BAB II
DASAR TEORI

2.1 KAJIAN PUSTAKA

Berdasarkan judul penelitian yang diambil, terdapat beberapa peneltian terdahulu yang dapat dijadikan acuan dalam penulisan penelitian ini. Berikut merupakan rangkuman dari kajian penelitian yang menjadi bahan acuan.

Pada tahun 2017, penelitian berjudul "*FlowNet 2.0: Evolution of Optical flow estimation with Deep Networks*" membahas terkait model *deep learning* yang telah diperbarui untuk estimasi *optical flow* yang bernama *FlowNet2.0*. Penelitian ini memiliki beberapa keunggulan yang signifikan dibandingkan dengan model sebelumnya. Pertama, model ini mencapai kualitas yang lebih baik dengan mengurangi kesalahan estimasi lebih dari 50%. Hal ini menunjukkan bahwa model ini mampu menghasilkan hasil yang lebih akurat dalam memperkirakan pergerakan objek dalam gambar. *FlowNet2.0* juga memiliki kecepatan yang lebih tinggi dengan menggunakan arsitektur bertumpuk dan teknik *warping*, model ini mampu berjalan pada *frame rate* interaktif. Model ini memberikan varian model yang lebih cepat dan memungkinkan perhitungan *optical flow* hingga 140fps dengan akurasi yang sama dengan model asli. Selain itu, penelitian ini juga memberikan kode dan data yang dapat digunakan oleh para peneliti dan praktisi untuk pengembangan lebih lanjut. Ini adalah langkah yang sangat baik dalam mempromosikan reproduktibilitas dan kemajuan ilmiah. Secara keseluruhan, penelitian ini merupakan kontribusi yang berharga dalam bidang estimasi *optical flow*. Dengan kualitas dan kecepatan yang lebih baik, serta ketersediaan kode dan data, FlowNet 2.0 memiliki potensi untuk menjadi standar dalam perhitungan *optical flow* yang akurat dan cepat[6].

Penelitian pada tahun 2019 dari Guo Lu, Wanli Ouyang, Dong Xu, Xiaoyun Zhang, Chunlei Cai, dan Zhiyong Gao yang berjudul "*DVC: An End-to-end Deep Video Compression Framework*" ini membahas terkait sistem kompresi video *end-to-end* yang memadukan teknik kompresi video tradisional dengan *neural network*. *Framework* ini terdiri dari sejumlah modul, seperti kuantisasi, estimasi laju bit, kompresi residual, estimasi gerakan, kompensasi gerakan, dan kompresi gerakan. Pertukaran antara penurunan bit kompresi dan peningkatan kualitas video diperhitungkan ketika modul-modul ini dioptimalkan secara bersama-sama oleh

single loss function. Hasil eksperimen menunjukan bahwa metode yang diusulkan berkinerja lebih baik daripada H.264 dalam hal PSNR dan sebanding dengan H.265 dalam hal MS-SSIM. Dengan membandingkan metode ini dengan teknik sebelumnya dan standar H.264, metode ini berhasil meningkatkan kinerja kompresi video. Metode ini memiliki jumlah parameter yang dapat diatur dan juga efektif secara komputasi [7].

Penelitian ditahun 2019 oleh Oren Rippel, Sanjay Nair, Carissa Lew, Steve Branson, Alexander G. Anderson, dan Lubomir Bourdev yang berjudul "*Learned Video Compression*" membahas terkait penyajian algoritma baru untuk pengkodean video, yang telah dipelajari *end-to-end* untuk mode latensi rendah. Penelitian ini melakukan pengujian kompresi video standar dengan berbagai resolusi, dan membandingkannya dengan semua *codec* komersial utama dalam mode latensi rendah. Hasil dari pengujian tersebut yaitu pada *standard-definition video*, HEVC/H.265, AVC/H.264 dan VP9 menghasilkan kode hingga 60% lebih besar dari algoritma yang didapatkan. Pada *high-definition videos* 1080p, H.265 dan VP9 menghasilkan kode hingga 20% lebih besar, dan H.264 hingga 35% lebih besar. Penelitian ini mengusulkan dua kontribusi utama. Yang pertama adalah arsitektur baru untuk kompresi video, yaitu menggeneralisasi *motion estimation* untuk melakukan kompensasi yang dipelajari, kemudian alih-alih secara ketat bergantung pada *frame* referensi yang ditransmisikan sebelumnya, lebih baik mempertahankan kondisi informasi *arbitrer* yang dipelajari oleh model, dan memungkinkan pengompresan semua sinyal yang ditransmisikan secara bersamaan (seperti *optical flow* dan *residual*). Usulan kedua dari penelitian ini yaitu membuat *framework* spasial berbasis *machine learning* yaitu sebuah mekanisme untuk menetapkan *bitrate* variabel di seluruh ruang untuk setiap *frame* [8]

Berdasarkan penelitian pada tahun 2020 oleh Prasanga Dhungel, Prashant Tandan, Sandesh Bhusal, Sobit Neupane, dan Subarna Shakya yang berjudul "*Video Compression for Surveillance Application using Deep Neural Network*" bertujuan melakukan pendekatan baru untuk kompresi video dengan menyempurnakan kekurangan konvensional yaitu mendekati dan mengganti setiap komponen tradisional dengan mitra jaringan saraf. Penelitian ini menggunakan model *motion estimation, compression and compensation and residue compression*, yang di pelajari dari *end-to-end* untuk meminimalkan pertukaran tingkat distorsi.

Seluruh model tersebut dioptimalkan menggunakan *single loss function*. Penelitian ini didasarkan pada metode standar untuk mengeksploitasi redudansi *spatio-temporal* dalam bingkai video untuk mengurangi laju bit seiring dengan meminimalisasi distorsi dalam bingkai yang didekodekan. Hasil dari penelitian ini yaitu diperoleh nilai MS-SSIM rata-rata 0,98 dan PSNR rata-rata 36 dB bersama dengan *bitrate* rata-rata 0,48. Hasil tersebut mengungguli MPEG standar dalam hal MS-SSIM dan PSNR serta sebanding dengan standar H.264 dalam hal MS-SSIM [5].

Penelitian ditahun 2020 oleh Raz Birman, Yoram Segal, dan Ofer Hadar dengan judul "*Overview of Research in the field of Video Compression using Deep Neural Networks*" ini mencakup elemen-elemen baru yaitu sebuah *auto-encoder* tunggal yang dapat mengompres data gerakan dan residu prediksi secara bersamaan, mempelajari keadaan dari *frame* sebelumnya dan diperbarui secara berulang, koreksi gerakan menggunakan banyak *frame* dan banyak aliran *optic*, serta elemen terakhir yaitu teknik kontrol kecepatan. Hasil dari penelitian ini yaitu ketika diuji dengan MS-SSIM, teknik ini mengungguli perangkat lunak HEVC (HM) [9].

Penelitian ditahun 2020 oleh Zhihao Hu, Zhenghao Chen, dan Dong Xu berjudul "*Improving Deep Video Compression by Resolution-adaptive Flow Coding*" membahas terkait *framework* yang disebut *Resolution-adaptive Flow Coding* (RaFC) yang digunakan untuk mengompresi *flow maps* dengan efektif secara global dan *local*. Penelitian ini menggunakan representasi *multi-resolution* pada masukan juga keluaran dari *flow maps* dan *motion features* dari MV *encoder*. Pengujian dilakukan dengan menggunakan empat *dataset benchmark* HEVC, VTL, UVG, dan MCL-JCV. Hasil dari penelitian ini yaitu menunjukkan keefektifan *framework* RaFC secara keseluruhan setelah menyisir *RaFC-frame* dan *RaFC-block* untuk kompresi video [10].

Penelitian pada tahun 2020 yang dilakukan oleh Zhibo Chen yang berjudul "*Learning for Video Compression*" membahas terkait konsep *PixelMotionCNN* (PMCNN) yang mencakup *motion extension* dan jaringan prediksi hibrida. PMCNN dapat memodelkan *spatiotemporal* koherensi secara efektif dan melakukan pengkodean prediktif. Hasil pengujian menunjukkan bahwa *codec* MPEG-2 mencapai hasil yang sebanding dengan *codec* H.264 [11].

