

2. LANDASAN TEORI

2.1 Tinjauan Pustaka

Bagian ini akan menjelaskan teori-teori yang sudah dipelajari dan akan dipakai dalam proses penelitian dan pembuatan skripsi ini.

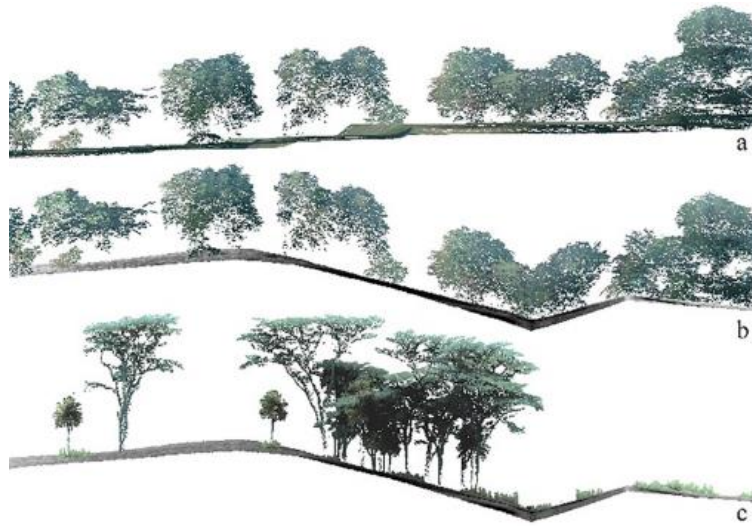
2.1.1 3D Object Reconstruction

3D *Object Reconstruction* adalah suatu proses pembuatan representasi tiga dimensi suatu object dari sekumpulan data, seperti gambar, *point cloud*, *depth map*, atau *voxel* (Singh & Tanna, 2020). Penelitian (Yu et al., 2023) menyatakan bahwa hasil dari 3D *Object Reconstruction* telah banyak digunakan untuk robot, *virtual reality*, *computer aided design* (CAD), *unmanned aerial vehicle* (UAV), dan konstruksi kota.

Sebagian besar metode 3D *Object Reconstruction* dari gambar mengandalkan *input multi-view*, yaitu beberapa gambar objek yang ditangkap dari beberapa perspektif yang berbeda (Ho et al., 2021). Bahkan dengan *input multi-view*, 3D *Object Reconstruction* mengalami tantangan ketika pencahayaan dan oklusi tidak konsisten di seluruh pandangan (Lei et al., 2020). Berbeda dengan manusia yang dapat mengestimasi struktur tiga dimensi suatu objek dengan mudah dari satu gambar, suatu metode 3D *Object Reconstruction* tidak memiliki pengetahuan yang kuat sebelumnya mengenai bentuk dan tekstur suatu objek tiga dimensi (Ho et al., 2021).

