IamAlpha: Instant and Adaptive Mobile Network for Alpha Matting

Avinav Goel
avinav.goel@samsung.com
Manoj Kumar
manoj.kumar5@samsung.com
Pavan Sudheendra
pavan.s@samsung.com

Visual Intelligence Team
Samsung R&D Institute
Bengaluru, India

Abstract

Extraction of high quality alpha mattes from natural images has been a crucial problem with wide range of applications in the real world. Currently, most of the image matting techniques require a marked unknown region known as “Trimap”, as input for estimating alpha. But due to lack of trimap, majority of the techniques tend to generate the trimap by eroding and dilating ground-truth alpha maps. This in turn makes the prior-art inflexible towards minor inaccuracies introduced while making use of segmentation-based trimaps. In this paper, we introduce a novel, state of the art alpha matting model, “IamAlpha”, which uses trimap adaptation as an auxiliary task to adapt and fix the input trimap errors so that our alpha network focuses primarily on estimating transparency of high-level features (fine structures like hair, furs etc.) crucial to image matting. This in-turn helps us to enable high quality matting applications in real time at 60fps on GPU and 30fps on mobile hardware.

1 Introduction

Image matting has been a well-known problem which involves separating the foreground and the background by estimating the pixel level transparency of an object in an image. The most commonplace application of the same can be witnessed in movies where shooting is done with blue/green screens and then replaced with appropriate VFX backgrounds. But the task becomes difficult when extracting foregrounds in naturally captured scenes where the foreground color matches the background. Any natural image composition can be mathematically written as follows:

$$I_i = \alpha_i * F_i + (1-\alpha_i) * B_i,\ \alpha_i \in [0,1] \tag{1}$$

Where I_i refers to colour values observed in final composed image at pixel position i formed by mixing colors from foreground(F_i) and background(B_i) based on their alpha strength(α_i) at pixel i. The value of α_i varies in the range of 0 to 1, representing value of 0 to be definite background pixel and 1 to be a definite foreground pixel and intermediate values representing a blend between the two. A typical trimap is also formed in the same manner with 0 value representing a definite background, 1 value representing a definite foreground and a gray region with unknown opacity. The current known techniques for creating trimap are either manual through hand scribbling[19] or automatically by using binary segmentation networks. Both of them seem to have high probability of producing errors in input trimap, thereby leading to false-positives or false-negatives in the final matte output. For instance, some common mistakes include marking part of the unknown region in trimap as foreground(1) or potentially completely missing out on the far away unknown hair region by marking it as a definite background(0) in the input trimap. Even the current state of the art networks like FBA[5], GCA[9] etc., are highly dependent upon the correctness of the input trimap. The core focus of such networks has been to only estimate the alpha-affinity of an unknown pixel based on the definite foreground in the trimap. But if segmentation based trimaps are used, the inaccuracies in segmentation maps might propagate to the final matting map(Fig.2:second row). To reduce such errors, there is a need for matting networks that adapt the input trimap along with estimating foreground transparency. Therefore, to make our network perceptive to inaccuracies in the input trimap, we introduced an auxiliary trimap decoder which completely focuses on rectifying the aforementioned issues in the trimap and alongside reduce the unknown area of the trimap. Using such an auxiliary decoder while training has not only enabled us to have a light weight matting network (Fig.1) but has also reduced load on alpha decoder and enhanced its quality, thereby making it the best among published work on alphamatting.com [12]. To summarize, the main contributions of our work are:

- Introduction of trimap adaptation as an auxiliary task for matting has helped in reducing the computational complexity of the network.
- Trimap adaptation as an auxiliary task has helped in making matting task invariant to unwanted artifacts in segmentation based trimaps. This has enabled automated end to end real time image matting applications. The effectiveness of the trimap adaptation to improve the accuracy is established through an ablation study in the paper. With this decoupling we are able to achieve performance of 60fps and 30fps on PC and mobile respectively.
- Re-binding decoupled tasks of trimap adaptation and alpha estimation with trainable weights has yielded good improvements in accuracy as established through an ablation study in the paper.