Penelitian yang dilakukan oleh Zachary Teed dan Jin Deng pada tahun 2020 yang berjudul "RAFT: *Recurrent All-Pairs Field Transform For Optical flow*" yang membahas metode baru untuk estimasi aliran optik yang disebut *Recurrent All-Pairs Field Transforms* (RAFT). RAFT menggunakan arsitektur jaringan saraf rekuren yang efisien dan mampu mencapai performa terbaik dalam *benchmark dataset*. Metode ini menggabungkan fitur per-*pixel*, volume korelasi multi-skala, dan pembaruan berulang pada aliran optik. RAFT juga memiliki kemampuan generalisasi yang kuat dan efisien dalam hal waktu inferensi, kecepatan pelatihan, dan jumlah parameter. Penelitian ini menunjukkan bahwa RAFT mengungguli metode-metode sebelumnya dalam hal akurasi dan *general*isasi, serta memberikan analisis komponen penting dalam arsitektur RAFT [12].

Penelitian pada tahun 2021 oleh Guo Lu, Xiaoyun Zhang, Wanli Ouyang, Li Chen, Zhiyong Gao, dan Dong Xu yang berjudul "*An End-to-end Learning Framework for Video Compression*" membahas terkait pendekatan kompresi video tradisional yang dibuat dengan menggunakan *hybrid framework* yaitu *motion-compensated prediction* dan *residual transform coding*. Penelitian ini memanfaatkan arsitektur kompresi klasik dan kemampuan representasi non-linear yang kuat dari jaringan saraf. *Framework* penelitian ini menggunakan informasi gerakan piksel yang dipelajari dari *optical flow network* dan selanjutnya dikompresi oleh *auto-encoder network* untuk menghemat bit. Modul-modul tersebut dioptimalkan dengan menggunakan *rate-distortion trade-off* dan dapat berkolaborasi satu sama lain. Hasil eksperimen yang komprehensif menunjukkan keefektifan kerangka kerja yang diusulkan pada *dataset benchmark* [13].

M. Akin Yilmaz dan A. Murat Tekalp pada tahun 2021 melakukan penelitian yang berjudul "*End-to-end Rate-Distortion Optimization for Bi-Directional Learned Video Compression*". Penelitian ini membahas terkait sebuah *framework* kompresi video yang dioptimalkan secara *end-to-end* dengan menggunakan estimasi aliran hierarkis *bi-directional*. Pendekatan ini melibatkan beberapa komponen, termasuk kompresi gambar, estimasi dan kompresi aliran, kompensasi gerakan, kompresi residu, dan pemrosesan pasca. Terdapat beberapa poin penting dalam penelitian yang dilakukan ini. Pertama, pendekatan ini menggunakan pre-*training* sub-modul jaringan sebelum melakukan optimasi *end-to-end*. Hal ini membantu dalam menghasilkan hasil yang lebih baik. Kedua, penelitian ini

menggunakan augmentasi data seperti pemangkasan acak, rotasi, dan pembalikan temporal selama pelatihan. Ini membantu dalam meningkatkan keberagaman data dan mencegah *overfitting*. Ketiga, penelitian ini menunjukkan bahwa pendekatan yang diusulkan ini dapat dengan mudah menggabungkan hasil baru dalam estimasi aliran optik dan pemodelan konteks untuk meningkatkan kinerja di masa depan. Hasil eksperimen menunjukkan bahwa pendekatan yang diusulkan dalam penelitian ini mengungguli metode kompresi berurutan yang telah ada sebelumnya, seperti x264 dan DVC, serta mendekati kinerja *codec* x265 pada *bitrate* yang lebih tinggi. Hasil visual dari kompresi menggunakan pendekatan ini juga menunjukkan hasil yang lebih halus dan bebas dari artefak blok dibandingkan dengan *codec* tradisional seperti x264 dan x26 [14]

Penelitian ditahun 2021 oleh Ying Liu, Pengli Du, dan Yuzhu Li yang berjudul "*Hierarchical Motion-Compensated Deep Network for Video Compression*" membahas terkait metode *hierarchical motion estimation* dan *compensation network* untuk kompresi video. Metode ini memanfaatkan *inter-frame dan intra-frame* untuk melakukan pengkodean video. *Intra-frame* akan dikompresi secara independen, kemudian sementara *intra-frame* dikompresi, *inter-frame* secara hierarkis diprediksi oleh *frame* yang berdekatan menggunakan *bi-directional motion prediction network.* Kompresi dari *inter-frame* tersebut menghasilkan residu yang rendah dan dapat dikompresi. *Frame* residu kemudian dikompresi melalui jaringan pengkodean residu yang dilatih secara terpisah. Hasil pengujian menunjukkan bahwa pengujian menggunakan metode ini menawarkan efisiensi pengkodean yang jauh lebih tinggi dan kualitas visual yang superior dibandingkan dengan sebelumnya [15].

Jiahao Li, Bin Li, dan Yan Lu melakukan sebuah penelitian pada tahun 2021 berjudul "*Deep Contextual Video Compression*" membahas terkait penggunaan *framework* kompresi video yang menggunakan kondisi kontekstual untuk meningkatkan efisiensi kompresi dan kualitas rekonstruksi video. *Framework* ini memanfaatkan konteks domain fitur sebagai kondisi untuk memanfaatkan konteks berdimensi tinggi guna meningkatkan kualitas video. Hasil eksperimen menunjukkan bahwa metode yang diusulkan mengungguli metode kompresi video berbasis *deep learning* terkini dan mencapai penghematan *bitrate* yang signifikan. DCVC juga dapat mengungguli model DVCPro dalam hal rasio kompresi dan

kualitas rekonstruksi. Selain itu, *framework* ini dapat diperluas dan dirancang secara fleksibel untuk kondisi yang berbeda. Hasil dari pengujian ini menunjukkan bahwa metode ini dapat mengungguli *state-of-the-art* (SOTA) dalam metode kompresi video sebelumnya. Jika dibandingkan dengan x265 yang menggunakan preset sangat lambat, hasil dari penelitian ini dapat mencapai *bitrate* 26,0% untuk video uji standar 1080P [16].

Tabel 2. 1 Rangkuman Kajian Pustaka

No	Judul, Tahun Terbit, Penulis	Metode dan Model Learning yang digunakan	Target dan Hasil
1	**Judul:** *FlowNet 2.0: Evolution of Optical flow estimation with Deep Networks* **Tahun:** 2017 **Penulis:** Guo Lu, Wanli Ouyang, Dong Xu, Xiaoyun Zhang, Chunlei Cai, dan Zhiyong Gao	**Metode:** *FlowNet* ***Model Learning:*** *Convolutional Neural Networks* (CNN)	**Target:** Mengembangkan model *deep learning* yang lebih baik untuk estimasi *optical flow* dan bertujuan untuk meningkatkan kualitas dan kecepatan estimasi *optical flow* dengan memperkenalkan perbaikan pada model *FlowNet* asli **Hasil:** Pengembangan model *FlowNet 2.0* yang berhasil meningkatkan kualitas dan kecepatan estimasi *optical flow*. Dalam penelitian ini, *FlowNet 2.0* mengurangi kesalahan estimasi lebih dari 50% dibandingkan dengan pendekatan sebelumnya dan memiliki

No	Judul, Tahun Terbit, Penulis	Metode dan Model Learning yang digunakan	Target dan Hasil
			performa yang sebanding dengan metode terkini. Selain itu, pengembangan ini juga memperhatikan pergeseran kecil, yang merupakan pergerakan objek yang lebih halus dan sulit untuk diestimasi. Dengan fokus pada pergeseran kecil, FlowNet 2.0 dapat menghasilkan estimasi *optical flow* yang lebih akurat pada pergerakan objek yang halus.
2	**Judul:** DVC: *An End-to-end Deep Video Compression Framework* **Tahun:** 2019 **Penulis:** Guo Lu, Wanli Ouyang, Dong Xu, Xiaoyun Zhang, Chunlei Cai, dan Zhiyong Gao	**Metode:** Pendekatan pembelajaran *end-to-end* untuk kompresi video yang mencakup beberapa komponen, termasuk jaringan kompensasi gerakan, jaringan enkoder dan dekoder gerakan, jaringan enkoder dan dekoder residual, serta jaringan enkoder dan dekoder aliran optik.	**Target:** Mengembangkan sistem kompresi video *end-to-end* menggunakan *deep learning* untuk mengoptimalkan kompresi video dengan mempertimbangkan gerakan dan sisa informasi untuk mencapai kualitas rekonstruksi yang sangat baik dengan jumlah bit yang lebih sedikit.

