INSTRUCTOR TRANSCRIPT · FALL 2026/27

How Machines Learn to Make Moving Worlds

Neural networks, CNNs, transformers, diffusion, Seedance, MiniMax H3, and an AI filmmaking workflow

102 slides · approximately one minute per slide

AMCC 5160 · AI-Driven Animation and Video Generation

Lecture map

Teaching intent. Move from first principles to current production practice. Each section gives students a mental model, a technical mechanism, a diagnostic question, and a concrete filmmaking consequence.

Act	Focus	Range
Act I	Orientation and course contract	Slides 1–8
Act II	Neural networks and CNNs	Slides 9–22
Act III	Attention and transformers	Slides 23–31
Act IV	Latent generative models	Slides 32–41
Act V	Video-generation systems and landscape	Slides 42–59
Act VI	Seedance 1.0–2.5	Slides 60–70
Act VII	MiniMax H3	Slides 71–83
Act VIII	A step-by-step AI filmmaking workflow	Slides 84–99
Act IX	Quiz review and in-class challenge	Slides 100–102

Act I · Orientation and course contract

SLIDE 001 · How machines learn to make moving worlds

AMCC 5160 · LECTURE 01

87 words · ≈1 minute

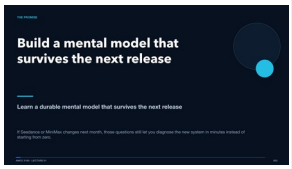
Welcome. Today I want to remove some of the magic without removing the wonder. We will begin with a single artificial neuron, build up to convolution and transformers, then open current video systems and finally assemble a film workflow you can run and critique.

By the end, a model demo should no longer look like a black box. You should be able to point to the representation, conditioning, sampling choice, and production decision behind it.

Let us start with the contract for how this class will work.

Sources: https://hyang.org/amcc5160/

SLIDE 002 · Build a mental model that survives the next release



THE PROMISE

73 words · ≈1 minute

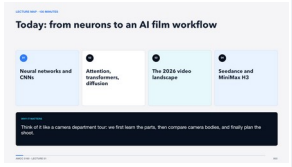
Model names will change faster than our semester. The useful skill is learning how to ask stable questions: what is represented, what is predicted, what conditions the prediction, what remains consistent over time, and where human judgment enters.

If Seedance or MiniMax changes next month, those questions still let you diagnose the new system in minutes instead of starting from zero.

Before the technical journey, here is exactly how the course is assessed.

Sources: https://hyang.org/amcc5160/

SLIDE 003 · Today: from neurons to an AI film workflow



LECTURE MAP · 100 MINUTES

65 words · ≈1 minute

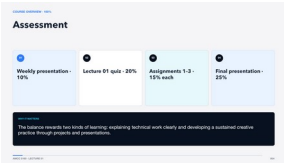
The lecture is deliberately paced as one idea per slide. The first half builds the machinery. The second half uses that machinery to read current systems and then convert model capability into a production pipeline.

Think of it like a camera department tour: we first learn the parts, then compare camera bodies, and finally plan the shoot.

A short assessment slide now prevents confusion later.

Sources: https://cs231n.stanford.edu/slides/2025/lecture_14.pdf · https://diffusion.csail.mit.edu/2025/ · https://cme296.stanford.edu/syllabus/

SLIDE 004 · Assessment



COURSE OVERVIEW · 100%

76 words · ≈1 minute

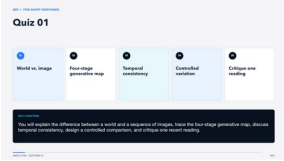
The course assessment has four parts. Weekly presentations account for ten percent. The Lecture 01 quiz is twenty percent. The three assignments are fifteen percent each, for forty-five percent in total. The final presentation is twenty-five percent. Together, these components add to one hundred percent.

The balance rewards two kinds of learning: explaining technical work clearly and developing a sustained creative practice through projects and presentations.

The quiz focuses on the concepts that organize this lecture.

Sources: https://hyang.org/amcc5160/ · https://hyang.org/amcc5160/quiz/

SLIDE 005 · Quiz 01



SEP 1 · FIVE SHORT RESPONSES

78 words · ≈1 minute

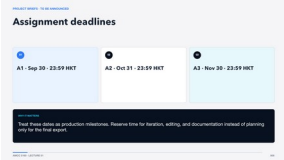
Quiz 01 contains five short responses worth four points each. A strong answer defines the idea, explains why it matters, and gives one specific example. The questions are designed to test understanding, not vocabulary memorization.

You will explain the difference between a world and a sequence of images, trace the four-stage generative map, discuss temporal consistency, design a controlled comparison, and critique one recent reading.

The assignment briefs will be announced separately, while the deadlines are already fixed.

Sources: https://hyang.org/amcc5160/quiz/

SLIDE 006 · Assignment deadlines



PROJECT BRIEFS · TO BE ANNOUNCED

72 words · ≈1 minute

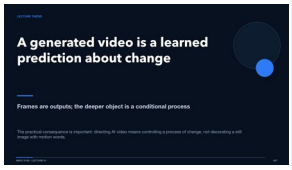
The three project briefs will be announced separately. Their deadlines are fixed: Assignment 1 is due September 30, Assignment 2 is due October 31, and Assignment 3 is due November 30. Each deadline is 23:59 Hong Kong time.

Treat these dates as production milestones. Reserve time for iteration, editing, and documentation instead of planning only for the final export.

With the course structure clear, we can begin with the central technical idea.

Sources: https://hyang.org/amcc5160/

SLIDE 007 · A generated video is a learned prediction about change



LECTURE THESIS

69 words · ≈1 minute

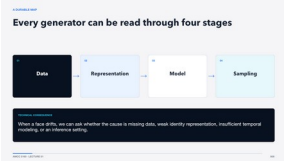
A video model does not store a shelf of finished films and retrieve one. It learns statistical regularities in images, motion, language, and sound, then produces one trajectory that is compatible with the conditions we provide.

The practical consequence is important: directing AI video means controlling a process of change, not decorating a still image with motion words.

To understand that process, we need to start with learning itself.

Sources: https://arxiv.org/abs/2604.06339

SLIDE 008 · Every generator can be read through four stages



A DURABLE MAP

70 words · ≈1 minute

Data supplies examples. Representation decides what information becomes compact and comparable. The model learns relationships in that representation. Sampling turns a learned distribution into one particular output. This four-stage map will reappear when we inspect Seedance, MiniMax, and ComfyUI.

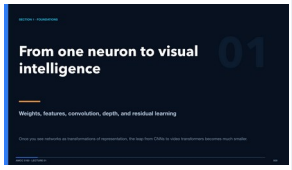
When a face drifts, we can ask whether the cause is missing data, weak identity representation, insufficient temporal modeling, or an inference setting.

First: what does a neural network actually learn?

Sources: https://arxiv.org/abs/2112.10752 · https://arxiv.org/abs/2212.09748

Act II · Neural networks and CNNs

SLIDE 009 · From one neuron to visual intelligence



SECTION 1 · FOUNDATIONS

67 words · ≈1 minute

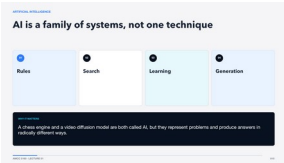
This section is intentionally basic, but not simplistic. We will build enough detail that later architecture diagrams become readable. You do not need to calculate gradients by hand; you do need to understand what information moves through the network and what training changes.

Once you see networks as transformations of representation, the leap from CNNs to video transformers becomes much smaller.

Begin with the smallest trainable unit.

Sources: https://arxiv.org/abs/1207.0580 · https://arxiv.org/abs/1512.03385

SLIDE 010 · AI is a family of systems, not one technique



ARTIFICIAL INTELLIGENCE

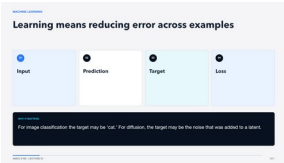
65 words · ≈1 minute

Artificial intelligence includes hand-written rules, search procedures, learned predictors, and generative systems. Modern video models sit mainly in the last two categories: they learn from examples and then synthesize new samples.

A chess engine and a video diffusion model are both called AI, but they represent problems and produce answers in radically different ways.

Machine learning narrows the question to how behavior changes from data.

SLIDE 011 · Learning means reducing error across examples



MACHINE LEARNING

73 words · ≈1 minute

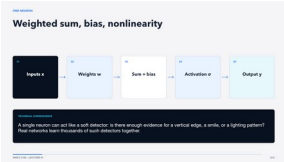
A supervised training step is simple to state. Give the network an input, produce a prediction, compare it with a target using a loss, and adjust parameters so future predictions improve. Generative training uses more elaborate targets, but the optimization loop remains recognizable.

For image classification the target may be ‘cat.’ For diffusion, the target may be the noise that was added to a latent.