2.1.2 3D Point Cloud

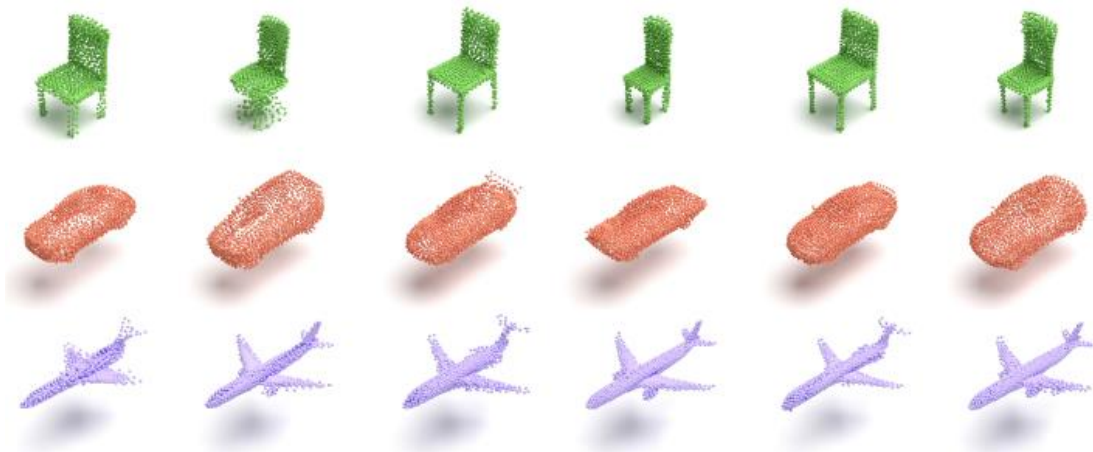
3D *Point Cloud* adalah representasi data tiga dimensi dari suatu objek atau lingkungan fisik dalam bentuk kumpulan titik (Hurtado & Valada, 2022). Setiap titik dalam *point cloud* akan menyimpan informasi tentang koordinat spasialnya, yang mencakup posisi x, y, dan z di dalam ruang tiga dimensi (Y. Zhang et al., 2023). *Point cloud* dapat diperoleh melalui berbagai metode pemindaian, seperti pemindaian laser (LiDAR), *photogrammetry*, atau pemindaian struktur berbasis *Kinect* (H. Zhang et al., 2023, p. 3). 3D *Point Cloud* digunakan di berbagai aplikasi, termasuk pemodelan objek dalam suatu *video game*, pengawasan konstruksi, ataupun untuk pemetaan lingkungan (Mirzaei et al., 2022).



Gambar 2.1 Ilustrasi Topografi Vegetasi Dalam Representasi 3D Point Cloud

Sumber: Urech, P. R. W., Dissegna, M. A., Girot, C., & Grêt-Regamey, A. (2020). Point cloud modeling as a bridge between landscape design and planning. *Landscape and Urban Planning*, 203, 103903. <https://doi.org/10.1016/j.landurbplan.2020.103903>

Representasi objek dengan 3D *Point Cloud* mempertahankan informasi geometris asli dalam ruang 3D dan karena itulah representasi 3D *Point Cloud* sering digunakan untuk aplikasi terkait pemahaman pemandangan seperti *autonomous driving* dan robotika (Guo et al., 2021). 3D *Point Cloud* mampu merepresentasikan detail yang sangat tinggi secara padat dibandingkan dengan *voxel*, meskipun 3D *Point Cloud* merupakan bentuk representasi yang lebih kompleks (G. Yang et al., 2019).



Gambar 2.2 Contoh Hasil 3D *Point Cloud* oleh *Deep Learning Model*.

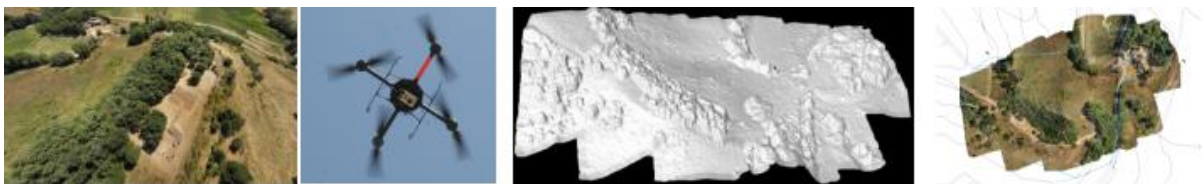
Sumber: Wen, C., Yu, B., Fu, R., & Tao, D. (2023). *Patch-Wise Point Cloud Generation: A Divide-and-Conquer Approach*. <https://doi.org/10.48550/ARXIV.2307.12049>

Dengan perkembangan *machine learning*, pemrosesan manual 3D *Point Cloud* yang mengonsumsi waktu dan kekuatan komputasi mulai dapat di automasi (Mirzaei et al., 2022). Dalam 3D *Object Reconstruction*, kelengkapan dan kualitas 3D *Point Cloud* suatu *scene* sangatlah

berdampak terhadap kemampuan algoritma lainnya dalam merencanakan rute dan *autonomous driving* (Fei et al., 2022).