No	Judul, Tahun Terbit, Penulis	Metode dan Model Learning yang digunakan	Target dan Hasil
		Model Learning: *Convolutional Neural Networks* (CNN)	**Hasil:** Mencapai hasil MS-SSIM sebesar 0.980 pada *dataset* HEVC Class B, 0.990 pada *dataset* HEVC Class E, dan 0.965 pada *dataset* UVG. Mencapai hasil PSNR sebesar 35.0 dB pada *dataset* HEVC *Class* B dengan *bitrate* 0.35 Bpp, 41.0 dB pada *dataset* HEVC *Class* E dengan bit rate 0.30 Bpp, dan 38.8 dB pada *dataset* UVG dengan *bitrate* 0.1 Bpp.
3	**Judul:** *Learned Video Compression* **Tahun:** 2019 **Penulis:** Oren Rippel, Sanjay Nair, Carissa Lew, Steve Branson, Alexander G. Anderson, dan Lubomir Bourdev	**Metode:** DenseNet dan Dual Path ***Model Learning*:** *Convolutional Neural Networks* (CNN)	**Target:** Mengembangkan algoritma kompresi video yang dapat mengungguli *codec* video komersial dalam hal efisiensi kompresi dan kualitas visual dan mengoptimalkan penggunaan *deep learning* dan kontrol laju spasial dalam proses kompresi video.

No	Judul, Tahun Terbit, Penulis	Metode dan Model Learning yang digunakan	Target dan Hasil
			Hasil: 1. Pada *standard-definition video*, HEVC/H.265, AVC/H.264 dan VP9 menghasilkan kode hingga 60% lebih besar dari algoritma yang didapatkan. 2. Pada *high-definition videos* 1080p, H.265 dan VP9 menghasilkan kode hingga 20% lebih besar, dan H.264 hingga 35% lebih besar.
4	**Judul:** *Video Compression for Surveillance Application using Deep Neural Network* **Tahun:** 2020 **Penulis:** Prasanga Dhungel, Prashant Tandan, Sandesh Bhusal, Sobit Neupane, dan Subarna Shakya	**Metode:** *Long Short Term Memory* ***Model Learning*:** *Convolutional Neural Networks* (CNN)	**Target:** Mengembangkan metode kompresi video menggunakan *deep learning* yang dapat mengurangi *bitrate* dan distorsi dalam *frame* video, khususnya untuk video pengawasan **Hasil:** Mengungguli MPEG standar dengan memperoleh MS-SSIM rata-rata 0,98 dan PSNR

No	Judul, Tahun Terbit, Penulis	Metode dan Model Learning yang digunakan	Target dan Hasil
			rata-rata 36dB bersama dengan *bitrate* rata-rata 0,48. Pendekatan tersebut unggul dalam hal MS-SSIM dan PSNR.
5	**Judul:** *Overview of Research in the field of Video Compression using Deep Neural Networks* **Tahun:** 2020 **Penulis:** Raz Birman, Yoram Segal, dan Ofer Hadar	**Metode:** *Autoencoders* untuk melakukan super-resolusi pada komponen luma. ***Model Learning*:** *Deep Neural Networks* (DNN)	**Target:** Mengembangkan algoritma kompresi video yang menggunakan metode *Deep Neural Networks* (DNN) dan teknik *Autoencoders*. **Hasil:** Ketika diuji dengan MS-SSIM, teknik ini mengungguli perangkat lunak HEVC.
6	**Judul:** *Improving Deep Video Compression by Resolution-adaptive Flow Coding* **Tahun:** 2020 **Penulis:** Zhihao Hu, Zhenghao Chen, dan Dong Xu	**Metode:** RaFC (*Resolution-adaptive Flow Coding*) yang digabungkan dengan teknik optimasi *rate-distortion* (RD) ***Model Learning*:** *Convolutional Neural Networks* (CNN)	**Target:** Mengembangkan metode kompresi video berbasis pembelajaran mendalam menggunakan RaFC (*Resolution-adaptive Flow Coding*) *framework.* **Hasil:** Menunjukkan bahwa metode RaFC dapat menghasilkan video dengan kualitas yang hampir sebanding dengan metode kompresi

No	Judul, Tahun Terbit, Penulis	Metode dan Model Learning yang digunakan	Target dan Hasil
			video standar seperti H.264 dan H.265, sementara tetap mencapai tingkat kompresi yang lebih tinggi.
7	**Judul:** *Learning for Video Compression* **Tahun:** 2020 **Penulis:** Zhibo Chen	**Metode:** PixelMotion ***Model Learning*:** *Convolutional Neural Networks* (CNN)	**Target:** Menciptakan teknik estimasi aliran optik yang inovatif, akurat, dan efisien dengan menggunakan arsitektur jaringan saraf tiruan. Berkenaan dengan waktu inferensi, kecepatan pelatihan, dan jumlah parameter, proyek ini berusaha untuk mendapatkan kinerja terbaik pada *dataset benchmark*, memiliki kemampuan generalisasi yang baik, dan efisien. **Hasil:** Berhasil menggabungkan *Convolutional Neural Network* (CNN) dan model *pixel motion* dan mampu mengungguli *codec video modern* seperti MPEG-2 dengan pengurangan BD-*Rate*

No	Judul, Tahun Terbit, Penulis	Metode dan Model Learning yang digunakan	Target dan Hasil
			sebesar 48.415% dan peningkatan BD-PSNR sebesar 2.39dB. Selain itu, PMCNN juga menunjukkan hasil yang sebanding dengan *codec* H.264 dengan peningkatan BD-Rate sekitar 8.175% dan penurunan BD-PSNR sebesar 0.41dB.
8	**Judul:** RAFT: *Recurrent All-Pairs Field Transform for Optical flow* **Tahun:** 2020 **Penulis:** Zachary Teed dan Jin Deng	**Metode:** *FlowNet2.0* ***Model Learning*:** *Recurrent Neural Network* (RNN) dan *Convolutional Neural Network* (CNN)	**Target:** Mengembangkan metode yang efisien dan akurat untuk estimasi aliran optik menggunakan arsitektur baru RAFT dan bertujuan untuk mencapai performa terbaik dalam *benchmark dataset*, memiliki kemampuan generalisasi yang kuat, dan efisien dalam hal waktu inferensi, kecepatan pelatihan, dan jumlah parameter. **Hasil:** Metode RAFT berhasil mencapai performa terbaik dalam

No	Judul, Tahun Terbit, Penulis	Metode dan Model Learning yang digunakan	Target dan Hasil
			estimasi aliran optik pada *dataset* Sintel dan KITTI dan memiliki kemampuan generalisasi yang kuat, efisien dalam hal waktu inferensi, kecepatan pelatihan, dan jumlah parameter. RAFT berhasil mengungguli metode-metode sebelumnya dalam hal akurasi dan generalisasi.
9	**Judul:** *An End-to-end Learning Framework for Video Compression* **Tahun:** 2021 **Penulis:** Guo Lu, Xiaoyun Zhang, Wanli Ouyang, Li Chen, Zhiyong Gao, dan Dong Xu	**Metode:** Model DVC_Pro. Model DVC_Lite, dan Model DVC_Pro_AQ ***Model Learning*:** *Convolutional Neural Network* (CNN)	**Target:** Mengembangkan model kompresi video berbasis *deep learning* yang efisien dan memiliki kinerja yang baik pada berbagai tingkat *bitrate* dan bertujuan untuk meningkatkan kualitas kompresi video dan mengurangi ukuran *file* video tanpa mengorbankan kualitas visual. **Hasil:** Menunjukkan bahwa model-model yang diusulkan memiliki kinerja yang baik dalam

No	Judul, Tahun Terbit, Penulis	Metode dan Model Learning yang digunakan	Target dan Hasil
			kompresi video pada berbagai tingkat *bitrate*.
10	**Judul:** *End-to-end Rate-Distortion Optimization for Bi-Directional Learned Video Compression* **Tahun:** 2021 **Penulis:** M. Akin Yilmaz dan A. Murat Tekalp	**Metode:** *Frame*work ini menggunakan komponen seperti *hierarchical bi-directional flow estimation, flow compression*, dan *frame prediction* yang dirancang menggunakan *filter* konvolusi yang dapat dipelajari dan dioptimalkan secara *end-to-end* dengan menggunakan *single rate-distortion loss*. Menggunakan *Motion Compensation Net* pada CNN. ***Model Learning*:** *Artificial Neural Network* (ANN) dan *Convolutional Neural Network* (CNN)	**Target:** Mengurangi jumlah bit yang diperlukan untuk merepresentasikan *Group of Picture* (GoP) dalam kompresi video sambil mempertahankan kualitas gambar yang baik. **Hasil:** Berhasil menghasilkan kinerja yang superior dibandingkan dengan metode kompresi video yang telah ada sebelumnya, seperti x264 dan DVC, serta mendekati kinerja *codec* x265 pada *bitrate* yang lebih tinggi dan menunjukkan bahwa penggunaan prediksi *frame bi-directional* dapat meningkatkan kinerja kompresi dibandingkan dengan prediksi *frame uni-directional* dengan hasil yang lebih halus dan