This paper has been organized as follows: In section 2, we discuss about the published work on matting systems. In section 3, we propose multiple network optimizations and losses which are then accompanied by experiments in Section 4 to support the same. Finally, in Section 5 we compare our method against other state of the art networks on popular benchmarks both qualitatively and computationally to arrive at a proper conclusion in Section 6.

2 Related Work

Matting is generally dealt in two ways, first being color sampling approaches[4] which collect a set of background and foreground colors to determine the alphas in the unknown region based on each pixel affinities. The other is the propagation based approach[7] which start propagating alpha gradient outwards from the foreground while accounting for the color affinity of neighboring pixels.

2.1 DNN Based Approaches

Shen *et al.* [15], was one of the first ones to introduce a fully automated network to generate alpha mattes but their work was only limited to portrait images. Techniques have been explored in the field of predicting both alpha and foreground simultaneously[5], [7], but all of these methods resulted in being highly dependent on trimaps and having high computational complexities(Table1).

2.2 Trimap Generation and Adaptation

Most image matting techniques make use of a guide, also known as a trimap which is a rough indicator of foreground, background and an unknown region. While Wang et al[16] present a technique that makes use of scribbles to mark trimap unknown area using human interaction, other automatic techniques[5], [9] make use of erosion and dilation on ground truth matting maps to generate input trimaps. Researchers in [1], [11] have induced separate branches in their network for targeting particular trimap areas, but had the effect of making the network more complex. Hence we have made trimap adaptation task an auxiliary task which is only part of the training process and not being used at the time of inference which helps us to achieve a real time matting network. None of the aforementioned techniques could be automated as they tend to induce errors while dealing with trimap created from automated techniques such as segmentation for the same image (Fig.2:second row).

2.3 Real Time Mobile Networks

As per our current knowledge, there exists no prior art in the field of "generic object" alpha matting that runs instantly on mobile devices. Though, Lin et al.[10] recently proposed a real time GPU network that runs at similar performance but is limited to only "human portraits" and also requires more data such as a 3 channel background RGB image input as opposed to our 1 channel trimap input. Levinshtein *et al.* [8] have also attempted making use of a Mobile-net architecture based model for human hair matting, but due to coarse estimations, it was only compared against existent human segmentation networks and not matting.

3 METHODOLOGY

We treat the task of Trimap-Based Alpha matting to be equally dependent on both alpha estimation as well as the trimap adaptation. But for the sake of achieving a real time network, both the tasks can be easily decoupled and still be linked together using a single weight adaptive trainable loss function while retaining superior quality.

3.1 Architecture

In this paper, we propose a light weight multi-task loss network with one common encoder and two decoders. We make use of modified ResNet50 as our encoder with 5x down-sampling-layers. The alpha learning task can essentially be divided into two sub tasks, namely, Regression (Alpha Estimation) and Classification (Trimap Adaptation).

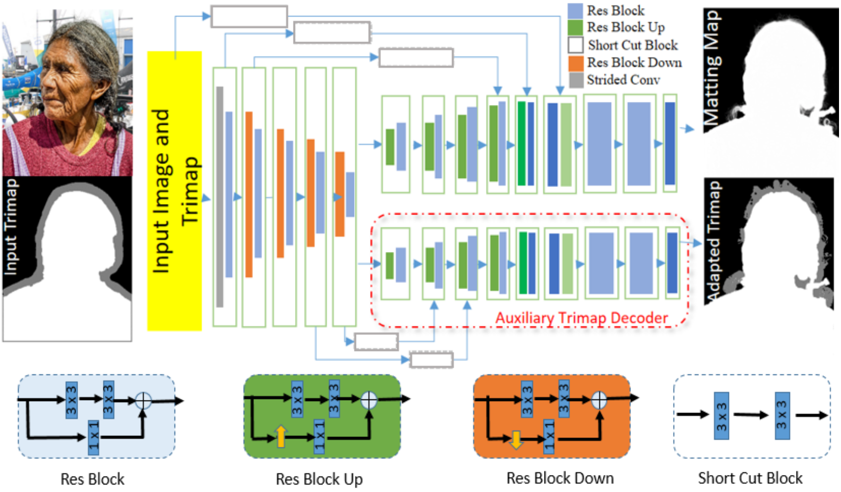


Figure 1: Model Architecture with Auxiliary Trimap Adaptation Network.