The adjustable parameters live inside neurons and layers.

Sources: https://arxiv.org/abs/2006.11239

SLIDE 012 · Weighted sum, bias, nonlinearity



ONE NEURON

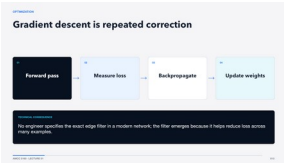
80 words · ≈1 minute

A neuron multiplies each input by a weight, adds the results and a bias, then passes the value through a nonlinear activation. The weights express which input directions matter. The nonlinearity lets stacked layers represent more than one large linear transformation.

A single neuron can act like a soft detector: is there enough evidence for a vertical edge, a smile, or a lighting pattern? Real networks learn thousands of such detectors together.

Training is the process that tunes those weights.

SLIDE 013 · Gradient descent is repeated correction



OPTIMIZATION

72 words · ≈1 minute

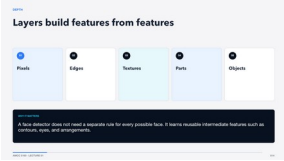
During the forward pass, data moves through the network. The loss turns performance into a scalar. Backpropagation uses the chain rule to estimate how each parameter contributed to that loss. An optimizer then nudges parameters in a direction expected to reduce future error.

No engineer specifies the exact edge filter in a modern network; the filter emerges because it helps reduce loss across many examples.

Stacking neurons creates representations of increasing abstraction.

Sources: https://arxiv.org/abs/1207.0580

SLIDE 014 · Layers build features from features



DEPTH

72 words · ≈1 minute

The familiar story is that early visual layers respond to simple edges, middle layers combine them into textures or parts, and deeper layers respond to larger semantic patterns. The hierarchy is not perfectly clean, but it is a useful mental model.

A face detector does not need a separate rule for every possible face. It learns reusable intermediate features such as contours, eyes, and arrangements.

Convolution made this hierarchy practical for images.

Sources: https://arxiv.org/abs/1207.0580

SLIDE 015 · Nearby pixels usually belong together

WHY IMAGES ARE SPECIAL

57 words · ≈1 minute

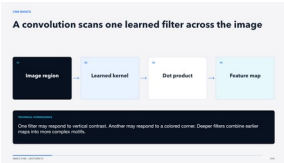
An image is not an arbitrary list of numbers. Neighbouring pixels are correlated, patterns repeat across locations, and the same object can appear anywhere. A good architecture should exploit those regularities instead of relearning them at every position.

Convolution encodes that prior directly: look locally with the same filter everywhere.

Let us make the convolution operation tangible.

Sources: https://arxiv.org/abs/1207.0580

SLIDE 016 · A convolution scans one learned filter across the image



CNN BASICS

70 words · ≈1 minute

A small kernel slides across the image. At each location it computes a weighted sum of nearby pixels. The result is a feature map showing where that pattern appears. During training, the kernel values are learned rather than designed by hand.

One filter may respond to vertical contrast. Another may respond to a colored corner. Deeper filters combine earlier maps into more complex motifs.

The key advantage is weight sharing.

Sources: https://arxiv.org/abs/1207.0580

SLIDE 017 · Four numbers control the spatial calculation

CONVOLUTION

58 words · ≈1 minute

Kernel size controls the local window. Stride controls how far the filter moves. Padding controls what happens near borders. Channels let the filter combine information across RGB or intermediate feature maps.

A three-by-three kernel with stride one preserves fine local detail; larger strides reduce resolution and computation but may discard small structures.

These choices determine the receptive field.

Sources: https://arxiv.org/abs/1207.0580

SLIDE 018 · A channel is a learned question asked everywhere

FEATURE MAPS

71 words · ≈1 minute

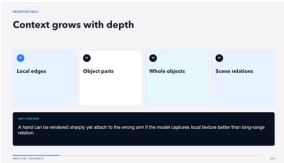
After convolution, each output channel is a spatial map of activations. It is useful to imagine a channel as a learned question. The network asks that question at every location and records the strength of the answer.

Later layers do not see raw pixels directly. They see stacks of these learned answers, which is why representation matters so much.

As layers stack, each activation sees a larger region of the image.

Sources: https://arxiv.org/abs/1207.0580

SLIDE 019 · Context grows with depth



RECEPTIVE FIELD

64 words · ≈1 minute

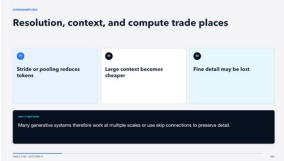
The receptive field is the portion of the input that can influence one activation. Stacking convolutions grows it. Early units see small patches; later units can integrate information across an object or scene.

A hand can be rendered sharply yet attach to the wrong arm if the model captures local texture better than long-range relation.

Downsampling expands context, but it also creates a trade-off.

Sources: https://arxiv.org/abs/1207.0580

SLIDE 020 · Resolution, context, and compute trade places



DOWNSAMPLING

57 words · ≈1 minute

Pooling or strided convolution reduces spatial size. This makes deeper processing cheaper and increases effective context. But information thrown away early is difficult to reconstruct later, especially text, fingers, and thin geometry.

Many generative systems therefore work at multiple scales or use skip connections to preserve detail.

Residual connections solve a different problem: training very deep networks.

Sources: https://arxiv.org/abs/1512.03385

SLIDE 021 · Learn a correction, not an entirely new representation

RESIDUAL LEARNING

64 words · ≈1 minute

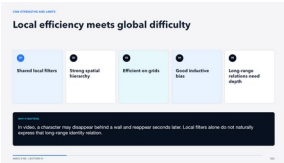
A residual block adds its learned transformation back to the original input. If the new transformation is temporarily unhelpful, the identity path still carries information forward. This makes very deep networks easier to optimize.

The same idea appears throughout modern architectures: preserve a stable path, then learn a useful update around it.

CNNs were transformative, but their local bias becomes awkward for global relationships.

Sources: https://arxiv.org/abs/1512.03385

SLIDE 022 · Local efficiency meets global difficulty



CNN STRENGTHS AND LIMITS

64 words · ≈1 minute

CNNs are excellent at exploiting locality and grids. Their weakness is not that they cannot model global structure, but that global interaction is indirect and often requires many layers or special modules.

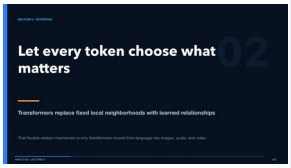
In video, a character may disappear behind a wall and reappear seconds later. Local filters alone do not naturally express that long-range identity relation.

Sequence models approach the problem from another direction.

Sources: https://arxiv.org/abs/1207.0580 · https://arxiv.org/abs/1512.03385

Act III · Attention and transformers

SLIDE 023 · Let every token choose what matters



SECTION 2 · ATTENTION

50 words · ≈1 minute

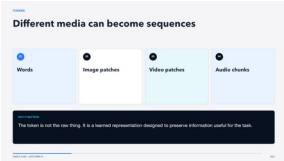
Attention begins with a provocative idea: instead of deciding in advance which neighbours matter, let each element compare itself with all relevant elements and construct a weighted mixture.

That flexible relation mechanism is why transformers moved from language into images, audio, and video.

We need one more abstraction first: tokens.

Sources: https://arxiv.org/abs/1706.03762

SLIDE 024 · Different media can become sequences



TOKENS

76 words · ≈1 minute

A token is a unit the model processes. In language it may be a word fragment. In vision it may be a patch embedding. In video it may represent a compressed block across space and time. Once converted to vectors, very different media can share transformer machinery.

The token is not the raw thing. It is a learned representation designed to preserve information useful for the task.

Embeddings place those tokens in a continuous vector space.

Sources: https://arxiv.org/abs/1706.03762 · https://arxiv.org/abs/2010.11929

SLIDE 025 · Meaning becomes geometry

EMBEDDINGS

63 words · ≈1 minute

An embedding maps a discrete or structured input into a vector. The coordinates do not have simple names, but distances and directions can capture useful relationships learned from data.

A text prompt, a reference image, and an audio cue may enter through different encoders, yet all must eventually become representations the generator can relate.

Attention performs that relation through queries, keys, and values.

Sources: https://arxiv.org/abs/1706.03762

SLIDE 026 · Queries ask; keys match; values contribute

SELF-ATTENTION

69 words · ≈1 minute

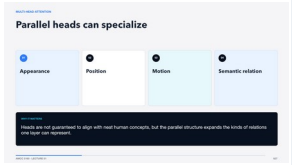
Each token produces a query, a key, and a value. Query-key similarity determines attention weights. The weighted sum of values becomes the updated token. The square-root scaling prevents dot products from becoming so large that softmax saturates.

For a red scarf across frames, the current token can attend to earlier tokens that carry the scarf's identity and appearance.

Multiple heads let the model learn several relationship types at once.