2.1.3 Photogrammetry

Photogrammetry adalah sebuah metode pengambilan gambar suatu objek ataupun struktur dari perspektif yang berbeda untuk mendapatkan pengukuran dan rekonstruksi objek atau lingkungan secara akurat (Aharchi & Ait Kbir, 2020). *Photogrammetry* mengandalkan konsep triangulasi, sehingga pengambilan beberapa gambar suatu objek dari perspektif yang berbeda perlu *overlap* (Jiang et al., 2018). Gambar-gambar ini dapat diambil menggunakan kamera biasa, drona, ataupun satelit. Dalam prosesnya, *photogrammetry* bertujuan untuk mengidentifikasi fitur-fitur dan menghitung posisi 3D-nya sehingga rekonstruksi objek dapat dilakukan (Marín-Buzón et al., 2021).



Gambar 2.3 Pemandangan Hasil Survei Sistem UAV

Sumber: Radford, A., Metz, L., & Chintala, S. (2015). *Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks*. <https://doi.org/10.48550/ARXIV.1511.06434>

Photogrammetry diterapkan di berbagai macam industri, seperti di bidang kartografi untuk melakukan pemetaan topografi, di sektor konstruksi untuk membantu pemodelan lokasi, ataupun dalam bidang *entertainment* seperti dalam pemodelan 3D dalam *video game* dan film (Saif & Alshibani, 2022). Perkembangan penerapan tersebut semakin pesat karena mudahnya penerapan teknik *photogrammetry* dalam pengambilan foto dengan kamera *digital*. Keakuratan *photogrammetry* bergantung besar pada dua faktor yaitu kualitas gambar dan jumlah titik referensi (Dębski et al., 2021).

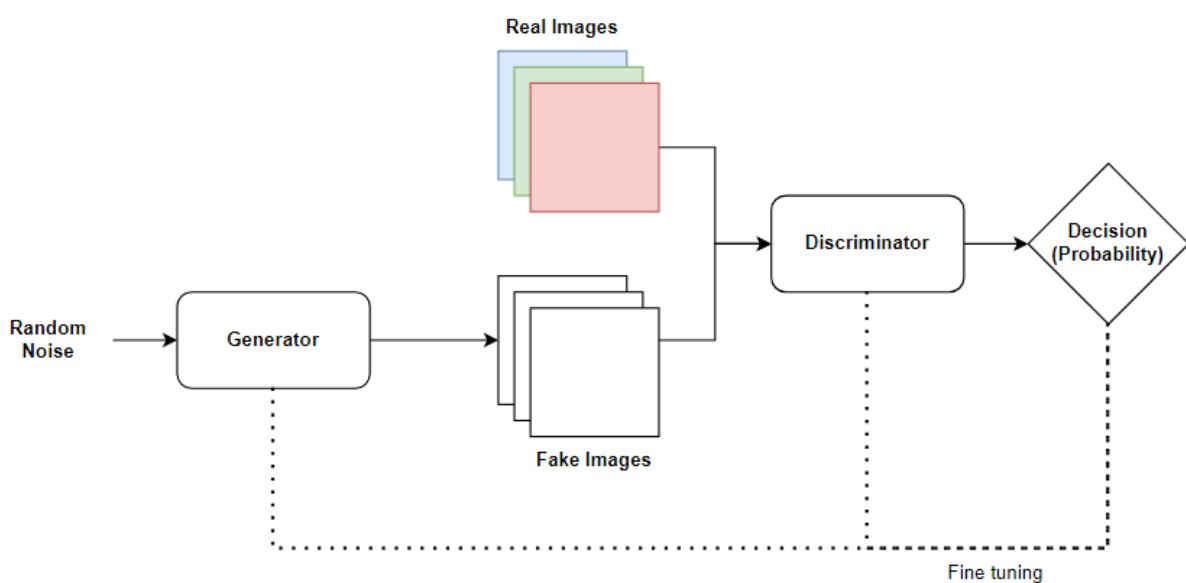
2.1.4 Generative Adversarial Network (GAN)

Generative Adversarial Network (GAN) adalah sebuah arsitektur *neural network* yang terdiri atas dua model, *generator* dan *discriminator*, yang bersaing dengan satu sama lain melalui *adversarial learning* (Goodfellow et al., 2014).