3.1.1 Alpha Estimation

The primary task of alpha estimation is to correctly estimate transparency in the foreground-background boundary. We make use of U-Net[13] like architecture with half-width residual blocks [6] and the encoder having 5x-down-sampling layers with skip connections after each block. We also make use of a direct input skip connection as the boundary sharpness is perfectly preserved in the input RGB map. Though our encoder contains a 5x down-sampler our alpha decoder only makes use of low level feature maps. This is because finer image features are lost on subsequent down sampling and high quality feature extraction is crucial to the underlying task of alpha estimation (Fig.1). Moreover we replaced all de-convolutions with fast Bilinear up-samplers as they are most efficient in preserving pixel gradients. The Residual Blocks[6], on the other hand have been designed specifically to use only small 1x1 or 3x3 kernels to achieve a lightweight high performance network.

3.1.2 Trimap Adaptation

This auxiliary network has been added for trimap correction and adaptation and it bears the greatest advantage of zero computational overhead because this decoder is not used during inference. As this task is mainly concerned with learning contextual information for trimap corrections, it is only fed with the higher level feature maps (Fig. 1). For training this particular task, the ground-truth trimaps are constructed from ground-truth alpha maps using the following equation:

$$Trimap_{GT} = \begin{cases} 2 & \alpha{=}1 \\ 1 & 0{<}\alpha{<}1 \\ 0 & \alpha{=}0 \end{cases} \quad (2)$$

where alpha(α) is the ground-truth alpha with values scaled between 0 to 1, with 0 as complete background and 1 as complete foreground.

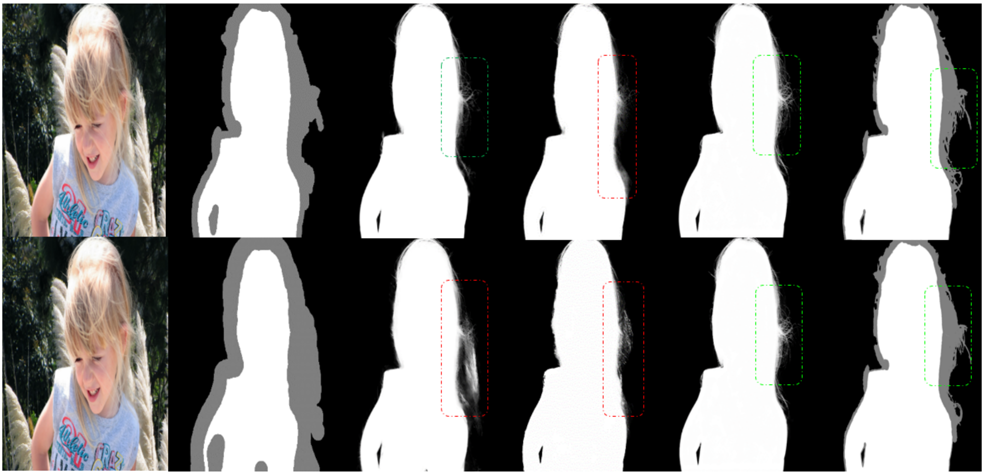


Figure 2: First row: Outputs generated with groundtruth-alpha based trimap, second row: outputs with segmentation based trimaps. Comparing top two rows shows that even the State of the art models, FBA [5] and GCA [9], only perform good on GT alpha based trimaps. *Green outlined boxes signifies better results than red outlined boxes.