Sources: https://arxiv.org/abs/1706.03762

SLIDE 027 · Parallel heads can specialize



MULTI-HEAD ATTENTION

65 words · ≈1 minute

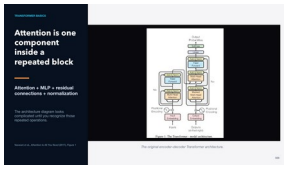
Instead of one attention map, multi-head attention projects tokens into several subspaces. One head may emphasize local appearance, another long-distance identity, another temporal order. The outputs are concatenated and mixed.

Heads are not guaranteed to align with neat human concepts, but the parallel structure expands the kinds of relations one layer can represent.

The transformer block wraps attention with normalization, an MLP, and residual paths.

Sources: https://arxiv.org/abs/1706.03762

SLIDE 028 · Attention is one component inside a repeated block



TRANSFORMER BASICS

61 words · ≈1 minute

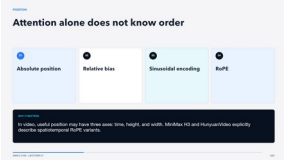
The original transformer used an encoder and decoder, each built from repeated blocks. Modern vision and diffusion systems modify this structure, but the recurring pattern remains: mix information across tokens, transform each token with an MLP, and preserve stable residual paths.

The architecture diagram looks complicated until you recognize those repeated operations.

A sequence still needs information about order and position.

Sources: https://arxiv.org/abs/1706.03762

SLIDE 029 · Attention alone does not know order



POSITION

69 words · ≈1 minute

Because attention compares a set of vectors, it needs extra structure to distinguish left from right or earlier from later. Position encodings inject coordinates. Rotary position embedding, or RoPE, rotates query and key components so relative position influences similarity.

In video, useful position may have three axes: time, height, and width. MiniMax H3 and HunyuanVideo explicitly describe spatiotemporal RoPE variants.

Vision Transformers apply the sequence idea to image patches.

Sources: https://arxiv.org/abs/1706.03762 · https://arxiv.org/abs/2010.11929 · https://www.minimax.io/news/minimax-h3-open-source · https://arxiv.org/abs/2511.18870

SLIDE 030 · An image becomes a sequence of patch tokens

VISION TRANSFORMER

65 words · ≈1 minute

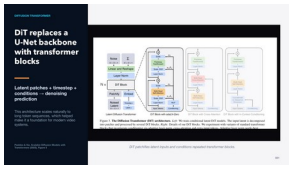
A Vision Transformer cuts an image into patches, flattens or projects each patch into a token, adds position information, and processes the sequence with transformer blocks. It replaces much of the CNN's local bias with data-driven global interaction.

The price is token count: smaller patches preserve detail but create longer, more expensive sequences.

Diffusion Transformers use related blocks to predict how noisy latents should change.

Sources: https://arxiv.org/abs/2010.11929

SLIDE 031 · DiT replaces a U-Net backbone with transformer blocks



DIFFUSION TRANSFORMER

66 words · ≈1 minute

DiT patchifies a noisy latent, adds timestep and conditioning information, processes the tokens through transformer blocks, and reconstructs a prediction in latent space. The paper showed that increasing transformer compute correlated with better image generation quality.

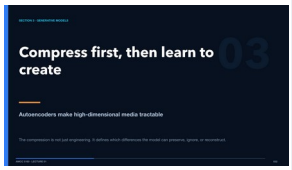
This architecture scales naturally to long token sequences, which helped make it a foundation for modern video systems.

Before diffusion, we need to understand why models generate in latent space.

Sources: https://arxiv.org/abs/2212.09748

Act IV · Latent generative models

SLIDE 032 · Compress first, then learn to create



SECTION 3 · GENERATIVE MODELS

49 words · ≈1 minute

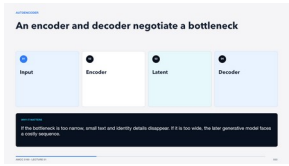
Raw video contains an enormous number of values. A practical generator usually compresses frames into latents, performs expensive modeling there, and decodes only at the end.

The compression is not just engineering. It defines which differences the model can preserve, ignore, or reconstruct.

An autoencoder provides the basic pattern.

Sources: https://arxiv.org/abs/1312.6114 · https://arxiv.org/abs/2112.10752

SLIDE 033 · An encoder and decoder negotiate a bottleneck



AUTOENCODER

65 words · ≈1 minute

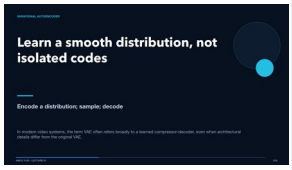
The encoder maps an input to a smaller latent representation. The decoder reconstructs the input from that bottleneck. Training rewards faithful reconstruction, so the latent must preserve information the decoder needs.

If the bottleneck is too narrow, small text and identity details disappear. If it is too wide, the later generative model faces a costly sequence.

A variational autoencoder adds structure to the latent distribution.

Sources: https://arxiv.org/abs/1312.6114

SLIDE 034 · Learn a smooth distribution, not isolated codes



VARIATIONAL AUTOENCODER

68 words · ≈1 minute

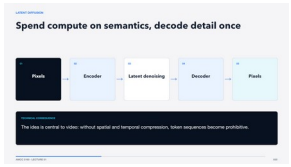
A VAE predicts parameters of a latent distribution rather than a single deterministic code. A regularization term encourages nearby, sampleable structure. This makes the latent space more suitable for generation, though reconstruction can become softer.

In modern video systems, the term VAE often refers broadly to a learned compressor-decoder, even when architectural details differ from the original VAE.

Latent diffusion moves the denoising process into this compressed space.

Sources: https://arxiv.org/abs/1312.6114 · https://arxiv.org/abs/2112.10752

SLIDE 035 · Spend compute on semantics, decode detail once



LATENT DIFFUSION

54 words · ≈1 minute

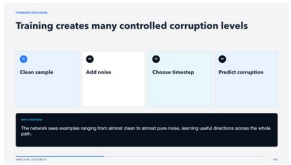
Latent diffusion reduces the spatial size before running the generative model. This lowers memory and compute while retaining enough perceptual information for high-quality decoding. Text or other conditions guide the denoising path.

The idea is central to video: without spatial and temporal compression, token sequences become prohibitive.

Diffusion itself begins by deliberately corrupting data.

Sources: https://arxiv.org/abs/2112.10752

SLIDE 036 · Training creates many controlled corruption levels



FORWARD DIFFUSION

64 words · ≈1 minute

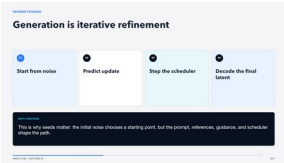
The forward process gradually adds Gaussian noise according to a schedule. During training, we can jump directly to a chosen noise level, ask the network to predict the added noise or a related target, and compute an error.

The network sees examples ranging from almost clean to almost pure noise, learning useful directions across the whole path.

Generation runs the learned process in reverse.

Sources: https://arxiv.org/abs/2006.11239

SLIDE 037 · Generation is iterative refinement



REVERSE PROCESS

63 words · ≈1 minute

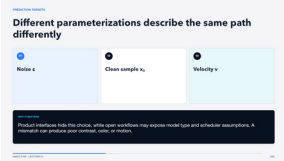
At inference, a sampler starts with random noise and repeatedly applies the model's prediction according to a numerical schedule. Each step moves the latent toward a sample compatible with training and conditioning.

This is why seeds matter: the initial noise chooses a starting point, but the prompt, references, guidance, and scheduler shape the path.

The training target can be parameterized in several ways.

Sources: https://arxiv.org/abs/2006.11239 · https://arxiv.org/abs/2210.02747

SLIDE 038 · Different parameterizations describe the same path differently



PREDICTION TARGETS

63 words · ≈1 minute

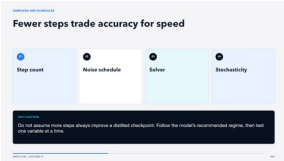
A diffusion model may predict noise, the clean sample, or a velocity-like combination. These parameterizations affect optimization and numerical behavior, even when they describe related generative trajectories.

Product interfaces hide this choice, while open workflows may expose model type and scheduler assumptions. A mismatch can produce poor contrast, color, or motion.

Sampling quality also depends on how many steps we take and where.

Sources: https://arxiv.org/abs/2006.11239 · https://arxiv.org/abs/2210.02747

SLIDE 039 · Fewer steps trade accuracy for speed



SAMPLERS AND SCHEDULES

66 words · ≈1 minute

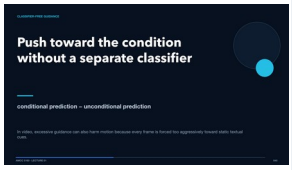
A sampler is a numerical method for following the reverse process. The schedule allocates effort across noise levels. Distilled models may work in very few steps because training has compressed a longer trajectory into a shorter one.