Generator akan memiliki *input layer* yang menerima input noise. Setelah itu, terdapat *transposed convolutional layer* (*deconvolution*) yang digunakan untuk meng-*upsample* input, meningkatkan resolusi spasialnya (Cao et al., 2020). Terdapat juga *layer batch normalization* untuk berguna untuk mengakselerasi dan menstabilkan *training*. Akan ada juga *layer* untuk *activation function* seperti *ReLU* atau *Leaky ReLU* agar *model* dapat mempelajari pola kompleks (Bai, 2022) dan

terakhir *output layer*. Tujuan dari *generator* adalah untuk menghasilkan suatu sampel yang akan diklasifikasikan sebagai data asli oleh *discriminator* (Cohen & Giryes, 2022).

Pada *discriminator*, *input layer* akan menerima sebuah gambar (asli atau hasil *generator*). Kebalikan dari *generator*, *discriminator* akan memiliki *convolutional layers* yang berguna untuk mereduksi resolusi dan mendeteksi fitur dari input (Alzubaidi et al., 2021). Akan terdapat juga *layer batch normalization*, *activation functions*, dan *fully connected layer* yang akan memproduksi suatu *output* yang merepresentasikan probabilitas *input* berupa gambar asli atau palsu (*generated*). Terakhir, *discriminator* akan memiliki *sigmoid activation* untuk menghasilkan sebuah *output* diantara 0 dan 1, dimana semakin *output* mendekati 0, maka input adalah gambar palsu, dan sebaliknya.



Gambar 2.4 Ilustrasi *Block Diagram* Arsitektur GAN.

2.1.5 Deep Convolutional Generative Adversarial Network (DCGAN)

Deep Convolutional Generative Adversarial Network atau DCGAN adalah salah satu jenis *Generative Adversarial Network* (GAN) yang didesain dalam upaya menghasilkan gambar yang lebih realistis. DCGAN memanfaatkan arsitektur *Convolutional Neural Network* atau CNN untuk *generator* dan *discriminator*, agar model dapat menghasilkan dan membedakan gambar dengan optimal (Radford et al., 2015b). DCGAN mampu memahami distribusi gambar yang lebih kompleks dibandingkan metode GAN yang tidak menggunakan *deep convolutional* (Liu et al., 2022).

2.1.6 ShapeNetCore Dataset

ShapeNetCore dataset (Chang et al., 2015) adalah sebuah subset dari *ShapeNet*, suatu *large-scale database* 3D model yang dilabeli dengan baik. *ShapeNetCore* mengandung lebih dari 51.000

objek 3D dengan 55 kelas umum, dimana 12 dari 55 kelas tersebut adalah kategori benda dari *dataset* PASCAL3D+ (Xie et al., 2020).

Tabel 2.1 Kategori-kategori pada Dataset

Airplane	Trash can	Bag	Basket	Bathtub
Bed	Bench	Bicycle	Birdhouse	Bookshelf
Bottle	Bowl	Bus	Cabinet	Camera
Can	Cap	Car	Chair	Clock

Computer	Dishwasher	Display	Earphone	Faucet
Spigot	File	Guitar	Helmet	Jar
Knife	Lamp	Laptop	Loudspeaker	Mailbox
Microphone	Microwave	Motorcycle	Mug	Piano
Pillow	Pistol	Pot	Printer	Remote Control
Rifle	Rocket	Skateboard	Sofa	Stove
Table	Telephone	Tower	Train	Vessel
Washer	Wine Bottle	-	-	-

Beberapa kelas umum tersebut dapat dipecahkan menjadi beberapa *sub-class* lagi, seperti kelas *bench* dapat dipisahkan lagi menjadi *flat bench*, *park bench*, *church bench*, atau *window seat*. Geometri dan tekstur dari setiap model disediakan dalam bentuk orisinilnya ataupun dalam bentuk OBJ/MTL. Untuk versi OBJ, ukuran dan *alignment* sudah dinormalisasi. Pada setiap model, *voxelization* yang *pre-computed* sudah disediakan dengan resolusi 128.



