INCORPORATED) 16 August 2018 See claims 1-10; and figure 1.	1-15
A	KR 10-2016-0062571 A (SAMSUNG ELECTRONICS CO., LTD.) 02 June 2016 See claims 1-18; and figures 1, 3-4.	1-15

☐ Further documents are listed in the continuation of Box C.

☒ See patent family annex.

* Special categories of cited documents:
"A" document defining the general state of the art which is not considered to be of particular relevance
"D" document cited by the applicant in the international application
"E" earlier application or patent but published on or after the international filing date
"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)
"O" document referring to an oral disclosure, use, exhibition or other means
"P" document published prior to the international filing date but later than the priority date claimed
"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention
"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone
"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents,such combination being obvious to a person skilled in the art
"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
19 February 2020 (19.02.2020)	**21 February 2020 (21.02.2020)**

Name and mailing address of the ISA/KR
International Application Division
Korean Intellectual Property Office
189 Cheongsa-ro, Seo-gu, Daejeon, 35208, Republic of Korea
Facsimile No. +82-42-481-8578

Authorized officer
KIM, Sung Hoon
Telephone No. +82-42-481-8710

INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No.

PCT/KR2019/014916

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
US 9396523 B2	19/07/2016	US 2015-0030237 A1	29/01/2015
US 9760978 B1	12/09/2017	None	
US 2018-0232887 A1	16/08/2018	US 2017-0287137 A1	05/10/2017
		US 9972092 B2	15/05/2018
KR 10-2016-0062571 A	02/06/2016	CN 105631813 A	01/06/2016
		EP 3026627 A2	01/06/2016
		EP 3026627 A3	02/11/2016
		JP 2016-100901 A	30/05/2016
		US 2016-0148349 A1	26/05/2016
		US 9811883 B2	07/11/2017