Do not assume more steps always improve a distilled checkpoint. Follow the model's recommended regime, then test one variable at a time.

Conditioning steers which solution the trajectory reaches.

Sources: https://arxiv.org/abs/2006.11239 · https://www.minimax.io/news/minimax-h3-open-source

SLIDE 040 · Push toward the condition without a separate classifier



CLASSIFIER-FREE GUIDANCE

57 words · ≈1 minute

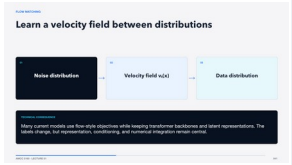
Classifier-free guidance compares a conditional and an unconditional prediction, then amplifies the difference. Stronger guidance can improve prompt adherence but may reduce diversity or create oversaturated, brittle results.

In video, excessive guidance can also harm motion because every frame is forced too aggressively toward static textual cues.

Flow matching offers another view of the same generative journey.

Sources: https://arxiv.org/abs/2212.09748 · https://arxiv.org/abs/2210.02747

SLIDE 041 · Learn a velocity field between distributions



FLOW MATCHING

65 words · ≈1 minute

Flow matching trains a vector field that transports samples from a simple distribution toward the data distribution. At inference, an ODE solver follows that learned field. Rectified-flow variants aim for straighter, easier trajectories.

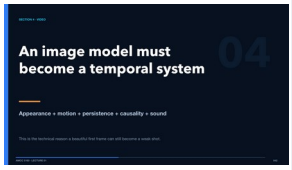
Many current models use flow-style objectives while keeping transformer backbones and latent representations. The labels change, but representation, conditioning, and numerical integration remain central.

Now extend those ideas from images into time.

Sources: https://arxiv.org/abs/2210.02747

Act V · Video-generation systems and landscape

SLIDE 042 · An image model must become a temporal system



SECTION 4 · VIDEO

61 words · ≈1 minute

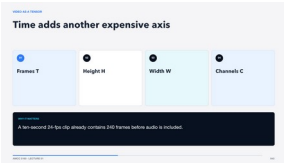
Video is not an image with more frames. The model must preserve identity, geometry, lighting, and world state while also producing meaningful change. It must decide what moves, when it moves, and how one event affects the next.

This is the technical reason a beautiful first frame can still become a weak shot.

Start by counting what a raw video contains.

Sources: https://arxiv.org/abs/2410.13720 · https://arxiv.org/abs/2503.20314

SLIDE 043 · Time adds another expensive axis



VIDEO AS A TENSOR

58 words · ≈1 minute

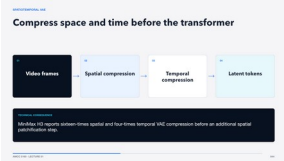
A video can be represented as a T-by-H-by-W-by-C tensor. Doubling duration doubles values. Doubling both spatial dimensions quadruples them. Attention over all raw locations would grow even faster because pairwise interaction is quadratic in token count.

A ten-second 24-fps clip already contains 240 frames before audio is included.

Compression is therefore a design requirement, not an optional optimization.

Sources: https://cs231n.stanford.edu/slides/2025/lecture_14.pdf · https://arxiv.org/abs/2410.13720

SLIDE 044 · Compress space and time before the transformer



SPATIOTEMPORAL VAE

56 words · ≈1 minute

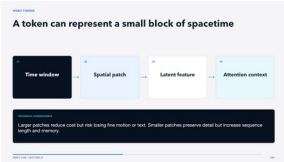
A video VAE compresses height and width and often reduces the time axis as well. Causal designs ensure a latent does not depend on future frames in ways that break streaming or extension.

MiniMax H3 reports sixteen-times spatial and four-times temporal VAE compression before an additional spatial patchification step.

The resulting latent blocks become video tokens.

Sources: https://www.minimax.io/news/minimax-h3-open-source · https://arxiv.org/abs/2511.18870

SLIDE 045 · A token can represent a small block of spacetime



VIDEO TOKENS

52 words · ≈1 minute

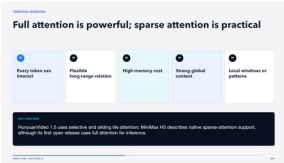
Patchification groups latent values into tokens. Each token carries content plus coordinates. The transformer then models relationships across space, time, and conditions.

Larger patches reduce cost but risk losing fine motion or text. Smaller patches preserve detail but increase sequence length and memory.

Temporal attention determines how broadly tokens communicate across frames.

Sources: https://arxiv.org/abs/2212.09748 · https://www.minimax.io/news/minimax-h3-open-source

SLIDE 046 · Full attention is powerful; sparse attention is practical



TEMPORAL MODELING

66 words · ≈1 minute

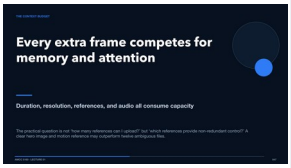
Full spatiotemporal attention provides maximum flexibility but becomes expensive as duration and resolution grow. Sparse, sliding, factorized, or block attention limits which tokens interact, trading some flexibility for practical scale.

HunyuanVideo 1.5 uses selective and sliding tile attention; MiniMax H3 describes native sparse-attention support, although its first open release uses full attention for inference.

Longer context is useful only if the representation preserves identity and state.

Sources: https://arxiv.org/abs/2511.18870 · https://www.minimax.io/news/minimax-h3-open-source

SLIDE 047 · Every extra frame competes for memory and attention



THE CONTEXT BUDGET

60 words · ≈1 minute

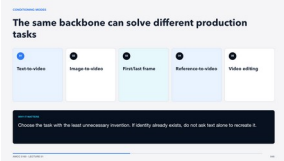
A multimodal generator has a finite context and compute budget. More reference files, higher resolution, longer duration, and native audio increase the sequence or conditioning burden.

The practical question is not ‘how many references can I upload?’ but ‘which references provide non-redundant control?’ A clear hero image and motion reference may outperform twelve ambiguous files.

That leads to multimodal conditioning.

Sources: https://arxiv.org/abs/2604.14148 · https://www.minimax.io/news/minimax-h3-open-source

SLIDE 048 · The same backbone can solve different production tasks



CONDITIONING MODES

64 words · ≈1 minute

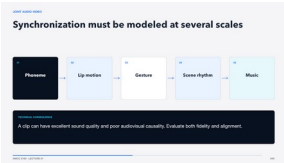
Text-to-video invents nearly everything. Image-to-video anchors appearance. First-and-last-frame workflows constrain endpoints. Reference-to-video can transfer identity, style, motion, or sound. Editing workflows preserve more of an existing clip while changing selected attributes.

Choose the task with the least unnecessary invention. If identity already exists, do not ask text alone to recreate it.

Audio is becoming a condition and an output rather than a separate afterthought.

Sources: https://arxiv.org/abs/2604.14148 · https://www.minimax.io/news/minimax-h3-open-source · https://docs.comfy.org/tutorials/video/wan/wan2_2

SLIDE 049 · Synchronization must be modeled at several scales



JOINT AUDIO-VIDEO

59 words · ≈1 minute

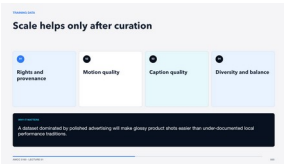
Native audio-video models predict related visual and audio latents. Synchronization is not one problem: lip motion may require frame-level alignment, footsteps need event timing, ambience needs scene consistency, and music operates across longer structure.

A clip can have excellent sound quality and poor audiovisual causality. Evaluate both fidelity and alignment.

Before training, the system needs usable data and descriptions.

Sources: https://arxiv.org/abs/2512.13507 · https://arxiv.org/abs/2604.14148 · https://www.minimax.io/news/minimax-h3-open-source

SLIDE 050 · Scale helps only after curation



TRAINING DATA

61 words · ≈1 minute

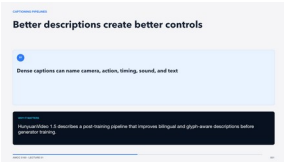
Video training data contains duplicates, cuts, overlays, watermarks, weak motion, missing audio, and uncertain rights. Curation filters and scores clips before training. The selection criteria silently define which kinds of worlds become easy for the model.

A dataset dominated by polished advertising will make glossy product shots easier than under-documented local performance traditions.

Captions turn raw clips into controllable training pairs.

Sources: https://arxiv.org/abs/2503.20314 · https://arxiv.org/abs/2511.18870 · https://arxiv.org/abs/2506.09113

SLIDE 051 · Better descriptions create better controls



CAPTIONING PIPELINES

55 words · ≈1 minute

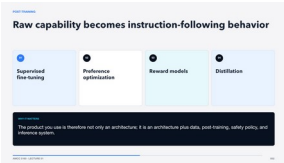
A weak caption says ‘a person outside.’ A production-grade caption can describe subject, action, shot size, camera movement, lighting, sound, and temporal order. Some systems train caption models and refiners specifically for video data.