3.2 Multi –Task Loss Function

The loss function here has been designed to train both the tasks of Alpha Estimation and Trimap Adaptation at the same time by dynamically adjusting individual task weights throughout the training. The multi-task loss function is as follows:

$$L(\{\widetilde{T},\widetilde{\alpha}\},\{T_{gt},\alpha_{gt}\}) = \frac{1}{2\sigma_1^2}L_T(\widetilde{T},T_{gt}) + \frac{1}{2\sigma_2^2}L_\alpha(\{\widetilde{T},\widetilde{\alpha}\},\alpha_{gt}) + \log 2\sigma_1\sigma_2 \tag{3}$$

where $\tilde{T}$and $\widetilde{\alpha}$ stand for the output of trimap adaptation and alpha estimation, σ_1 and σ_2 stand for dynamically adjusted task weights[3] (during training using Back-Propagation algorithm), L_T and L_α stand for trimap cross entropy and alpha estimation loss, respectively. Alpha estimation loss is a combination of L1, L2, Composition loss and SSIM loss. Alpha estimation loss is only applied on unknown area of estimated trimap ($\widetilde{T}$).

3.3 Real Time Matting

Achieving real-time performance for the task of generic object matting has been a challenging problem with no solution till date. With the help of our lightweight inference network, IamAlpha, for the first time, we are able to realize real-time high quality alpha maps with lowest computational costs (Table 1) till date. Our network provides state of the art quality at 60fps on a NVIDIA GTX 1080Ti GPU and 30fps on NPU of Qualcomm Snapdragon 888 mobile chip set.

4 Experiments

4.1 Dataset and Evaluation Metrics

4.1.1 Training Dataset

The data used for training is generated by composing foregrounds with random backgrounds based on their alpha. Due to lack of publicly available dataset for training, we collected a set of 440 unique foregrounds from copyright-free websites and annotated them in-house. This dataset collection maintains an equal distribution among all human and non-human objects distributed again over various transparencies, namely, Highly, Strongly, Medium and Little Transparent [12]. These unique foreground images were then composed over 2,000 unique and real backgrounds resulting in a dataset of 880k images for training.

4.1.2 Evaluation Metrics

We compare our model both qualitatively and quantitatively against prior art based on these six parameters: Mean Squared Error(MSE), Gradient Error (GRAD), Sum of Absolute Differences(SAD), Connectivity Error(Connect), Giga Multiply Accumulate Operations(GMAC) and Number of parameters as in Table 1. These comparisons have been made on leading benchmarks, namely, alphamatting.com[12] and Adobe Composition 1k Test-set [17] which are some of the most commonly used benchmark metrics used for comparing state of the art alpha matting models.

4.1.3 Training Details

The model is trained using Adam Optimizer with base learning rate of 1e-4 and a momentum of 0.9. We make use of polynomial learning rate scheduler which drops over to 1e-6 over 300k iteration of batch 16. The training runs over a span of 2 weeks on a Tesla P40 GPU. Input preprocessing and augmentations play a critical role here. The training images are randomly down-sampled to a 0.8 scale, to avoid loss in quality. Augmentations such as rotation, color jitter, translation are performed. Lanczos resizer is used to down-sample so as to retain maximum edges and gradients in the alpha.

4.2 Ablation Study

In order to prove the effectiveness of the chosen model architecture, we further discuss the extensive studies that were carried out.

4.2.1 Auxiliary Trimap Decoder

Trimaps play a very crucial role in determining the quality of the output. Xu et al [17], like others, have also pointed out how the alpha quality degrades as the trimap unknown increases. Also, the trimap adaptation task becomes necessary when dealing with Trimaps created from segmentation networks. So we performed an ablation study which analyzes the impact of using trimap correction as an auxiliary task, where a common encoder learns to target only a small set of unknown pixels instead of the entire image. A similar attempt was also made in Index-Net[11], where they developed deep custom architecture layers to