Gambar 2.5 Contoh Objek dari *ShapeNetCore*

Sumber: Chang, A. X., Funkhouser, T., Guibas, L., Hanrahan, P., Huang, Q., Li, Z., Savarese, S., Savva, M., Song, S., Su, H., Xiao, J., Yi, L., & Yu, F. (2015). *ShapeNet: An Information-Rich 3D Model Repository*. <https://doi.org/10.48550/ARXIV.1512.03012>

2.1.7 Chamfer Distance (CD)

Chamfer Distance (CD) adalah sebuah *evaluation metric* yang sering digunakan dalam *computer vision* dan analisa gambar untuk membandingkan kesamaan antara dua gambar (Lin et al., 2023). *Chamfer Distance* menghitung seberapa besar *edges* pada suatu objek mendekati *reference image* dengan cara mengkalkulasi transformasi jarak kedua objek dan menetapkan jarak setiap *pixel* pada objek dengan *pixel background* terdekatnya (Wu et al., 2021).

CD dihitung dengan membandingkan jarak *pixel* demi *pixel* dan menjumlahkannya (Chen et al., 2021) dan semakin kecil nilai tersebut, maka semakin dekat juga kemiripan antara dua gambar.

Chamfer Distance juga dapat digunakan untuk menghitung perbedaan jarak antara dua *point cloud*, sehingga *evaluation metric* tersebut juga sering digunakan dalam *augmented reality*.

2.1.8 Earth Mover's Distance (EMD)

Earth Mover's Distance (EMD) atau *Wasserstein distance* adalah suatu *evaluation metric* yang digunakan untuk menghitung ketidaksamaan atau kesamaan antara dua distribusi probabilitas (Hargreaves et al., 2020). EMD juga bisa diaplikasikan untuk menghitung kesamaan antara dua *point cloud* yang bisa dianggap sebagai sekumpulan *sample* (Rubner et al., 2000).

Nilai EMD didapatkan dengan menghitung seberapa besar jumlah material yang perlu dipindahkan untuk mengubah suatu *point cloud* menjadi *reference point cloud* (Chen et al., 2021). Hal ini berarti semakin kecil nilai EMD, persamaan antara dua *point cloud* semakin tinggi.

2.1.9 Coverage

Coverage adalah sebuah indikator seberapa banyak *point cloud* pada B yang berhasil dicocokkan dengan *point cloud* pada A (G. Yang et al., 2019). *Coverage* dapat dihitung dengan dua *distance metrics* yaitu *Chamfer Distance* (CD) atau *Earth Mover's Distance* (EMD). Nilai *coverage* yang tinggi menandakan bahwa sebagian besar dari B telah terepresentasikan dalam A. *Coverage* dalam GAN digunakan untuk mengukur kreativitas atau kemampuan model GAN untuk menghasilkan *generated set* yang berbeda-beda dan mencegah *overfitting*.

2.1.10 Minimum Matching Distance (MMD)

Minimum Matching Distance adalah *evaluation metric* yang sering digunakan Bersama dengan *Coverage* karena MMD digunakan untuk mengukur kualitas dari *generated point cloud*. Untuk setiap *point cloud* di set referensi, jarak terhadap *nearest neighbor* pada *generated set* dikomputasi dan di rata-rata (G. Yang et al., 2019). MMD dalam GAN digunakan untuk menilai kemiripan *generated set* dengan referensinya.

2.2 Tinjauan Studi

2.2.1 3D point cloud generation reconstruction from single image based on image retrieval (Chen et al., 2021)

Masalah yang diangkat di penelitian ini adalah cara menghasilkan *point cloud* dari beberapa objek yang terdapat dalam satu gambar. Dalam penelitian-penelitian sebelumnya, setiap input fokus terhadap satu objek yang berada di satu gambar.