HunyuanVideo 1.5 describes a post-training pipeline that improves bilingual and glyph-aware descriptions before generator training.

Pretraining creates capability; post-training shapes usefulness.

Sources: https://arxiv.org/abs/2511.18870 · https://arxiv.org/abs/2506.09113

SLIDE 052 · Raw capability becomes instruction-following behavior



POST-TRAINING

59 words · ≈1 minute

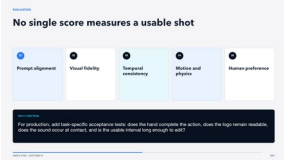
After large-scale pretraining, teams refine instruction following, aesthetic preference, motion quality, safety, and speed. Seedance 1.0 reports supervised fine-tuning and video RLHF with multidimensional rewards. Distillation compresses inference into fewer steps.

The product you use is therefore not only an architecture; it is an architecture plus data, post-training, safety policy, and inference system.

Evaluation tries to separate those effects.

Sources: https://arxiv.org/abs/2506.09113 · https://cme296.stanford.edu/syllabus/

SLIDE 053 · No single score measures a usable shot



EVALUATION

71 words · ≈1 minute

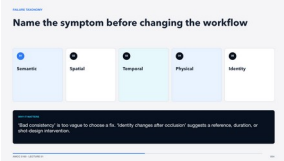
Benchmarks combine automated metrics, multimodal judges, and human ratings. Each has blind spots. A model may score well on average yet fail the exact identity, camera, or causal requirement of your scene.

For production, add task-specific acceptance tests: does the hand complete the action, does the logo remain readable, does the sound occur at contact, and is the usable interval long enough to edit?

A failure vocabulary makes those tests precise.

Sources: https://arxiv.org/abs/2604.06339 · https://arxiv.org/abs/2604.14148

SLIDE 054 · Name the symptom before changing the workflow



FAILURE TAXONOMY

73 words · ≈1 minute

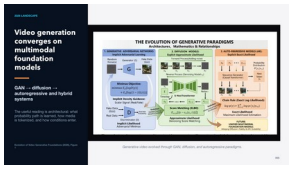
Semantic failures change what the scene means. Spatial failures break geometry and contact. Temporal failures flicker or drift. Physical failures violate motion or causality. Identity failures change a subject across time. Aesthetic failure can overlay all five when the result contradicts the intended language.

‘Bad consistency’ is too vague to choose a fix. ‘Identity changes after occlusion’ suggests a reference, duration, or shot-design intervention.

Now place these mechanisms in the current research landscape.

Sources: https://arxiv.org/abs/2604.06339

SLIDE 055 · Video generation converges on multimodal foundation models



2026 LANDSCAPE

63 words · ≈1 minute

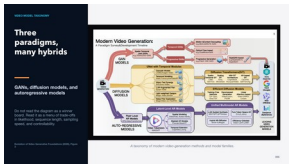
The 2026 survey frames the field as an evolution from adversarial learning to diffusion and autoregressive approaches, with current systems converging toward unified multimodal foundation models. The categories are not clean product boxes; many systems combine ideas.

The useful reading is architectural: what probability path is learned, how media is tokenized, and how conditions enter.

A taxonomy shows the major branches more concretely.

Sources: https://arxiv.org/abs/2604.06339

SLIDE 056 · Three paradigms, many hybrids



VIDEO MODEL TAXONOMY

65 words · ≈1 minute

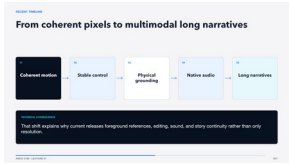
The survey maps modern video generation across GAN, diffusion, and autoregressive branches. The dominant frontier currently uses diffusion or flow with transformer backbones, while token-based autoregressive and hybrid systems continue to grow.

Do not read the diagram as a winner board. Read it as a menu of trade-offs in likelihood, sequence length, sampling speed, and controllability.

The timeline reveals how quickly the capability frontier moved.

Sources: https://arxiv.org/abs/2604.06339

SLIDE 057 · From coherent pixels to multimodal long narratives



RECENT TIMELINE

55 words · ≈1 minute

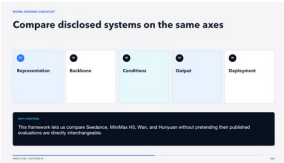
The survey's timeline shows the problem changing. Early systems competed on coherent motion. Later work emphasized stability and control, then physical grounding, multimodal integration, and longer narrative structure.

That shift explains why current releases foreground references, editing, sound, and story continuity rather than only resolution.

A compact comparison will help us choose models for production.

Sources: https://arxiv.org/abs/2604.06339

SLIDE 058 · Compare disclosed systems on the same axes



MODEL READING CHECKLIST

65 words · ≈1 minute

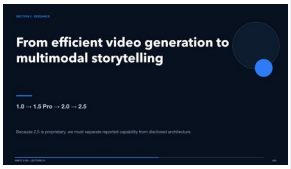
Marketing comparisons often mix resolution, duration, task, and human preference. A more technical comparison asks how media is compressed, which network predicts latents, what conditions are supported, what the output contains, and whether the system is open, hosted, or hybrid.

This framework lets us compare Seedance, MiniMax H3, Wan, and Hunyuan without pretending their published evaluations are directly interchangeable.

We begin with ByteDance's Seedance line.

Sources: https://arxiv.org/abs/2604.14148 · https://www.minimax.io/news/minimax-h3-open-source · https://arxiv.org/abs/2503.20314 · https://arxiv.org/abs/2511.18870

SLIDE 059 · From efficient video generation to multimodal storytelling



SECTION 5 · SEEDANCE

58 words · ≈1 minute

Seedance is best understood as a sequence of system-level changes. Version 1.0 established a scalable video foundation. Version 1.5 Pro added native audio-video generation. Version 2.0 unified multimodal inputs and editing. Version 2.5 extends duration and one-take control.

Because 2.5 is proprietary, we must separate reported capability from disclosed architecture.

Start with what the 1.0 report actually reveals.

Sources: https://arxiv.org/abs/2506.09113 · https://arxiv.org/abs/2512.13507 · https://arxiv.org/abs/2604.14148 · https://seed.bytedance.com/en/blog/one-take-creation-flexible-referencing-introducing-seedance-2-5

Act VI · Seedance 1.0–2.5

SLIDE 060 · A foundation built across the whole lifecycle

SEEDANCE 1.0

62 words · ≈1 minute

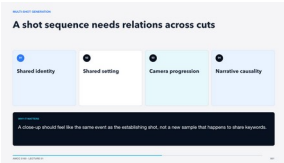
The 1.0 report emphasizes that architecture alone was not enough. Multi-source curation and precise video captioning improved supervision. One model handled text-to-video and image-to-video. Post-training used multidimensional reward signals, and engineering plus distillation targeted practical speed.

The report gives a useful production lesson: quality emerges from the whole pipeline, not from one clever block.

Its native multi-shot ability anticipates later storytelling features.

Sources: https://arxiv.org/abs/2506.09113

SLIDE 061 · A shot sequence needs relations across cuts



MULTI-SHOT GENERATION

69 words · ≈1 minute

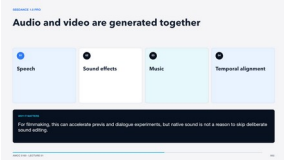
Multi-shot generation means more than producing several pretty clips. The system must preserve entities and setting while allowing viewpoint and action to change. That requires modeling relations across shot boundaries rather than only smooth motion within one shot.

A close-up should feel like the same event as the establishing shot, not a new sample that happens to share keywords.

Seedance 1.5 Pro added a second difficult axis: native sound.

Sources: https://arxiv.org/abs/2506.09113

SLIDE 062 · Audio and video are generated together



SEEDANCE 1.5 PRO

60 words · ≈1 minute

The 1.5 Pro report presents a native audio-visual joint generation foundation model. Joint generation lets visual events and sound influence a shared process instead of relying entirely on post-hoc audio generation.

For filmmaking, this can accelerate previs and dialogue experiments, but native sound is not a reason to skip deliberate sound editing.

Seedance 2.0 expands the input side as well.

Sources: https://arxiv.org/abs/2512.13507

SLIDE 063 · Four input modalities feed one audio-video generator

SEEDANCE 2.0

63 words · ≈1 minute

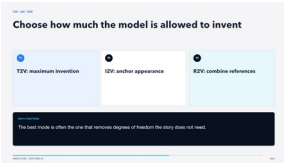
Seedance 2.0 accepts text, images, audio, and video as multimodal context and supports generation plus editing. The technical report describes a unified, large-scale architecture but does not publish enough block-level detail to reconstruct it.

That disclosure boundary matters. We can discuss supported modalities and evaluations confidently, while treating internal tokenization and transformer design as unknown.