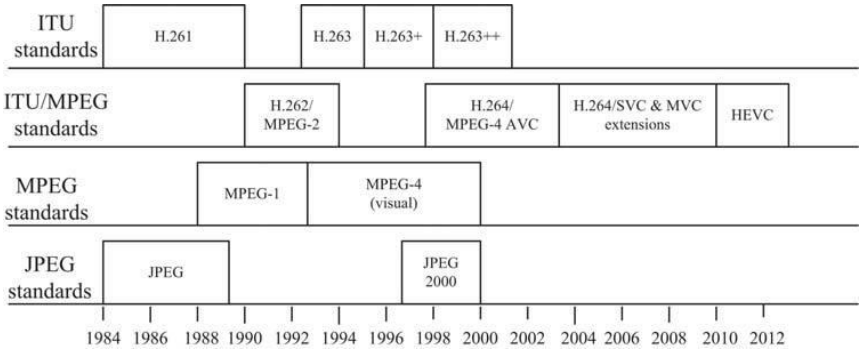
No	Judul, Tahun Terbit, Penulis	Metode dan Model Learning yang digunakan	Target dan Hasil
			bebas dari artefak blok dibandingkan dengan *codec* tradisional seperti x264 dan x265.
11	**Judul:** *Hierarchical Motion-Compensated Deep Network for Video Compression* **Tahun:** 2021 **Penulis:** Ying Liu, Pengli Du, dan Yuzhu Li	**Metode:** *Hierarchical Coding Structure* ***Model Learning*:** *Convolutional Neural Network* (CNN)	**Target:** Mengembangkan metode kompresi video yang efisien menggunakan struktur hierarkis dengan *Convolutional Neural Network* (CNN). **Hasil:** Mampu menghasilkan kualitas video yang baik dengan ukuran *file* yang lebih kecil dibandingkan metode kompresi video tradisional. Penelitian ini juga menunjukkan bahwa penggunaan struktur hierarkis dan CNN dapat meningkatkan efisiensi kompresi video dibandingkan dengan metode kompresi video *intra-frame* dan *inter-frame* yang hanya memiliki satu lapisan prediksi dan pengkodean *residue*.

No	Judul, Tahun Terbit, Penulis	Metode dan Model Learning yang digunakan	Target dan Hasil
12	**Judul:** *Deep Contextual Video Compression* **Tahun:** 2021 **Penulis:** Jiahao Li, Bin Li, dan Yan Lu	**Metode:** *Deep Contextual Video Compression* (DCVC). *Conditional Coding*, *Hyper Prior Encoder* (HPE), *Hyper Prior Decoder* (HPD), *Arithmetic Encoder* (AE), dan *Arithmetic Decoder* (AD) ***Model Learning*:** *Recurrent Neural Network* (RNN) dan *Convolutional Neural Network* (CNN)	**Target:** Mengembangkan sebuah *frame*work kompresi video yang menggunakan kondisi kontekstual untuk meningkatkan efisiensi kompresi dan kualitas rekonstruksi video. **Hasil:** Pada berbagai *dataset* video dengan berbagai resolusi dan karakteristik konten yang berbeda, DCVC mampu menghasilkan penghematan *bitrate* hingga 26%. Selain itu, DCVC juga dapat mengungguli metode-metode terbaru dalam kompresi video seperti RY_CVPR20, LU_ECCV20, dan HU_ECCV20.
13	**Judul:** *Flow estimation Video Compression Using Deep learning* **Tahun:** 2023	**Metode:** RAFT (*Recurrent All-Pairs Field Transform*) ***Model Learning*:** *Recurrent Neural Network* (RNN) dan	**Target:** Mengetahui penerapan *flow estimation* pada kompresi video meggunakan *deep learning* dan mendapatkan hasil kompresi yang baik.

No	Judul, Tahun Terbit, Penulis	Metode dan Model Learning yang digunakan	Target dan Hasil
	Penulis: Leliza Febrianti C	*Convolutional Neural Network* (CNN)	**Hasil:** Hasil kompresi terbaik yaitu pada *epoch* ke 100 dengan nilai PSNR yaitu 32 dB dan nilai MSE 0.001

2.2 DASAR TEORI

2.2.1 PERKEMBANGAN FORMAT VIDEO



Gambar 2. 1 Garis Sejarah Standar Format Video oleh ITU-T dan ISO/IEC [17]

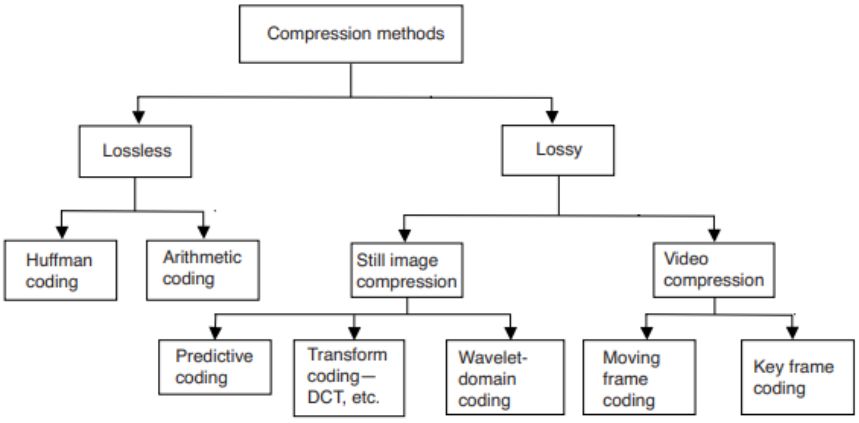
Serangkaian bingkai gambar dapat ditransmisikan, direkam, dan dirangkai menggunakan teknologi video untuk menghasilkan gerakan. *Frame* dapat berbentuk video komposit yang mengandung elemen warna, kecerahan, dan sinkronisasi gambar, atau dapat juga berbentuk gelombang analog. Banyak orang menikmati aspek visual dan pendengaran dari video karena memungkinkan untuk mengasimilasi informasi secara alami dan efektif. *Ampex Quadruplex*, format video pertama, diperkenalkan pada tahun 1956, menandai dimulainya era format video. Membandingkan nama format ini dengan *codec* video yang terkenal seperti AVI dan MP4, kedengarannya aneh dan sulit untuk diucapkan. Seiring berjalannya waktu, berbagai format video inovatif, termasuk kaset HDTV W-VHS *Betamax* dan JVC, gagal mendapatkan daya tarik di pasar. Berbagai format penyimpanan video, termasuk kaset digital, *laserdisc*, dan DVD, juga dikembangkan selama masa ini dan masih digunakan sampai sekarang [18].

Sektor standardisasi telekomunikasi dari *International Telecommunication Union* (ITU-T) menciptakan standar video digital pertama yaitu H.120 pada tahun 1984. Penciptaan format *file* video digital dimulai dengan standar ini. Standar kompresi video digital pertama, H.261, dibuat oleh ITU-T pada tahun 1988 dan berfungsi sebagai fondasi untuk berbagai format dan *codec* video lainnya [19].

Standar video MPEG-1, dengan kecepatan bit 1,5 Mbit/dtk, dikembangkan oleh *Movie Picture Experts Group* (MPEG) pada tahun 1991 untuk kompresi video kualitas VHS dan *compact disc*. Meskipun MPEG menawarkan rekomendasi untuk *bitstreaming* dan *decoding* video, MPEG tidak menyebutkan teknik kompresi atau penyandian video. Microsoft memperkenalkan format *file* video AVI pada tahun 1992, yang sangat disukai tetapi memiliki beberapa kekurangan. Pada tahun 1994, MPEG-2 atau H.262 mulai tersedia, menawarkan resolusi yang lebih besar dan kecepatan bit yang lebih tinggi. *Codec* video ini berevolusi menjadi standar industri untuk DVD dan televisi digital. Real Network menciptakan Real Media pada tahun 1997 untuk *streaming* media melalui internet berdasarkan standar H.263. Ketika MPEG-4 diperkenalkan pada tahun 1998, MPEG-4 dengan cepat menjadi standar industri untuk video definisi tinggi dan mendukung teknologi mutakhir termasuk video 3D, *Digital Rights Management* (DRM), dan resolusi tinggi. Pada tahun 2001, MP4 juga muncul dan menjadi kontainer standar untuk menonton video MPEG-4 Part 14. Standar ini dibuat langsung berdasarkan format *Quicktime*, namun juga menyediakan dukungan untuk fitur-fitur yang dimiliki oleh MPEG, seperti *Initial Object Descriptors* [20].