Method	Compute Cost		Adobe Composition 1k Dataset				AlphaMatting.com Benchmark			
	GMAC	MParam	MSE	Grad	SAD	Connect	MSE	Grad	SAD	Connect
KNN[2]	-	-	103.0	124.1	175.4	176.4	39.3	46.3	43.7	35.3
DIM[17]	44.56	30.5	14.0	31.0	50.4	50.8	19.6	24.1	16.4	21.0
Ada-Matting[1]	-	-	10.2	16.9	41.7	-	13.5	13.2	13.0	23.5
GCA[9]	25.21	25.27	9.1	16.9	35.3	32.5	15.6	13.8	14.8	22.8
Index-Net[11]	71.41	35.95	13.0	25.9	45.8	43.7	23.5	18.9	20.0	26.0
BM[14]*	985.3	17.91	21.0	19.9	16.07	18.1	12.5	12.0	13.1	15.0
RTBM[10]*	12.67	40.25	12.0	8.42	12.8	11.1	-	-	-	-
FBA[5]	35.436	34.69	**5.2**	**10.6**	**25.8**	**20.8**	-	-	-	-
Base	10.005	14.72	16.5	46.2	59.27	49.7				
Base+AN	10.005	14.72	9.0	16.8	37.2	36.1	**4.7**	**5.2**	**4.0**	**13.2**
Base+AN+TW	**10.005**	14.72	8.3	15.1	34.6	30.5				

Table 1: Quantitative Comparison on Composition 1k Set and alphaMatting.com[12]. MSE, Grad and SAD are scaled to x103. "-" indicates data/model not available. "*" refers to models computations for only "Human" Images and input Background RGB. AN: Auxiliary Network, TW: Trainable Weights, GMAC: Giga Multiply Accumulate Operations (800x800), MParam: Million Model Parameters.
* Metrics for FBA and RTBM is not avalable on AlphaMatting.com

concentrate only on certain indexes of the trimap. Our study of decoupling Trimap Adaptation task proved effective in reducing our losses by half without any computational overhead (Base vs Base + AN (Auxiliary Network) in Table1).

4.2.2 Trainable Weights

Our loss function has been specially designed to have trainable weights(TW) for alpha and trimap tasks. Initially both weights σ_1 and σ_2 are initialized equally with a value of 4. Addition of trainable weights to optimizer after 10% of the training made the loss weights dynamically adjust to the individual tasks at hand and produce the best metrics overall (Base+AN vs Base+AN+TW in Table1).

5 Results and Discussion

In this section, we report and benchmark our model, IamAlpha, against top ranking networks both qualitatively and quantitatively.

5.1 Quantitative Evaluations

5.1.1 AlphaMatting.com Online Benchmark[12]

`alphamatting.com`[12] is a very popular benchmark for "generic object" image matting which compares low resolution images of different alpha transparencies(Strong, Highly, Medium and Little Transparent) against various trimap configurations(small, large and user). We have achieved the highest rank in all metrics, namely SAD, MSE, Grad and Connectivity among all published work currently present at the time of submission(Table 1). This proves the effectiveness of our trimap correction technique when compared against "user" trimaps present on this benchmark. The corresponding alpha matte outputs for the same are also available on their website. Results for FBA[5] have not been quoted as the website seems to have removed it and as this benchmark provides scores based on current ranks, the old numbers cited in the paper might not be indicative of it's current position on the benchmark.

5.1.2 Adobe Composition 1k Dataset[17]

This test dataset contains over 50 unique generic object FGs each composited over 20 unique backgrounds from Pascal VOC Dataset. Though this test-set makes use of trimaps derived from groundtruth-alpha maps, our lightweight mobile model, IamAlpha, still outranks most of the state of the art GPU networks. Our parameters and GMACs are almost one-third of the leading FBA[5] matting model. FBA [5] model makes use of a very deep model to predict all of Foreground Background and Alpha simultaneously. This makes the model too heavy for any future real-time applications.