The system exposes three especially useful task families.

Sources: https://arxiv.org/abs/2604.14148

SLIDE 064 · Choose how much the model is allowed to invent



T2V · I2V · R2V

57 words · ≈1 minute

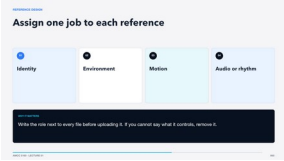
Text-to-video is appropriate for ideation and broad exploration. Image-to-video is better when composition or identity already exists. Reference-to-video can combine images, clips, and audio to specify appearance, motion, rhythm, or voice.

The best mode is often the one that removes degrees of freedom the story does not need.

Reference capacity is not the same as reference clarity.

Sources: https://arxiv.org/abs/2604.14148

SLIDE 065 · Assign one job to each reference



REFERENCE DESIGN

66 words · ≈1 minute

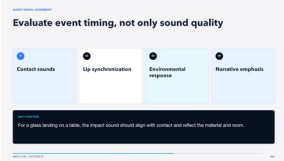
A clean reference stack gives each file a role. One image anchors character identity, another defines environment, a short video specifies motion, and audio establishes performance or rhythm. Redundant or contradictory references make the control problem harder.

Write the role next to every file before uploading it. If you cannot say what it controls, remove it.

Seedance 2.0 reports direct audio-video generation over short shot durations.

Sources: https://arxiv.org/abs/2604.14148

SLIDE 066 · Evaluate event timing, not only sound quality



AUDIO-VISUAL ALIGNMENT

65 words · ≈1 minute

The Seedance 2.0 evaluation separates audio quality, audio prompt following, and audio-visual synchronization. That is the right conceptual split. A beautiful soundtrack can still be wrong if it ignores events or changes the scene's emotional direction.

For a glass landing on a table, the impact sound should align with contact and reflect the material and room.

Multi-shot and longer-form control are where 2.5 moves next.

Sources: https://arxiv.org/abs/2604.14148

SLIDE 067 · One-take creation extends a controlled event

SEEDANCE 2.5

64 words · ≈1 minute

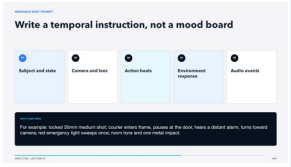
ByteDance describes 2.5 as inheriting the unified multimodal audio-video architecture of 2.0 while improving long-form storytelling, flexible references, and editing. A single generation can reach thirty seconds, with further extensions.

That duration is useful only when the shot has internal beats. ‘Thirty seconds of cinematic motion’ is weaker than a timed sequence of setup, action, reaction, and release.

Prompting should therefore resemble shot direction.

Sources: https://seed.bytedance.com/en/blog/one-take-creation-flexible-referencing-introducing-seedance-2-5

SLIDE 068 · Write a temporal instruction, not a mood board



SEEDANCE SHOT PROMPT

63 words · ≈1 minute

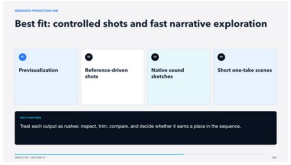
A strong prompt names the starting state, camera behavior, ordered action, environmental consequence, and sound. Separate fixed constraints from events that change over time.

For example: locked 35mm medium shot; courier enters frame, pauses at the door, hears a distant alarm, turns toward camera; red emergency light sweeps once; room tone and one metal impact.

Use one generation to test one shot proposition.

Sources: https://seed.bytedance.com/en/blog/one-take-creation-flexible-referencing-introducing-seedance-2-5

SLIDE 069 · Best fit: controlled shots and fast narrative exploration



SEEDANCE PRODUCTION USE

66 words · ≈1 minute

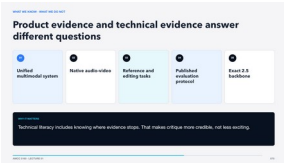
Seedance is valuable when a team needs to explore an event with multiple reference types, preview audio-video timing, or create a longer shot with internal development. It is not a replacement for editing, coverage, or continuity management across a whole film.

Treat each output as rushes: inspect, trim, compare, and decide whether it earns a place in the sequence.

The technical unknowns also require disciplined claims.

Sources: https://arxiv.org/abs/2604.14148 · https://seed.bytedance.com/en/blog/one-take-creation-flexible-referencing-introducing-seedance-2-5

SLIDE 070 · Product evidence and technical evidence answer different questions



WHAT WE KNOW · WHAT WE DO NOT

64 words · ≈1 minute

The reports support claims about modalities, tasks, duration, and evaluation. They do not fully reveal the 2.5 backbone, parameter count, token packing, or training data composition. Saying 'probably a DiT' is an inference, not a disclosed fact.

Technical literacy includes knowing where evidence stops. That makes critique more credible, not less exciting.

MiniMax H3 gives us a contrasting case with unusually detailed open documentation.

Sources: https://arxiv.org/abs/2604.14148 · https://seed.bytedance.com/en/blog/one-take-creation-flexible-referencing-introducing-seedance-2-5

SLIDE 071 · H3

SECTION 6 · MINIMAX

79 words · ≈1 minute

We now turn to H3, MiniMax's open audio-video foundation model. H3 is important because the release describes a complete multimodal generation system rather than only a product interface: structured context, a large omni-modal transformer, separate visual and audio latent spaces, and a high-resolution regeneration stage.

The architecture gives us enough detail to connect familiar ideas—encoders, patchification, attention, latent compression, and conditioning—to a contemporary production model.

Begin with the system-level path from the user's material to the final 2K result.

Sources: https://www.minimax.io/news/minimax-h3-open-source · https://www.minimax.io/blog/minimax-h3

Act VII · MiniMax H3

SLIDE 072 · Context-IR → H3-Base → Regenerate-2K

H3 SYSTEM

63 words · ≈1 minute

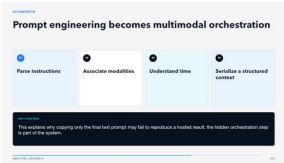
H3-Context-IR turns free-form multimodal inputs into a structured intermediate representation. H3-Base generates synchronized audio and video at 768p. H3-Regenerate-2K feeds the result and original context back into the model for a higher-resolution regeneration.

Notice that 2K is not described as ordinary super-resolution. The system reuses semantic context while generating detail again.

Context-IR is essential, but it is not part of the open weights.

Sources: https://www.minimax.io/news/minimax-h3-open-source

SLIDE 073 · Prompt engineering becomes multimodal orchestration



H3-CONTEXT-IR

70 words · ≈1 minute

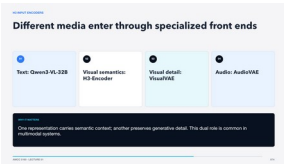
Context-IR reasons about how text, images, video, and audio relate to the intended output. MiniMax says the hosted system may supplement underspecified details without changing the user's intent. It distills large source material into a much smaller context representation.

This explains why copying only the final text prompt may fail to reproduce a hosted result: the hidden orchestration step is part of the system.

H3-Base then encodes each modality differently.

Sources: https://www.minimax.io/news/minimax-h3-open-source · https://www.minimax.io/blog/minimax-h3

SLIDE 074 · Different media enter through specialized front ends



H3 INPUT ENCODERS

54 words · ≈1 minute

The H3 encoder uses full pretrained Qwen3-VL-32B weights and passes hidden states from layer fifty to the generator. Visual inputs also go through the VisualVAE, while audio uses a separate AudioVAE.

One representation carries semantic context; another preserves generative detail. This dual role is common in multimodal systems.

The visual compressor is unusually aggressive.

Sources: https://www.minimax.io/news/minimax-h3-open-source

SLIDE 075 · 16× spatial, 4× temporal, then 2×2 patchification

H3 VISUAL VAE

57 words · ≈1 minute

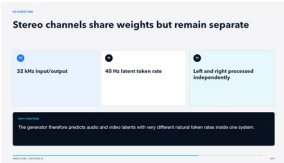
H3-VisualVAE is temporally causal with a spatial compression factor of sixteen, temporal factor of four, and twenty-four latent channels. A one-by-two-by-two patchification further halves each spatial dimension before the transformer.

That creates an effective thirty-two-times spatial downsampling at the token level, dramatically reducing sequence length while demanding a very capable decoder.

Audio follows a separate compression path.

Sources: https://www.minimax.io/news/minimax-h3-open-source

SLIDE 076 · Stereo channels share weights but remain separate



H3 AUDIO VAE

58 words · ≈1 minute

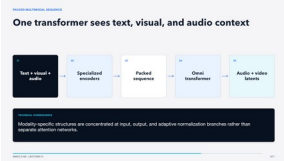
H3-AudioVAE uses the same encoder and decoder parameters for left and right channels while processing them independently, then recombines them as stereo. Each channel's 32-kilohertz audio becomes latent tokens at forty hertz.