Format video lain yang sesuai untuk ponsel juga telah berkembang seiring dengan kebutuhan kamera ponsel, seperti 3GP, spesifikasi dan standar video telekomunikasi seluler yang dikembangkan oleh 3rd *Generation Partnership Project*. MKV adalah format video gratis dan terbuka yang juga banyak digunakan. Format ini dapat menyimpan video, musik, teks film, dan gambar dalam jumlah yang tidak terbatas dalam satu *file*. Format WebM, yang digunakan untuk video dalam HTML5, terungkap didasarkan pada format MKV pada tahun 2010. Dengan demikian, sejak diperkenalkannya format video pertama pada tahun 1956 hingga format video kontemporer yang digunakan saat ini, perkembangan format dan teknologi video terus berlanjut [17].

2.2.2 KOMPRESI VIDEO



Gambar 2. 2 Klasifikasi Teknik Kompresi Citra [18]

Kompresi video adalah metode untuk mengurangi ukuran video dengan mengurangi ukuran gambar dan suara yang ada dalam video hingga kualitas yang tetap dianggap baik. Video memiliki tiga dimensi, yaitu dua dimensi spasial dan satu dimensi waktu. Dalam video, ada dua elemen yang dapat dikompresi, yaitu *frame* (gambar diam) dan audio. Redundansi data dalam video terjadi baik pada tingkat spasial dengan perubahan warna dalam gambar diam, maupun pada tingkat temporal dengan perubahan antar *frame*. Pengurangan redundansi spasial atau kompresi antar *frame* dilakukan dengan memanfaatkan fakta bahwa manusia cenderung lebih sensitif terhadap kecerahan daripada perbedaan warna, sehingga gambar dalam video dapat dikompresi. Sementara itu, pengurangan redundansi temporal atau kompresi *interframe* dilakukan dengan hanya mengirim dan mengkodekan *frame* yang berubah, sementara data yang sama pada *frame* lain tetap disimpan. Terdapat 2 jenis teknik kompresi, yaitu:

a. *Lossless Compression*

   Lossless compression merujuk pada teknik kompresi gambar dengan kondisi tidak ada kehilangan informasi, sehingga gambar yang didekompresi menjadi identik dengan gambar aslinya. Meskipun demikian, jenis kompresi ini memiliki tingkat kompresi yang sangat rendah. Beberapa aplikasi memerlukan kompresi tanpa kehilangan, seperti radiografi, kompresi gambar diagnostik medis, atau kompresi foto satelit, di mana setiap kehilangan informasi bisa menghasilkan hasil yang tidak diinginkan. Beberapa contoh teknik kompresi

lossless termasuk *Run Length Encoding* (RLE), *Entropy Coding* (Huffman, aritmatik), dan *Adaptive Dictionary Based* (LZW) [18].

b. *Lossly Compression*

Lossly compression adalah teknik kompresi citra dengan hasil dekompresi dari citra yang telah dikompresi tidak akan sama persis dengan citra aslinya karena ada beberapa informasi yang hilang. Namun, tingkat kehilangan tersebut masih dapat diterima oleh persepsi mata manusia, karena mata tidak dapat membedakan perubahan kecil pada gambar. Metode ini memungkinkan untuk mencapai rasio kompresi yang lebih tinggi dibandingkan dengan metode *Lossless*. Contoh metode *Lossy compression* meliputi reduksi warna, *chroma sub-sampling*, dan pengkodean transformasi seperti transformasi *Fourier, wavelet*, dan teknik lainnya [18].

2.2.3 *ARTIFICIAL INTELLIGENCE* (AI)

Artificial Intelligence (AI) atau yang dikenal juga dengan kecerdasan buatan merupakan salah satu sistem komputer yang di *program* dalam mesin agar bisa berpikir seperti manusia. *Artificial Intelligence* dibuat dengan tujuan untuk memudahkan manusia dalam menyelesaikan tugas-tugasnya. Sebagaimana manusia, *Artificial Intelligence* adalah teknologi yang membutuhkan banyak data untuk menjadi cerdas. *Artificial Intelligence* bekerja sesuai dengan pemrograman yang telah dibuat pada sistem computer. Algoritma tersebut berfungsi sebagai kerangka berpikir dalam memproses banyak data yang masuk. Peningkatan kecerdasannya, AI membutuhkan banyak data dan pengalaman. *Learning, reasoning, and self-correction* adalah langkah penting dalam proses pembentukan AI [21]. Secara umum, kecerdasan buatan dapat melakukan 4 tugas yang tercantum dibawah ini:

1. Sistem yang dapat bertindak selayaknya manusia yang disebut dengan *Acting Humanly*.
2. Sistem yang bisa berpikir selayaknya manusia yang disebut dengan *Thinking Humanly*.
3. Sistem yang dapat berpikir secara rasional yang disebut dengan *Think Rationally*.

4. Sstem yang mampu bertindak secara rasional yang disebut dengan *Act Rationally*.

Selain tugas-tugas tersebut kecerdasan buatan juga memiliki beberapa keunggulan yaitu sebagai berikut:

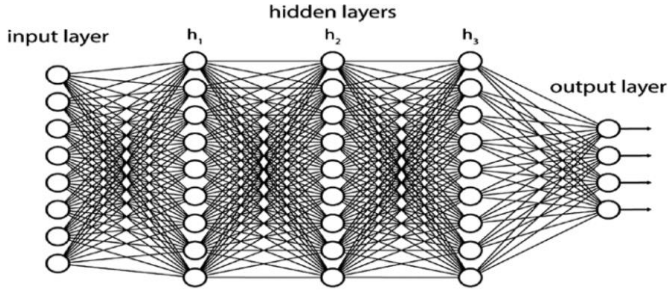
1. Tidak dapat berubah atau bersifat permanen.
2. Dapat menyimpan banyak data tanpa adanya batasan waktu.
3. Dapat ditransfer dan diduplikasi ke komputer lain.
4. Dapat mempemudah pekerjaan yang kompleks.
5. Lebih cepat dan akurat.
6. Penggunaannya dapat dilakukan dilakukan terus menerus tanpa batasan waktu.

Selain kelebihan yang telah disebutkan, *Artificial Intelligence* juga memiliki memiliki beberapa kekurangan yaitu:

1. Kecerdasan yang dimiliki hanya bergantung pada *program* yang telah dibuat.
2. Kecerdasan buatan tidak dapat mengembangkan pengetahuannya sendiri karena bergantung pada sistem yang dibangun[21].

2.2.4 *DEEP LEARNING*

Deep learning adalah sub-bidang *machine learning* yang menggunakan jaringan syaraf tiruan dengan banyak lapisan yang saling berhubungan untuk meramalkan masa depan atau membuat keputusan hari ini. *Deep learning* berbeda dari *machine learning* karena dapat mengekstraksi pola yang lebih rumit dari data sebab jaringan saraf tiruan yang digunakan dapat memproses data dengan banyak lapisan. Perbedaan lainnya antara *deep learning* dan *machine learning* yaitu *deep learning* dapat melakukan pelabelan dari data seperti gambar, video, atau teks secara otomatis layaknya prmikiran manusia. *Deep learning* sering digunakan untuk tugas pemrosesan data yang rumit, seperti pemrosesan wajah, suara, dan bahasa alami [22]



Gambar 2. 3 ***Layer Deep Neural Network*** [23]

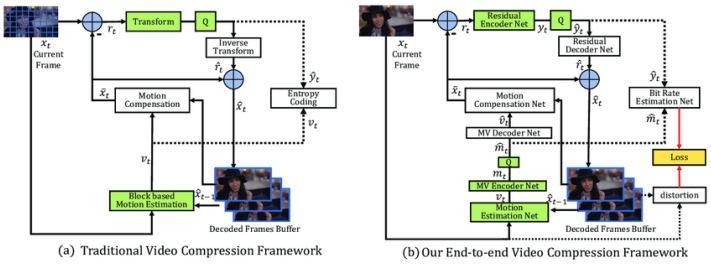
Deep learning telah berevolusi dengan definisi algoritma yang lebih luas dan mampu memahami berbagai tingkat representasi data sesuai dengan hierarki kompleksitas yang berbeda, meskipun dimulai dibidang yang mirip dengan *machine learning*, fokus utama dari algoritma ini adalah kendala kepuasan untuk berbagai tingkat kompleksitas. Dengan kata lain, algoritma ini dapat mempelajari berbagai tingkat kompleksitas selain kemampuan prediksi dan klasifikasinya. Struktur dari model *Deep learning* yang sering digunakan yaitu menampilkan lapisan *neuron*. Lapisan *neuron* merupakan unit non-linier yang digunakan untuk memproses data. Setiap lapisan dalam model ini memproses data pada tingkat abstraksi yang berbeda [22].

