Table 2: Summarized Objects in Mixed-media Tutorials by Human Roles

Human Role	Topic	Source Count
Create manually	General [80], Cooking [49, 83], Lecture [43]	4
No intervention	Software [24]	1
Refine computational results	Cooking [13, 55]	2

Table 3: The Roles of Human in Extracting Steps in Mixed-media Tutorials

Human Role	Topic	Source Count
Create manually	General [49, 60, 80], Cooking [83], Lecture [79]	5
No intervention	General [72], Software [15, 24], Makeup [74]	4
Provide input for computation	General [36], Software [77]	2
Refine computational results	Cooking [13, 55], Lecture [59]	3

Table 4: Summarized Dependencies in Mixed-media Tutorials

Human Role	Topic and Relation	Source Count
Create manually	Cooking: food processing order [55, 83], spatial relations [83]; Lecture: concept prerequisites [43]	4
No intervention	Makeup: spatial relations [74]	1
Refine/Input for computational methods	Cooking: food processing order [13]	1

A APPENDIX

A.1 Model evaluation details

Dataset: YouCook2 [91] comprising 2000 untrimmed cooking videos with human annotations, averaging 5.27 minutes in length and containing 3-16 steps per video. Each step is annotated with the start time, end time, and text descriptions. The dataset splits are training (67%), validation (23%), and testing (10%). Only the training and validation sets have object annotations (bounding boxes and labels). Since some models were pre-trained on the training subset, we exclusively utilize the validation set. After filtering for auto-generated English transcripts, 347 videos remain. Auto-generated transcripts for each video were sourced from the YouTube API [63].

Extractive methods necessitate the parameter step count, K. For consistent benchmarking, we set K as the ground truth steps of each video. For LexRank [18] and TextRank [52], we used Sumy's implementation [5]. For LLM prompting, we prompted GPT-3 with "summarize the recipe in K steps". For BART [21] and T5 [64], we used the HuggingFace [20] implementation of both methods with default parameters.

ROUGUE-n scores measure the overlap of n-grams between generated and ground-truth summaries, and ROUGE-L is the Longest Common Subsequence (LCS)-based statistics.

For pipeline 2, we abstained from gauging the efficacy of shot boundary detection methods in extracting step thumbnails since no ground-truth thumbnails exist, and shot boundary detection yields multiple frame candidates.

For video dense captioning, we assume the ground-truth step timestamps are known. Since the goal of dense captioning is not summarization, but scene description, we do not compute ROUGE scores but manually inspect the results. After reviewing segment descriptions from a sample of 20 videos, errors are evident in object names and actions. For example, in the video "How to Make Fried Calamari | Hilah Cooking"[9], the human annotation is "drop the squid pieces into the oil", but the dense captioning returns "add the chicken in a pot of water boil".

Method	Type	ROUGE-1	ROUGE-2	ROUGE-L
LexRank	extractive	0.25	0.06	0.21
TextRank	extractive	0.25	0.06	0.20
BART	abstractive	0.22	0.03	0.19
T5	abstractive	0.19	0.02	0.16
GPT-3	abstractive	**0.37**	**0.12**	**0.31**

Table 5: Average ROUGE (Recall-Oriented Understudy for Gisting Evaluation) F1 scores of different summarization methods.

For POS taggers, we designated words labeled as NN, NNS, NNP, or NNPS [61] as nouns. For traditional object detector, we chose faster-r-CNN trained on Visual Genomes due to benefit from the large number of object categories. We down-sampled videos to one frame every 10 seconds and retained detected objects with confidence scores above 0.4, selecting the top 10 objects per frame.

