

756 A ADDITIONAL ABLATIONS

757
758 A.1 IP ADAPTER VS CAT

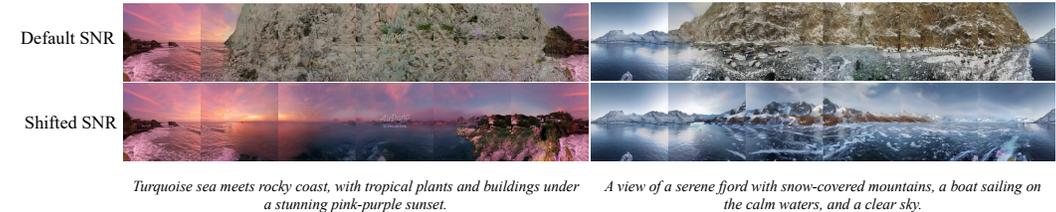
759
760 A popular method for image-conditioning in image diffusion models is IP Adapter [Ye et al. \(2023\)](#).
761 A CNN feature extractor takes the conditional input views and extracts features that will then be
762 added to the intermediate features in the diffusion model forward pass. Here we compare it to using
763 the conditioning method from CAT3D that directly uses conditional inputs as frames to the diffusion
764 model without noising. We generally find that they are similar but IP adapter can exhibit more abrupt
765 transitions between the input condition and neighbouring regions in the generated panorama. We
766 show a few examples in [Fig. A1](#).



773 Figure A1: Qualitative figure comparing IP vs CAT type architecture for input conditioning. When
774 using IP adapter, the consistency between input conditioning views and neighbouring views (high-
775 lighted in red box) is worse compare to CAT.
776

777
778 A.2 ABLATING THE EFFECTS OF SHIFTING THE NOISE SCHEDULE

779
780 During inference we use up to $8 \times 16 = 128$ frames which is much larger than the 16 frames used
781 by the base video model. As mentioned in [§ 3.1](#) the increased data dimensionality also requires a
782 corresponding increase in terminal noise to minimize the terminal step gap with the noise prior. In
783 particular we interpolate between the standard noise schedule and a noise schedule that has been
784 shifted by 10. We compare these qualitatively in [Fig. A2](#). Note that without changing the noise
785 schedule, the model is largely incapable of generating plain regions such as clear sky or white snow
786 fields and instead fills in the frame with visual clutter.



794 Figure A2: Qualitative comparison of shifting the noise schedule in the video-conditioned setting.
795 Each of the six horizontal views is visualized independently before stitching into a panorama. With-
796 out shifting toward higher noise levels, the model struggles to generate clear skies or water, intro-
797 ducing objects that disrupt scene cohesion (e.g., sudden mountains and rocks).
798

799
800 A.3 ABLATING THE EFFECT OF NOISE AUGMENTATION FOR AUTOREGRESSIVE GENERATION

801
802 In this section, we qualitatively analyze the impact of noise augmentation during training on the
803 model’s autoregressive generation performance. To demonstrate this, we compare two models: one
804 trained with noise augmentation and the other without. To maximize the effect of error accumula-
805 tion, we use both models to 6 frames at a time, for a total of 10 iterations to get a video consisting
806 of $10 \times 5 + 1 = 51$ frames. [Figure A3](#) shows a side-by-side comparison of the different scenarios.

807 As autoregressive iterations increase, the model without noise-augmentation produces increasingly
808 saturated frames. While the model trained with noise augmentation also shows some degradation, it
809 maintains significantly better output quality over time, demonstrating its usefulness in reducing the
severity of error accumulation.



822
823
824
825
826
827

Figure A3: Qualitative comparison of autoregressive generation with and without noise augmentation. Both models exhibit a decline in output quality over time, but the model trained without noise augmentation shows a more rapid and severe degradation, with frames becoming increasingly saturated. In contrast, the model with noise augmentation deteriorates more gradually.

828 A.4 ABLATING THE EFFECTS OF FREEZING BASE MODEL LAYERS



841
842
843
844

Figure A4: Qualitative figure comparing text conditional panorama video generation using base model freezing vs no freezing. Freezing model weights is better able to retain some of the prior knowledge on out of distribution prompts.

845
846
847
848
849

When finetuning our model for multi-view generation we choose to freeze the base model layers. We ablate this choice qualitatively here on the text conditional panorama video generation task. We evaluate out of distribution prompts that make the overfitting behaviour very obvious when not freezing any base layers as can be seen in Fig.A4.

850 B ADDITIONAL VIDEO CONDITIONAL RESULTS

851
852
853
854

We show more video conditional generation results in Fig. B1 where we also apply autoregressive generation to extend the video length.

855 C ADDITIONAL TRAINING DETAILS

856
857
858
859
860
861
862
863

During the first stage of training we adapt the base video model towards the shifted and interpolated noise schedule as well as the v-prediction parameterization. This stage is trained for 10,000 iterations on the original dataset and a batch size of 128. Following that we insert the multi-view attention layers and train our model using the multiview video data. The batch size for this phase is 32 and we train these models for 15,000 iterations. Both stages use a constant learning rate of 0.0001. Most of our experiments are conducted on 32 A100 GPUs (or lower using gradient accumulation).

864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917

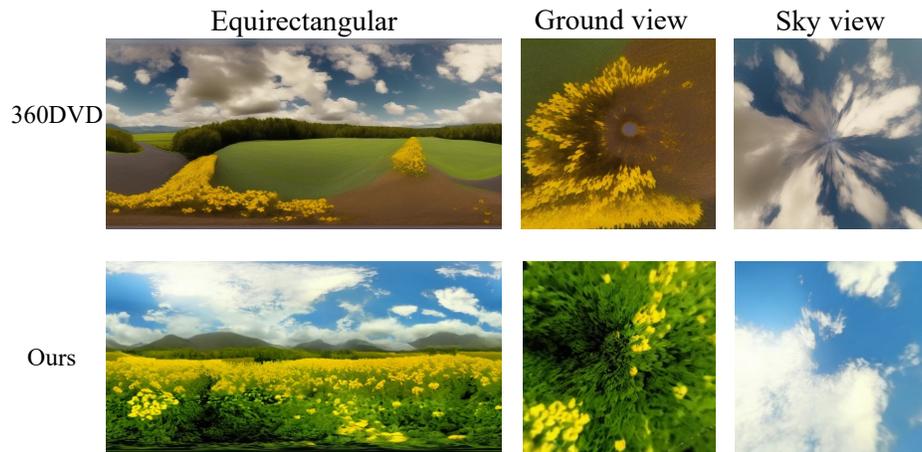


Figure A5: Qualitative figure comparing text conditional video generation, 360DVD VS ours and highlighting the distortion in 360DVD near the poles. Note that both generations were first transformed to the same equirectangular format before consistent sky and ground views were extracted. 360DVD struggles in these views as the distortion is highest here and deviates the most from perspective view images whereas we natively generate perspective views.

918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971



Figure B1: More results of video conditional autoregressive generation on out of distribution videos.