5.2 Qualitative Evaluations

The qualitative results of our proposed model have been evaluated against the state of the art GPU models on these two parameters:

5.2.1 Complex Backgrounds

As shown in Fig.4, even one of the best GPU Matting models, FBA[5] shows false positives with highly textured backgrounds despite such structures being marked as a part of definite background in the input trimap. As Composition 1k dataset[17] makes use of low resolution backgrounds(Fig.3), the test-set becomes insufficient to support real world scenarios. In such cases, our model utilizes the ability of the auxiliary trimap to classify unknown pixels in the image based on their global context and simultaneously reduce the unknown areas in input trimaps(Fig.3 Input Trimap(b) vs Adapted Trimap(f)).

5.2.2 Crude Trimaps

A good trimap either generated from human interaction or ground truth alpha mattes is not viable in real life situations. The best possible way to automatically generate trimaps is to make use of real-time semantic segmentation networks such as BiSeNetv2[18]. Our model, in Fig.1 and Fig.2(Row 1 vs Row 2), shows that it can adapt in such situations also and extend trimap to regions with features similar to the foreground and thus generate proper alpha mattes. On the other hand, we can see that even the best networks like FBA[5] can only work when provided with groundtruth-alpha based trimaps thereby making our claim for trimap correction even more salient for alpha matting tasks.

5.3 Computational Complexity Evaluations

To build a faster network, Sengupta *et al.* [14] have introduced a new limited utility network only for Human Portrait Matting, RTBM[10], where they pass in down-scaled composite and background RGB image as input to their network and internally use bilinear up-sampler to resize back to provide Full HD matting maps at 60fps on GPU. Our generic object matting mobile network, IamAlpha, on the other hand processes full resolution inputs and uses task decoupling to produce even superior metrics with fewer GMACs and model Parameters (Table 1) with performance more than 60fps on GPU and 30fps on mobile hardware. This helps

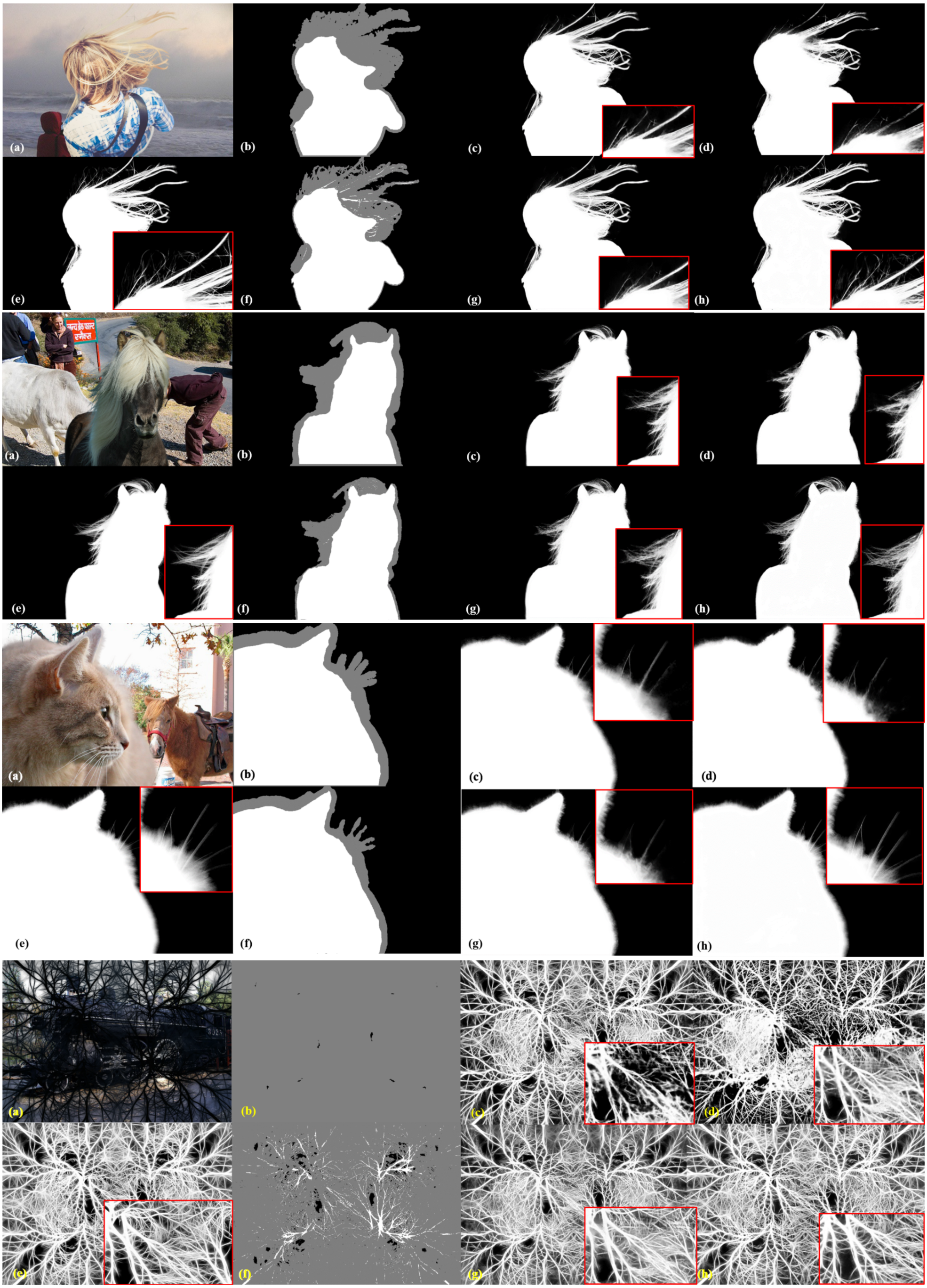
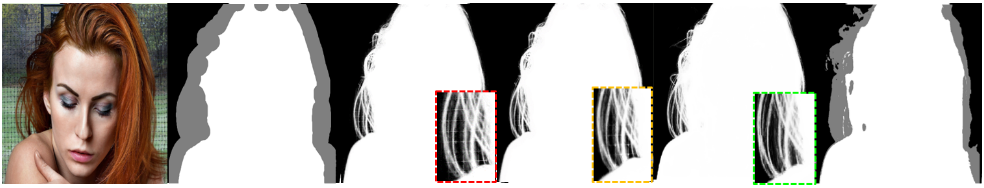