Metode yang diusulkan penelitian ini adalah perkembangan terhadap *3D-LMNet* dengan tiga tahap. Tahap pertama adalah *target cropping*, yang akan mensegmentasi gambar menjadi beberapa gambar dengan satu objek utama pada setiap gambar. Tahap kedua adalah *image retrieval module*, dimana model akan membandingkan *feature* yang terdeteksi pada input dan berusaha mencari 3D

model dengan *feature* yang termirip. Terakhir adalah tahap rekonstruksi *point cloud* dengan struktur *encoding* dari *PointNet*.

Hasil dari penelitian ini adalah suatu metode *point cloud generation* yang bisa menghasilkan beberapa *point cloud* dari satu gambar dengan nilai *Chamfer Distance* (CD) yang 12.81% lebih rendah dari PGSN (*Points-Guided Sampling Net*) dan nilai *Earth Mover's Distance* (EMD) 14.97% lebih rendah dari PGSN.

Perbedaan dengan penelitian ini adalah metode rekonstruksi *point cloud*. Pada penelitian ini, metode yang digunakan adalah arsitektur *PointNet* yang menghasilkan akan menghasilkan *point cloud*, sedangkan pada skripsi ini, metode yang akan digunakan adalah *structure generator* terhadap input gambar dan digabung menjadi bentuk tiga dimensi.

2.2.2 Flow-based GAN for 3D Point Cloud Generation from a Single Image (Wei et al., 2022)

Masalah yang diangkat di penelitian ini adalah cara menghasilkan *point cloud* dari satu *input* gambar yang seringkali terbatas secara kualitas. Penelitian ini berupaya untuk mengatasi permasalahan tersebut dengan memanfaatkan skema *hybrid explicit-implicit generative modelling*.

Metode yang diusulkan penelitian ini adalah suatu *generative adversarial framework* dengan sebuah *generator* yang dibuat dari *normalizing flow*. Hasil *point cloud* dipastikan serupa dengan *input* secara struktur dengan bantuan *discriminator*.

Hasil dari penelitian ini adalah nilai *Chamfer Distance* (CD) senilai 5.32 yang lebih rendah dibandingkan metode lainnya seperti AtlasNet, Pixel2Mesh, dan MixNFs. Nilai *Earth Mover's Distance* (EMD) yang diraih setara dengan metode PRN sebesar 11.0.

Perbedaan dengan penelitian ini adalah jumlah *input* gambar yang diterima. Pada penelitian ini, jumlah *input* yang diterima berupa satu gambar, sedangkan pada skripsi ini, jumlah *input* yang diterima akan lebih dari satu.

2.2.3 Conditional GAN for Point Cloud Generation (Z. Yang et al., 2023)

Masalah yang diangkat di penelitian ini adalah ketidakmampuan beberapa metode untuk menghasilkan objek sesuai dengan kategori yang diinginkan. Penelitian ini akan memberikan *conditional information* kepada model agar dapat menghasilkan kategori yang diinginkan.

Metode yang diusulkan penelitian ini adalah PC-cGAN yang menggunakan *Conditional GAN* (CGAN) dengan konvolusi network BranchGCN untuk meningkatkan performa *multi-class conditional generation* karena kemampuan BranchGCN untuk mengagregasikan informasi dari *ancestor* dan *neighbor*.

Hasil dari penelitian ini adalah nilai *Chamfer Distance* (CD) dan *Earth Mover's Distance* (EMD) yang lebih rendah dibandingkan beberapa *state-of-the-art* model seperti tree-GAN dan r-GAN. Nilai CD yang diperoleh oleh PC-cGAN pada kelas *airplane* adalah 0.006, dimana performa tree-GAN lebih baik dengan hasil 0.0004. Sedangkan nilai EMD yang diperoleh adalah 0.061, lebih baik dibandingkan nilai tree-GAN yaitu 0.068.

Perbedaan dengan penelitian ini adalah tipe GAN yang digunakan. Penelitian ini menggunakan CGAN dimana *conditional modifier* dapat mengondisikan *output* dengan kategori spesifik. Pada skripsi ini, tipe GAN yang digunakan adalah *Deep Convolutional GAN* (DCGAN).