Deep learning dapat digunakan dalam proses kompresi video dengan cara menggunakan jaringan saraf tiruan untuk mengenali dan menghilangkan informasi yang tidak penting dari video. Hal ini dapat mengurangi ukuran *file* video tanpa menurunkan kualitas video yang signifikan. Salah satu contoh penggunaan *Deep learning* dalam kompresi video adalah dengan menggunakan jaringan saraf tiruan untuk mengenali objek yang ada dalam video dan menghilangkan informasi yang tidak penting dari objek tersebut. Contohnya, jika sebuah video menampilkan sebuah gedung di latar belakang, maka jaringan saraf tiruan dapat mengenali gedung tersebut dan menghilangkan informasi yang tidak penting dari gedung, seperti warna langit di latar belakang atau pohon di sekitarnya. Dengan cara ini, ukuran *file* video dapat dikompresi tanpa menurunkan kualitas video yang signifikan. Selain itu, dengan memanfaatkan kemampuan jaringan saraf tiruan untuk mempelajari dan menangkap pola-pola yang terdapat dalam data, *Deep learning* dapat digunakan untuk mencari cara terbaik dalam mengompresi video dengan mengoptimalkan algoritma kompresi yang ada [22].

2.2.4.1 *OPTICAL FLOW ESTIMATION*

Metode yang digunakan untuk memperkirakan mobilitas atau aliran objek dalam video disebut *optical flow estimation*. Tepatnya, *optical flow estimation* adalah algoritma yang digunakan untuk mengestimasi gerakan objek dari *frame* ke *frame* dalam sebuah video. *Optical flow estimation* juga bekerja untuk memperkirakan perbedaan antara *frame* video dan menggunakan informasi tersebut untuk mengompresi video secara lebih efektif. Perkiraan perbedaan antara *frame*

video, arsitektur *optical flow estimation* melewati sejumlah proses, termasuk mendeteksi karakteristik yang ada dalam *frame* video, memperkirakan gerakan objek, dan menggabungkan informasi gerakan dengan data dari *frame* sebelumnya. Algoritma ini dapat digunakan dalam proses kompresi video untuk mengurangi ukuran *file* video dengan mengompresi informasi gerakan objek dari *frame* ke *frame* tersebut [13].



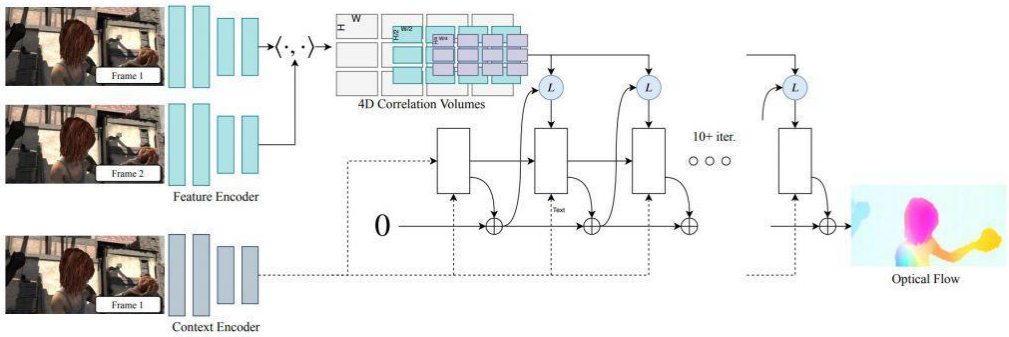
Gambar 2. 4 Perbandingan arsitektur *flow estimation* [13]

Deep learning dapat digunakan untuk meningkatkan kemampuan algoritma *optical flow estimation* dalam mengestimasi gerakan objek dari *frame* ke *frame* dengan cara menggunakan jaringan saraf tiruan untuk mempelajari dan menangkap pola-pola yang terdapat dalam data gerakan objek. Dengan demikian, jaringan saraf tiruan dapat menghasilkan estimasi gerakan yang lebih akurat dan tepat dibandingkan dengan algoritma *optical flow estimation* tradisional. Selain itu, *Deep learning* juga dapat digunakan untuk mengoptimalkan algoritma *optical flow estimation* yang ada dengan cara memanfaatkan kemampuan jaringan saraf tiruan untuk mempelajari dan menangkap pola-pola yang terdapat dalam data. Dengan cara ini, algoritma *optical flow estimation* yang dioptimalkan dengan *Deep learning* dapat menghasilkan estimasi gerakan yang lebih akurat dan tepat, sehingga dapat mengurangi ukuran *file* video tanpa menurunkan kualitas video yang signifikan. Secara umum, penggunaan *Deep learning* dalam algoritma *optical flow estimation* dapat membantu meningkatkan kemampuan algoritma untuk mengestimasi gerakan objek dari *frame* ke *frame*, sehingga dapat mengurangi ukuran *file* video tanpa menurunkan kualitas video yang signifikan [5].

2.2.4.2 *RECURRENT ALL-PAIRS FIELD TRANSFORMS* (RAFT)

Pendekatan terbaru yang dianggap canggih, berdasarkan penilaian menggunakan standar SINTEL, melibatkan kombinasi antara arsitektur CNN dan RNN yang diperkenalkan pada tahun 2020. Pendekatan inovatif ini dikenal dengan

nama *Recurrent All-Pairs Field Transforms* (RAFT). RAFT adalah sebuah arsitektur jaringan dalam yang digunakan untuk estimasi aliran optik. RAFT melakukan ekstraksi fitur per piksel, membangun volume korelasi multi-skala untuk semua pasangan piksel, dan secara iteratif memperbarui medan aliran melalui unit rekurensi yang melakukan pencarian pada volume korelasi. RAFT telah mencapai kinerja terbaik pada *dataset benchmark* seperti KITTI dan Sintel, serta memiliki kemampuan generalisasi yang kuat dan efisiensi tinggi dalam hal waktu inferensi, kecepatan pelatihan, dan jumlah parameter[12]



Gambar 2. 5 Arsitektur RAFT [12]

Gambar 2.6 merupakan arsitektur dari RAFT. RAFT memiliki 3 arsitektur yang terdiri dari:

1. *Feature Extraction*
   Proses ini melibatkan jaringan yang terdiri dari dua frame berurutan. Untuk mengambil ciri dari kedua gambar ini, digunakan dua jaringan CNN dengan bobot yang identik. Pendekatan ini menyerupai struktur arsitektur FlowNetCorr, di mana ekstraksi fitur dari kedua gambar dilakukan secara terpisah. Arsitektur CNN terdiri dari 6 lapisan residual, sejajar dengan konfigurasi lapisan ResNet, dengan resolusi yang terus berkurang pada setiap lapisan berikutnya, sambil meningkatkan jumlah saluran [12].
2. *Visual Similarity*
   Kesamaan visual dihitung sebagai hasil perkalian dalam dari seluruh pasangan peta fitur. Akibatnya, akan terbentuk sebuah tensor 4D yang disebut sebagai volume Korelasi, yang memberikan informasi penting tentang pergeseran piksel dalam skala kecil hingga besar. Pendekatan ini harus dibedakan dari lapisan Korelasi dalam FlowNetCorr. Pada FlowNetCorr, kita menggunakan