Figure 3: (a)Composite RGB Image from Adobe Composition 1k TestSet[17], (b) Input Trimap[17], (c)FBA[5], (d)DIM[17], (e) Groundtruth Alpha Matte[17], (f) Our Adapted Trimap, (g) GCA[2],(h) IamAlpha(Ours)



(a) (b) (c) (d) (e) (f)

Figure 4: (a):Input RGB (b):Input Trimap (c):FBA[5] (d):GCA[9] (e):IamAlpha (f):Output Trimap.
It shows state of the art models suffering from False Positives while dealing with real complex backgrounds

our network to grasp fine matting structures from full scale inputs with just a single channel input trimap easily producible from any semantic segmentation network.

6 Conclusion

A novel technique of decoupling alpha matting and trimap adaptation tasks has been introduced. For the first time, we have achieved high quality alpha matting network, IamAlpha, with the ability to deploy on mobile devices at 30 fps and GPU at 60fps for Full HD image. Our proposed network not only targets unknown area reduction but also tries to handles false positives in segmentation based trimaps. Having a detachable trimap network has helped us to reduce load on alpha decoder and thereby provide a lightweight inference network for Alpha Matting which currently yields state of the art results, both qualitatively and quantitatively on global benchmark alphamatting.com[12] amongst all published works.

References

[1] Shaofan Cai, Xiaoshuai Zhang, Haoqiang Fan, Haibin Huang, Jiangyu Liu, Jiaming Liu, Jiaying Liu, Jue Wang, and Jian Sun. Disentangled image matting. In *Proc. IEEE Int. Conf. Comput. Vis.*, volume 2019-Octob, pages 8818–8827, 2019. ISBN 9781728148038. doi: 10.1109/ICCV.2019.00891.