The generator therefore predicts audio and video latents with very different natural token rates inside one system.

Those tokens are packed into a single multimodal sequence.

Sources: https://www.minimax.io/news/minimax-h3-open-source

SLIDE 077 · One transformer sees text, visual, and audio context



PACKED MULTIMODAL SEQUENCE

57 words · ≈1 minute

H3 organizes encoded modalities into one packed sequence, adds rotary position information for spatial and temporal relationships, and passes the result to a single-stream transformer. The transformer jointly predicts visual and audio latents.

Modality-specific structures are concentrated at input, output, and adaptive normalization branches rather than separate attention networks.

The core transformer is large but architecturally simple.

Sources: https://www.minimax.io/news/minimax-h3-open-source

SLIDE 078 · 33B dense, single-stream, task-general

H3 OMNI TRANSFORMER

60 words · ≈1 minute

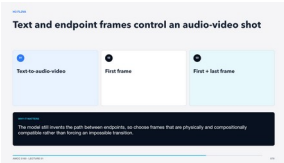
MiniMax reports a thirty-three-billion-parameter dense transformer. Roughly thirteen billion parameters belong to adaptive layer-normalization branches whose outputs can be precomputed and cached, so inference-only deployments do not need to load them in the same way.

The attention and feed-forward layers are modality-agnostic. That simplicity is intended to support many tasks without separate backbones.

Two released checkpoints specialize the input conditions.

Sources: https://www.minimax.io/news/minimax-h3-open-source

SLIDE 079 · Text and endpoint frames control an audio-video shot



H3 FL2VA

55 words · ≈1 minute

The FL2VA checkpoint supports text-only generation and first/last-frame-conditioned audio-video generation. Endpoint frames are powerful when the narrative requires a defined entrance and exit state.

The model still invents the path between endpoints, so choose frames that are physically and compositionally compatible rather than forcing an impossible transition.

The Ref2VA checkpoint accepts a richer reference package.

Sources: https://www.minimax.io/news/minimax-h3-open-source

SLIDE 080 · References can specify more than appearance

H3 REF2VA

60 words · ≈1 minute

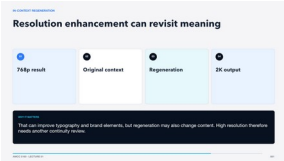
Ref2VA accepts multimodal references within a twelve-file total. Images can anchor characters or style, videos can supply motion or camera language, and audio can define voice, rhythm, or event timing.

Again, the limit is not a target. A smaller set with explicit roles reduces ambiguity and makes failures easier to diagnose.

The final 2K stage regenerates rather than merely enlarges.

Sources: https://www.minimax.io/news/minimax-h3-open-source

SLIDE 081 · Resolution enhancement can revisit meaning



IN-CONTEXT REGENERATION

60 words · ≈1 minute

Conventional super-resolution mainly predicts missing high-frequency detail from a low-resolution image or video. H3-Regenerate-2K conditions on both the 768p result and the original multimodal context, allowing text and references to reassert fine detail.

That can improve typography and brand elements, but regeneration may also change content. High resolution therefore needs another continuity review.

Open deployment brings a different constraint: memory.

Sources: https://www.minimax.io/news/minimax-h3-open-source · https://www.minimax.io/blog/minimax-h3

SLIDE 082 · Open weights still require production infrastructure

H3 DEPLOYMENT

65 words · ≈1 minute

The open release provides BF16 checkpoints for FL2VA and Ref2VA, with support paths through SGLang, vLLM, Diffusers, and ComfyUI. Sparse attention is described as native, but the first released inference path uses full attention.

A realistic class workflow may use hosted generation or cloud nodes while using ComfyUI to make the pipeline explicit and reproducible.

Now compare the systems without pretending they publish identical evidence.

Sources: https://www.minimax.io/news/minimax-h3-open-source

SLIDE 083 · Choose the system that fits the shot

MODEL SELECTION

71 words · ≈1 minute

Seedance emphasizes polished multimodal creation and longer one-take workflows. H3 exposes a detailed open omni-modal architecture. Wan offers a broad open ecosystem with strong ComfyUI support. HunyuanVideo 1.5 targets efficient consumer-grade inference with an 8.3B model and a cascaded super-resolution stage.

The right model is the one whose controllable task matches the shot and whose deployment constraints fit the production.

We can now turn architecture knowledge into a repeatable film workflow.

Sources: https://seed.bytedance.com/en/blog/one-take-creation-flexible-referencing-introducing-seedance-2-5 · https://www.minimax.io/news/minimax-h3-open-source · https://arxiv.org/abs/2503.20314 · https://arxiv.org/abs/2511.18870

Act VIII · AI filmmaking workflow

SLIDE 084 · A film is a sequence of decisions, not one generation

SECTION 7 · AI FILMMAKING

44 words · ≈1 minute

Current models are shot engines, reference engines, and editing engines. A film still requires dramatic structure, coverage, continuity, selection, sound, rhythm, and responsibility.

The workflow becomes reliable when each stage produces an inspectable artifact for the next stage.

Begin before prompts, with a brief.

Sources: https://arxiv.org/abs/2603.23415

SLIDE 085 · Five gates prevent expensive confusion

PRODUCTION PIPELINE

57 words · ≈1 minute

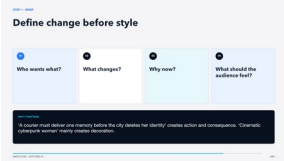
The brief defines purpose. The world bible fixes continuity. The animatic tests sequence and duration. Shot generation produces controlled candidates. Post-production creates rhythm, sound, repair, color, titles, and disclosure.

Do not move forward because a stage feels exciting. Move forward because its artifact answers the decisions needed by the next stage.

Gate one is a dramatic unit.

Sources: https://arxiv.org/abs/2603.23415

SLIDE 086 · Define change before style



STEP 1 · BRIEF

65 words · ≈1 minute

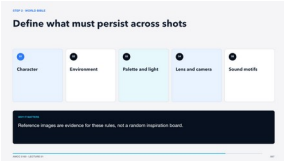
A useful brief can fit in four sentences. Name the agent, goal, change, stakes, and intended audience experience. Avoid beginning with visual adjectives, because style cannot rescue an event with no dramatic direction.

'A courier must deliver one memory before the city deletes her identity' creates action and consequence. 'Cinematic cyberpunk woman' mainly creates decoration.

The world bible turns that brief into stable visual rules.

Sources: https://arxiv.org/abs/2603.23415

SLIDE 087 · Define what must persist across shots



STEP 2 · WORLD BIBLE

57 words · ≈1 minute

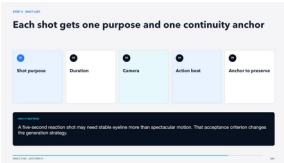
The bible should identify invariants and allowed variation. Character silhouette and coat color may be fixed while fabric texture can vary. The room geometry may be fixed while light state changes. The camera may remain observational rather than heroic.

Reference images are evidence for these rules, not a random inspiration board.

Next convert story beats into coverage.

Sources: https://arxiv.org/abs/2603.23415

SLIDE 088 · Each shot gets one purpose and one continuity anchor



STEP 3 · SHOT LIST

64 words · ≈1 minute

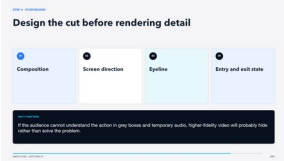
Write the shot list before generation. A shot should answer why it exists in the edit, how long it needs to be, what the camera does, which action completes, and what must match adjacent shots.

A five-second reaction shot may need stable eyeline more than spectacular motion. That acceptance criterion changes the generation strategy.

Storyboard and animatic expose sequence problems while changes are cheap.

Sources: https://arxiv.org/abs/2603.23415

SLIDE 089 · Design the cut before rendering detail



STEP 4 · STORYBOARD

65 words · ≈1 minute

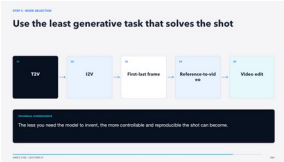
Storyboard frames establish spatial relations and continuity. The animatic adds approximate timing, temporary sound, and cuts. It is acceptable for these materials to be rough; their job is to reveal missing geography and rhythm.

If the audience cannot understand the action in grey boxes and temporary audio, higher-fidelity video will probably hide rather than solve the problem.

Now choose a generation mode for each shot.

Sources: https://arxiv.org/abs/2603.23415

SLIDE 090 · Use the least generative task that solves the shot



STEP 5 · MODE SELECTION

58 words · ≈1 minute

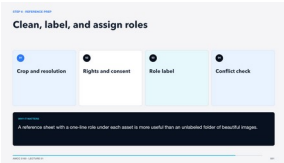
Choose task mode shot by shot. Text-to-video is useful when invention is desirable. Image-to-video preserves a designed frame. First-last-frame controls endpoints. Reference generation anchors reusable assets. Video-to-video or editing preserves blocking and timing.