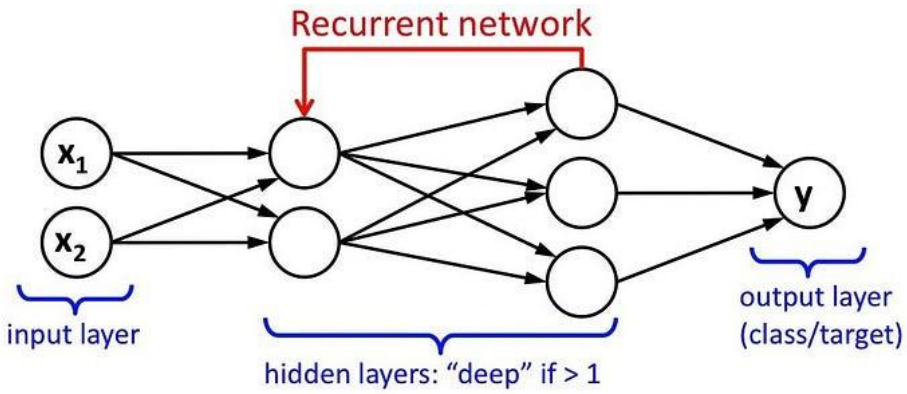
korelasi berbasis patch, sementara dalam RAFT, korelasi dihitung untuk semua pasangan peta fitur tanpa adanya jendela ukuran tetap. Setelah itu, piramida korelasi 4 lapis dibangun dengan menggabungkan dua dimensi terakhir dari tensor 4D ini dengan kernel ukuran 1, 2, 4, 8. Piramida Korelasi digunakan untuk menciptakan fitur kemiripan gambar multi-skala untuk membuat pergerakan mendadak lebih terlihat. Oleh karena itu, Piramida memberikan informasi mengenai perpindahan yang kecil dan besar [12].

3. *Iterative Update*

   Pembaruan iteratif adalah urutan sel *Gated Recurrent Unit* (GRU) yang menggabungkan semua data yang telah kita hitung sebelumnya. Sel GRU meniru algoritme pengoptimalan berulang dengan satu perbaikan - ada lapisan konvolusi yang dapat dilatih dengan bobot bersama di sana [12].

2.2.4.2.1 *RECURENT NEURAL NETWORK*

Recurrent neural networks (RNN) adalah model yang dibuat untuk memecahkan masalah yang berkaitan dengan pengenalan pola. Jaringan ini dibangun berdasarkan konsep yang sama dengan MLP, tetapi dengan siklus terarah yaitu *input* diubah menjadi *output*. Tidak seperti MLP, RNN tidak memiliki banyak lapisan. *Recurrent Neural Network* menggunakan penalaran dari pengalaman sebelumnya untuk menginformasikan kejadian yang akan datang. Kemampuan model *Recurrent Neural Network* untuk mengurutkan vektor sangat berguna, yang membuka API untuk melakukan tugas-tugas yang lebih rumit [25].



Gambar 2. 6 Lapisan *Recurrent Neural Network* [24]

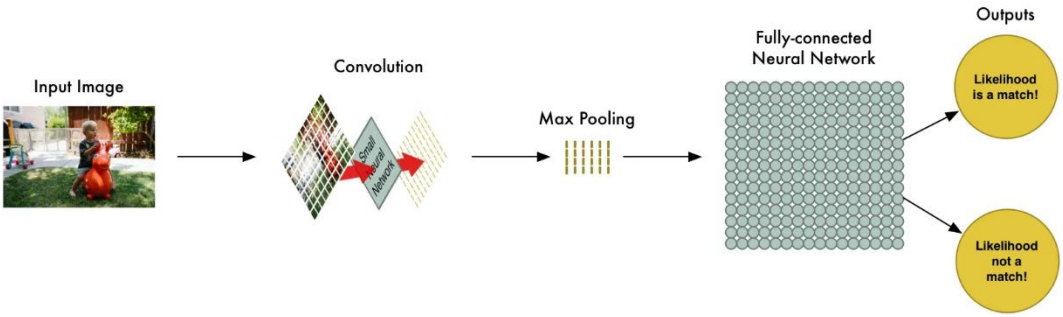
Recurrent Neural Networks dapat dianggap sebagai serangkaian jaringan yang saling terhubung. Jaringan ini sering kali memiliki arsitektur seperti rantai, sehingga dapat digunakan untuk tugas-tugas seperti pengenalan suara, penerjemahan bahasa, dll. RNN dapat dirancang untuk beroperasi di seluruh urutan vektor dalam *input*, *output*, atau keduanya. Sebagai contoh, *input* yang diurutkan dapat mengambil sebuah kalimat sebagai *input* dan menghasilkan nilai sentimen positif atau negatif. Atau, *output* yang diurutkan dapat mengambil gambar sebagai *input*, dan menghasilkan kalimat sebagai *output*. RNN memiliki arsitektur yang dirancang khusus untuk data berbentuk sekuensial dan list. RNN memiliki beberapa keunggulan yaitu mampu menangani data yang memiliki keterkaitan dalam waktu, mampu mengngiat konteks sebelumnya, mampu menggenerasi teks, dan mampu menangani data yang tidak linear. Berdasarkan konsep dan cara kerja tersebut RNN telah menunjukkan keberhasilan yang cukup besar beberapa tahun terakhir dalam 31 menyelesaikan masalah termasuk *speech recognition, machine translation, sentiment analysis, image captioning,* dan masih banyak lagi [24].

2.2.4.2.2 *CONVOLUTIONAL NEURAL NETWORK* (*CNN*)

Convolutional Neural Network (CNN) merupakan salah satu algoritma dari *deep learning* yang sering digunakan untuk menyelesaikan masalah yang menantang. CNN mampu mengidentifikasi objek secara efisien dan mempelajari atribut yang sangat abstrak. Selain itu CNN juga mampu mengatasi kelemahan teknik pembelajaran mesin konvensional dan banyak digunakan di berbagai bidang, termasuk klasifikasi gambar, deteksi objek, pengenalan suara, dan masih banyak lagi [22].

Convolutional Neural Network (CNN) bekerja dengan cara mengekstraksi informasi lokal menggunakan *receptive field*, yang merupakan sebuah koneksi lokal dari *neuron-neuron* di lapisan sebelumnya. Untuk membuat *feature map, weight vector* atau *filter* yang terhubung ke *neuron* tertentu pada *layer* berikutnya akan digeser ke atas *input vector*. Dengan menurunkan jumlah parameter yang perlu dilatihkan, metode pembagian bobot ini meningkatkan generalisasi dan menurunkan *overfitting*. Dengan menggunakan *error-backpropagation*, seluruh jaringan dilatih untuk memberikan *output* yang cukup mendekati hasil yang diinginkan dengan melakukan *backpropagasi* gradien yang

telah dihitung. Berdasarkan kinerjanya yang luar biasa, CNN digunakan secara luas di berbagai bidang, termasuk klasifikasi gambar, deteksi objek, dan pengenalan suara [25].



Gambar 2. 7 Cara Kerja dari CNN [22]

Convolutional Neural Network memiliki empat *layer* yaitu *convolution layer*, *pooling layer*, *activation function*, dan *fully connected layer*. Keempat layer tersebut memiliki kegunaan berbeda-beda, penjelasan mengenai keempat layer tersebut sebagai berikut:

1. *Convolution Layer*

Lapisan ini adalah tempat sebagian besar komputasi dalam CNN terjadi, maka dari itu lapisan ini merupakan lapisan pertama setelah *input* yang dilewati gambar. Lapisan ini terdiri dari *neuron* yang tersusun sedemikian rupa sehingga membentuk sebuah *filter* dengan panjang dan tinggi (*pixel*). Lapisan konvolusi akan melalui berbagai iterasi yang berbeda dan menghasilkan beberapa peta fitur. Proses pencocokan fitur dengan fitur yang diberikan *patch* gambar di setiap posisi yang memungkinkan dikenal sebagai konvolusi gambar.

2. *Pooling Layer*

Lapisan *pooling* terdiri dari sebuah *filter* dengan ukuran dan jarak tertentu yang bergerak melintasi seluruh area peta fitur. *Max Pooling* dan *Average Pooling* adalah jenis *pooling* yang sering digunakan. Penggunaan lapisan *pooling* bertujuan untuk mengurangi dimensi peta fitur (*downsampling*), sehingga mempercepat komputasi karena jumlah parameter yang perlu di *update* menjadi lebih sedikit dan membantu mengatasi masalah *overfitting*.

Lapisan *pooling* bekerja pada setiap tumpukan peta fitur dan mengurangi ukurannya. Salah satu bentuk lapisan *pooling* yang paling umum adalah menggunakan *filter* berukuran 2x2 yang diterapkan dengan langkah sebanyak 2,

sehingga beroperasi pada setiap potongan *input*. Dalam bentuk ini, ukuran peta fitur dapat berkurang hingga 75% dari ukuran aslinya. Sebagai contoh *Max Pooling*, lapisan *pooling* akan bekerja pada setiap bagian kedalaman dari volume *input* secara bergantian.