[2] Qifeng Chen, Dingzeyu Li, and Chi Keung Tang. KNN matting. In *IEEE Trans. Pattern Anal. Mach. Intell.*, volume 35, pages 2175–2188, 2013. ISBN 9781467312264. doi: 10.1109/TPAMI.2013.18.

[3] Roberto Cipolla, Yarin Gal, and Alex Kendall. Multi-task Learning Using Uncertainty to Weigh Losses for Scene Geometry and Semantics. In *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, pages 7482–7491, 2018. ISBN 9781538664209. doi: 10.1109/CVPR.2018.00781.

[4] Xiaoxue Feng, Xiaohui Liang, and Zili Zhang. A cluster sampling method for image matting via sparse coding. In *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, volume 9906 LNCS, pages 204–219, 2016. ISBN 9783319464749. doi: 10.1007/978-3-319-46475-6_13.

[5] Marco Forte and François Pitié. \$F\$, \$B\$, Alpha Matting, 2020. URL http://arxiv.org/abs/2003.07711.

[6] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, volume 2016-Decem, pages 770–778, 2016. ISBN 9781467388504. doi: 10.1109/CVPR.2016.90.

[7] Qiqi Hou and Feng Liu. Context-aware image matting for simultaneous foreground and alpha estimation. In *Proc. IEEE Int. Conf. Comput. Vis.*, volume 2019-Octob, pages 4129–4138, 2019. ISBN 9781728148038. doi: 10.1109/ICCV.2019.00423.

[8] Alex Levinshtein, Cheng Chang, Edmund Phung, Irina Kezele, Wenzhangzhi Guo, and Parham Aarabi. Real-time deep hair matting on mobile devices, 2018.

[9] Yaoyi Li and Hongtao Lu. Natural Image Matting via Guided Contextual Attention, 2020. ISSN 2159-5399.

[10] Shanchuan Lin, Andrey Ryabtsev, Soumyadip Sengupta, Brian Curless, Steve Seitz, and Ira Kemelmacher-Shlizerman. Real-Time High-Resolution Background Matting. 2020. URL http://arxiv.org/abs/2012.07810.

[11] Hao Lu, Yutong Dai, Chunhua Shen, and Songcen Xu. Indices matter: Learning to index for deep image matting. In *Proc. IEEE Int. Conf. Comput. Vis.*, volume 2019-Octob, pages 3265–3274, 2019. ISBN 9781728148038. doi: 10.1109/ICCV.2019.00336.

[12] Christoph Rhemann, Carsten Rother, Jue Wang, Margrit Gelautz, Pushmeet Kohli, and Pamela Rott. A perceptually motivated online benchmark for image matting, 2009.

[13] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, volume 9351, pages 234–241, 2015. ISBN 9783319245737. doi: 10.1007/978-3-319-24574-4_28.

[14] Soumyadip Sengupta, Vivek Jayaram, Brian Curless, Steve Seitz, and Ira Kemelmacher-Shlizerman. Background matting: The world is your green screen. In *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, pages 2288–2297, 2020. doi: 10.1109/CVPR42600.2020.00236.

[15] Xiaoyong Shen, Xin Tao, Hongyun Gao, Chao Zhou, and Jiaya Jia. Deep automatic portrait matting, 2016. ISSN 16113349.

[16] Jue Wang and Michael F. Cohen. An iterative optimization approach for unified image segmentation and matting, 2005. ISSN 15505499.

[17] Ning Xu, Brian Price, Scott Cohen, and Thomas Huang. Deep image matting. In *Proc. - 30th IEEE Conf. Comput. Vis. Pattern Recognition, CVPR 2017*, volume 2017-Janua, pages 311–320, 2017. ISBN 9781538604571. doi: 10.1109/CVPR.2017.41.

[18] Changqian Yu, Changxin Gao, Jingbo Wang, Gang Yu, Chunhua Shen, and Nong Sang. Bisenet v2: Bilateral network with guided aggregation for real-time semantic segmentation, 2020.

[19] Yuanjie Zheng and Chandra Kambhamettu. Learning based digital matting, 2009.