The less you need the model to invent, the more controllable and reproducible the shot can become.

Reference preparation is therefore a technical production task.

Sources: https://arxiv.org/abs/2604.14148 · https://www.minimax.io/news/minimax-h3-open-source · https://docs.comfy.org/tutorials/video/wan/wan2_2

SLIDE 091 · Clean, label, and assign roles



STEP 6 · REFERENCE PREP

59 words · ≈1 minute

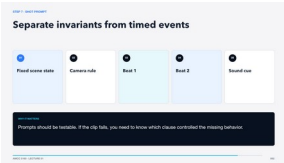
Prepare references at sensible resolution, remove irrelevant borders or overlays, verify rights and consent, and label what each file controls. Check for conflicts in age, costume, lighting, viewpoint, or movement.

A reference sheet with a one-line role under each asset is more useful than an unlabeled folder of beautiful images.

Then write the shot instruction as a timed system.

Sources: https://arxiv.org/abs/2604.14148 · https://www.minimax.io/news/minimax-h3-open-source

SLIDE 092 · Separate invariants from timed events



STEP 7 · SHOT PROMPT

59 words · ≈1 minute

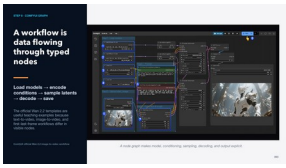
Begin with what must remain fixed: subject, environment, lens, lighting, and style constraints. Then state ordered events with simple timing words. End with negative or exclusion constraints only when they address observed failures.

Prompts should be testable. If the clip fails, you need to know which clause controlled the missing behavior.

ComfyUI makes this pipeline visible as a graph.

Sources: https://seed.bytedance.com/en/blog/one-take-creation-flexible-referencing-introducing-seedance-2-5 · https://www.minimax.io/news/minimax-h3-open-source

SLIDE 093 · A workflow is data flowing through typed nodes



STEP 8 · COMFYUI GRAPH

60 words · ≈1 minute

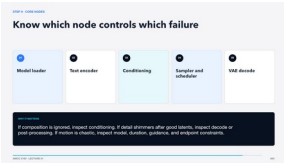
ComfyUI represents the generation pipeline as connected nodes. The graph makes hidden dependencies explicit: checkpoint, text encoder, VAE, conditioning, latent dimensions, sampler, scheduler, and output. Workflow JSON also becomes a reproducible production record.

The official Wan 2.2 templates are useful teaching examples because text-to-video, image-to-video, and first-last-frame workflows differ in visible nodes.

Read the graph from inputs toward media output.

Sources: https://docs.comfy.org/tutorials/video/wan/wan2_2

SLIDE 094 · Know which node controls which failure



STEP 9 · CORE NODES

76 words · ≈1 minute

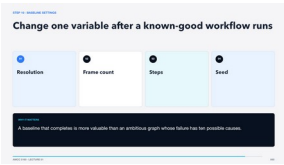
The loader selects weights and precision. The text encoder converts language into condition vectors. Image or frame nodes add visual constraints. The sampler integrates the generative trajectory. The VAE decodes latents into frames. Video combine nodes set frame rate and container output.

If composition is ignored, inspect conditioning. If detail shimmers after good latents, inspect decode or post-processing. If motion is chaotic, inspect model, duration, guidance, and endpoint constraints.

Now set a controlled baseline before tuning.

Sources: https://docs.comfy.org/tutorials/video/wan/wan2_2

SLIDE 095 · Change one variable after a known-good workflow runs



STEP 10 · BASELINE SETTINGS

59 words · ≈1 minute

Start from an official template and recommended model settings. Choose a modest resolution and duration that fit hardware. Record frame count, frame rate, steps, guidance, sampler, scheduler, and seed. Run once before adding custom nodes.

A baseline that completes is more valuable than an ambitious graph whose failure has ten possible causes.

Controlled batches turn generation into an experiment.

Sources: https://docs.comfy.org/tutorials/video/wan/wan2_2

SLIDE 096 · Hold the shot constant; vary one decision

STEP 11 · CONTROLLED BATCH

57 words · ≈1 minute

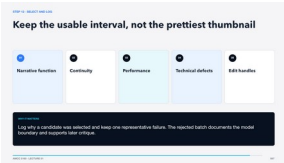
Decide what you are testing: camera strength, endpoint frame, guidance, motion reference, or prompt wording. Hold the other important conditions stable. Generate three to six candidates and write observations before choosing a favorite.

Changing model, seed, prompt, duration, and reference at once may create variety, but it creates almost no knowledge.

Selection now becomes an editorial decision.

Sources: https://docs.comfy.org/tutorials/video/wan/wan2_2

SLIDE 097 · Keep the usable interval, not the prettiest thumbnail



STEP 12 · SELECT AND LOG

66 words · ≈1 minute

Review clips at full speed, frame by frame, and with neighbouring shots. Mark the usable in and out points. Judge whether the action completes, the identity holds, the eyeline matches, and there are enough frames for the cut.

Log why a candidate was selected and keep one representative failure. The rejected batch documents the model boundary and supports later critique.

Post-production restores control across the sequence.

Sources: https://arxiv.org/abs/2603.23415

SLIDE 098 · Edit first; repair only what survives the cut

STEP 13 · POST

69 words · ≈1 minute

Build the picture edit before spending time on repair. Composite or inpaint only shots that survive. Design dialogue, ambience, Foley, and music at sequence level. Interpolate when motion needs it, upscale after editorial lock, then color and title the film.

Recent filmmaking research describes AI as reconfiguring production roles and timing; it does not remove the need for editorial judgment.

The final gate is quality control and responsible release.

Sources: https://arxiv.org/abs/2603.23415 · https://docs.comfy.org/tutorials/utility/frame-interpolation · https://docs.comfy.org/tutorials/utility/video-upscale

SLIDE 099 · Check the film as evidence, experience, and record

STEP 14 · QC AND RELEASE

66 words · ≈1 minute

Watch once for story and feeling, once for continuity, once for audio synchronization, and once for technical artifacts. Confirm rights, consent, and credits. Record model names, versions, dates, prompts, references, important settings, edits, and disclosure language.

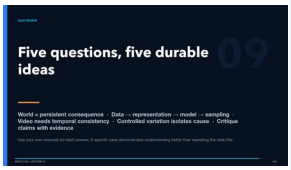
Provenance is not paperwork added after creativity. It is what lets collaborators understand and responsibly stand behind the work.

We can now return to the quiz with a complete map.

Sources: https://arxiv.org/abs/2603.23415

Act IX · Review and challenge

SLIDE 100 · Five questions, five durable ideas



QUIZ REVIEW

66 words · ≈1 minute

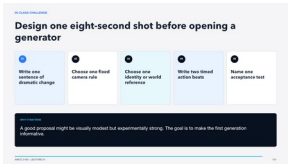
These are the five ideas the quiz asks you to explain. Notice that they also summarize the lecture: what makes a world, how a generator works, why video is difficult, how to experiment, and how to critique current claims.

Use your own example for each answer. A specific case demonstrates understanding better than repeating the slide title.

One final production exercise turns the lecture into action.

Sources: https://hyang.org/amcc5160/quiz/

SLIDE 101 · Design one eight-second shot before opening a generator



IN-CLASS CHALLENGE

74 words · ≈1 minute

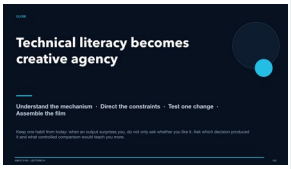
Work in pairs for a few minutes. Do not generate yet. Write an eight-second shot with one meaningful change, a camera constraint, a reference role, two ordered beats, and one acceptance test. Then choose whether T2V, I2V, first-last-frame, or reference-to-video gives the model the least unnecessary freedom.

A good proposal might be visually modest but experimentally strong. The goal is to make the first generation informative.

We finish by returning to the opening promise.

Sources: https://docs.comfy.org/tutorials/video/wan/wan2_2

SLIDE 102 · Technical literacy becomes creative agency



CLOSE

78 words · ≈1 minute

A neural network learns transformations. CNNs build local visual hierarchies. Transformers relate tokens globally. VAEs compress media. Diffusion and flow learn generative paths. Video systems extend those ideas across time and modalities. Filmmaking turns the resulting shots into meaning.

Keep one habit from today: when an output surprises you, do not only ask whether you like it. Ask which decision produced it and what controlled comparison would teach you more.

That is where technical literacy becomes creative agency.

Sources: https://arxiv.org/abs/1706.03762 · https://arxiv.org/abs/2112.10752 · https://arxiv.org/abs/2212.09748 · https://arxiv.org/abs/2604.14148 · https://www.minimax.io/news/minimax-h3-open-source