3. *Activation Function*

Proses aktivasi terjadi sebelum dan setelah lapisan *pooling* dan sesudah lapisan konvolusi pada *convolutional neural network*. Pada langkah ini, hasil dari operasi konvolusi mengalami transformasi dengan diterapkannya fungsi aktivasi. Terdapat beberapa jenis fungsi aktivasi yang umum digunakan dalam jaringan konvolusi, salah satunya adalah tangen hiperbolik atau ReLU. ReLU menjadi pilihan yang banyak disukai oleh beberapa peneliti karena memiliki sifat yang efektif dalam penggunaannya.

Dalam penggunaan fungsi ReLU sebagai aktivasi, *output neuron* akan menjadi nol jika *input*nya negatif. Namun, jika *input* fungsi aktivasi positif, *output neuron* akan sama dengan nilai *input* dari fungsi aktivasi tersebut.

4. *Fully Connected Layer*

Lapisan *Fully Connected* merujuk pada lapisan di mana semua aktivitas *neuron* dari lapisan sebelumnya tersambung dengan semua *neuron* di lapisan berikutnya, seperti dalam jaringan saraf tiruan. Setiap aktivitas yang berasal dari lapisan sebelumnya perlu diubah menjadi bentuk data satu dimensi sebelum dapat dihubungkan dengan seluruh *neuron* di lapisan *Fully Connected*.

Lapisan *Fully Connected* kerap digunakan dalam metode *Multi-layer Perceptron* dengan tujuan untuk mengolah data agar dapat diklasifikasikan. Perbedaan utama antara lapisan *Fully Connected* dan lapisan konvolusi adalah bahwa *neuron* dalam lapisan konvolusi hanya terhubung dengan area tertentu pada *input*. Di sisi lain, lapisan *Fully Connected* memiliki *neuron* yang terhubung secara menyeluruh. Meskipun begitu, keduanya masih melibatkan operasi perkalian titik (*dot product*), sehingga fungsinya tidak berbeda secara signifikan.

2.2.5 PARAMETER PENGUKURAN

2.2.5.1 *MEAN SQUARE ERROR* (MSE)

Pengukuran metrik kualitas gambar yang paling umum adalah *Mean Square Error* (MSE). MSE merupakan nilai *error* kuadrat rata-rata antara citra asli dengan

citra manipulasi. Nilai pengukuran MSE yang paling baik adalah yang mendekati nol. MSE juga dapat dikatakan sebagai *Mean Square Deviation* (MSD) dari suatu estimator. Estimator disebut sebagai prosedur untuk mengukur kuantitas gambar yang tidak teramati. MSE atau MSD mengukur rata-rata kuadrat kesalahan. Kesalahan yang dimaksud adalah perbedaan antara estimator dan hasil estimasi. [26]. MSE didefinisikan sebagai berikut:

$$MSE = \frac{1}{MN}\sum_{n=0}^{M}\sum_{m=1}^{N}[\hat{g}(n,m) - g(n,m)]^2 \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots(1)$$

Keterangan:

M = Dimensi citra asli

N = Dimensi citra rekonstruksi

Mean Squared Error (MSE) memiliki beberapa keuntungan sebagai ukuran ketepatan sinyal dalam pemrosesan sinyal:

1. Kemudahan dalam Representasi Matematis: MSE adalah metrik yang langsung dan mudah dipahami, yang dapat dihitung dengan menggunakan operasi matematika dasar.
2. Sifat Konveks dan Kemampuan Diferensiasi: MSE memiliki sifat-sifat yang diinginkan seperti konveksitas, simetri, dan dapat dihitung turunan, sehingga cocok digunakan dalam masalah optimisasi.
3. Konservasi Energi: MSE merupakan ukuran energi yang tetap konsisten setelah transformasi linier ortogonal atau transformasi kesatuan, misalnya seperti transformasi Fourier.
4. Penambahan Terhadap Distorsi Independen: MSE bersifat aditif terhadap sumber distorsi yang independen, memungkinkan analisis dan optimisasi untuk beberapa distorsi secara bersamaan.
5. Penggunaan Konvensi yang Umum: MSE telah secara luas digunakan sebagai metrik akurasi sinyal dalam berbagai aplikasi pengolahan sinyal. Hal ini menjadikannya sebagai standar yang diterima dan banyak digunakan untuk membandingkan serta mengevaluasi beragam algoritma [27].

2.2.5.2 *PEAK SIGNAL-TO-NOISE RATIO* (PSNR)

Peak signal-to-*noise* adalah penilaian kualitas yang paling umum digunakan untuk mengukur kualitas rekonstruksi kompresi gambar *lossy codec*. Sinyal dianggap sebagai data asli dan derau adalah kesalahan dihasilkan oleh kompresi

atau distorsi. PSNR adalah perkiraan perkiraan persepsi manusia terhadap kualitas rekonstruksi dibandingkan dengan codec kompresi. Perbandingan antara gambar yang direkonstruksi dan gambar asli diperlukan untuk pengembangan dan pelaksanaan rekonstruksi gambar. *Peak Signal to Noise Ratio* (PSNR) adalah metrik yang sering digunakan untuk tujuan ini [28].

Dalam penurunan kualitas kompresi gambar dan video, nilai PSNR bervariasi dari 30 hingga 50 dB untuk representasi data 8-bit dan dari 60 hingga 80 dB untuk 16-bit data. Nilai PSNR yang lebih tinggi dari 30 dB menunjukkan bahwa gambar yang direkonstruksi dan gambar asli lebih mirip dibanding nilai PSNR dibawah 30 dB. Dalam transmisi nirkabel, kisaran penurunan kualitas yang dapat diterima adalah sekitar 20 - 25 dB Satuan yang digunakan dalam PSNR adalah Desibel (dB). Kualitas gambar sampul sebelum dan sesudah pesan dimasukkan dan dibandingkan menggunakan PSNR. Nilai PSNR yang tinggi penting dalam meminimalkan degradasi yang disebabkan oleh *noise* selama transmisi video karena hal ini mengindikasikan tingkat kualitas sinyal yang lebih tinggi dan distorsi yang lebih sedikit. PSNR mengukur rasio antara daya maksimum yang mungkin dari suatu sinyal dan daya dari *noise* yang mempengaruhi ketepatan sinyal. Nilai PSNR yang lebih tinggi menunjukkan jumlah *noise* yang lebih kecil dan ketepatan sinyal video yang lebih tinggi, yang berarti bahwa video akan tampak lebih jernih dan lebih menyenangkan secara visual bagi pemirsa. Dimana nilai dari MSE (*Mean Square Error*) harus ditetapkan sebelum PSNR dapat dihitung [26]. PSNR didefinisikan dalam aritmatika sebagai berikut:

$$PSNR = 10 \log_{10} \left(\frac{C^2_{max}}{MSE} \right) \text{.....(2)}$$

Keterangan:

$C^2_{\max}$ = Nilai pixel tertinggi dari keseluruhan dimensi

MSE = Nilai eror kuadrat rata-rata (*Mean Square Error*)

Peak Signal Noise Ratio (PSNR) memiliki beberapa keuntungan sebagai parameter kualitas kompresi:

1. Kemudahan dan Kalkulasi Sederhana: PSNR dapat dengan mudah dihitung menggunakan formula sederhana yang membandingkan sinyal asli dengan yang telah terkompresi. Ini menjadikan PSNR sebagai metrik yang praktis dan efisien untuk menilai mutu video.

2. Penggunaan yang Luas: PSNR telah secara luas digunakan dalam dunia penelitian dan industri sebagai alat pengukuran kualitas video. Dengan begitu, metrik ini menjadi umum dan dapat dibandingkan dengan hasil-hasil penelitian sebelumnya.
3. Reaksi Terhadap Perubahan Kecil: PSNR memiliki sensitivitas yang tinggi terhadap perubahan kecil dalam sinyal video. Ini mengizinkan identifikasi perubahan-perubahan kecil dalam mutu video yang mungkin tidak terdeteksi oleh mata manusia.
4. Korelasi dengan Persepsi Manusia: Walaupun PSNR tidak sepenuhnya merefleksikan persepsi manusia terhadap kualitas video, terdapat hubungan antara PSNR yang tinggi dan kualitas video yang lebih baik menurut pandangan manusia. Skor PSNR yang tinggi menunjukkan tingkat distorsi yang rendah dan mutu video yang lebih baik [28].