Open-vocabulary detectors: we evaluated both OWL-ViT and MDETR [34]. For OWL-ViT, we provided OWL-ViT with object

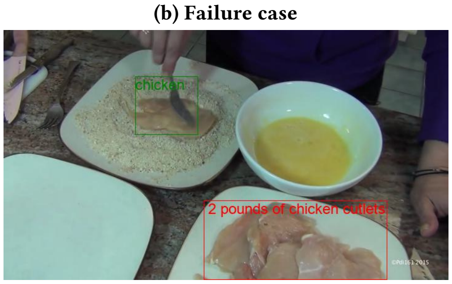
[9] https://www.youtube.com/watch?v=-k7trpuj3X8

Method	True positives	Label unavailable	Missing	False positives
Visual Detector [66]	2.8	2.9	4.2	43.6
POS tagging [3]	7.0	**1.1**	**1.5**	32.4
GPT-3 with prompt	**7.4**	**1.1**	**1.5**	**6.8**

Table 6: A quantitative comparison of object detection methods. On average, videos contain 9.6 ground-truth objects. Label unavailable: the object is not in the Visual Genome [37] dataset or is unmentioned in the transcript. Missing: fails to detect the object when the label is available. False positives: detections irrelevant to the cooking process.

(a) Successful case

(b) Failure case



(c) Successful case not in ground-truth

Figure 11: Image grounding examples returned by GPT-3 + OWL-ViT, an open-vocabulary detector. Green box: human annotation; red: returned by OWL-ViT. (a) GPT3: "fish sauce"; ground-truth: "sauce"; (b) GPT3: "salt"; ground-truth: "salt"; (c): GPT3: "2 pounds of chicken cutlets"; ground-truth: "chicken"; though the IOU is 0, it's a correct detection.

names extracted by GPT-3 from the transcript, among 3440 objects returned by GPT-3 that are also included in the human annotations, the mean IOU (Intersection over Union) of the ground truth bounding boxes and the predicted bounding boxes is 0.38. Examples of success and failure cases are shown in Figure 11. For MDETR [34], it has similar results, but the inference cost is much higher, therefore, we chose the HuggingFace implementation [19] of OWL-ViT [53].

A.2 Limitations of ML pipelines

We noticed two bottlenecks in our ML pipeline. One is the maximum number of tokens the text summarization method can take: Currently, we use GPT-3/3.5 API to process transcripts, which has a limit of 4096 tokens (a token is about 0.75 word) in a single round of conversation, e.g., both input and GPT-generated output. Empirically, that's a 10-15 min instructional video's transcript length and summarized steps. Fortunately, we see progress in this area, e.g., the newly released GPT-4 supports at most 32768 tokens [50].

Another bottleneck is the open-vocabulary object detector. In the user studies, AI-generated bounding boxes received the lowest quality scores from participants. As the vision-language model is still an emerging research area, we expect the results to improve steadily in the future.

We also noticed the hallucination problems of LLM, e.g., it generates details like "4 eggs" and "all-purpose flour" when the transcript only mentions "eggs" and "flour". Other factors also influence step summarization quality, including automatic speech recognition (ASR) errors, shown in Table 8.

A.3 Generality of TutoAI

We showed TutoAI's consistent performance in instructional videos across domains via user studies, including cooking, furniture assembly, craft, and vehicle. Unlike previous work which focus specifically on a single domain [55, 74], TutoAI has demonstrated its versatility empowered by LLMs and vision-language models.

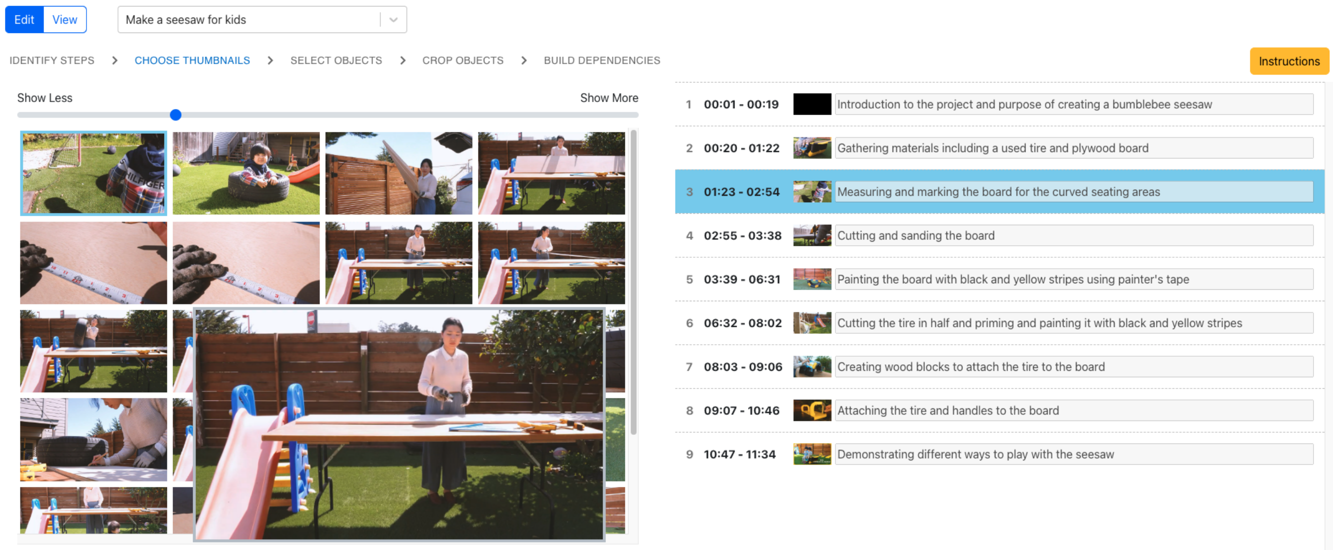


Figure 12: Choose thumbnails. The goal is to choose a representative image for each step. On the left are video frames selected by TutoAI. Hovering over a frame will show an enlarged version. Creators can control the number of displayed frames by dragging the slider toward "show more"/"show less." On the right are steps (now the editing is disabled).

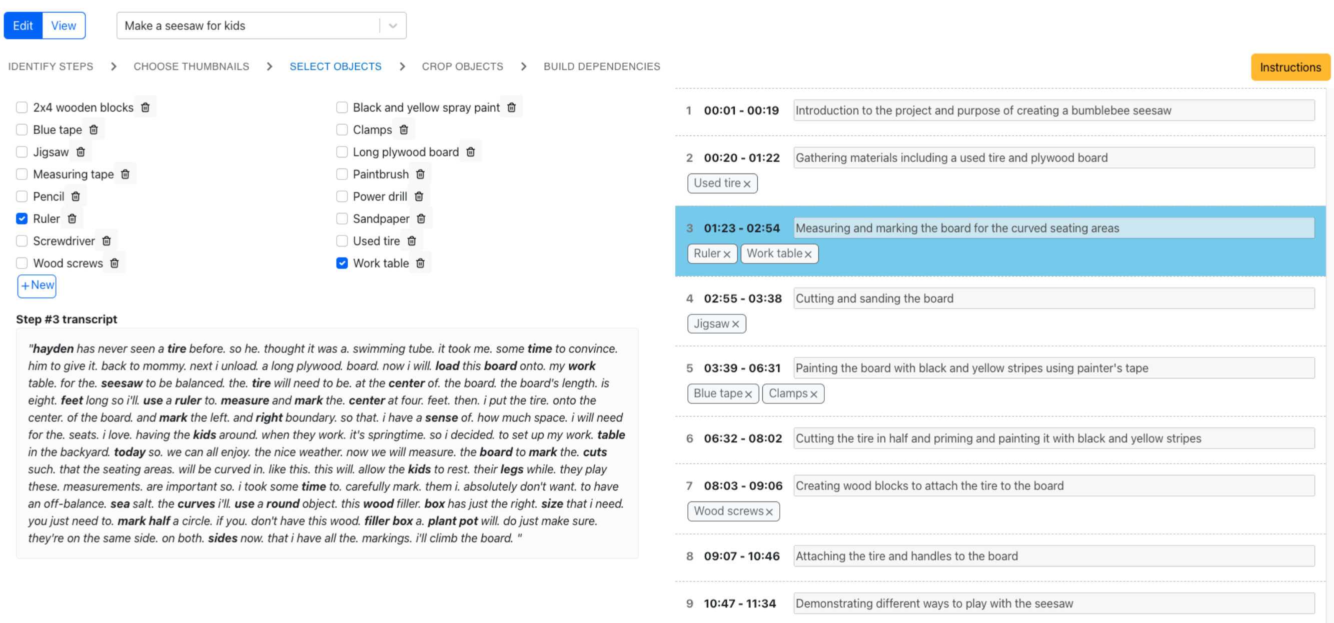


Figure 13: Select objects. The goal is to associate objects with each step to build a dependency between steps in later stages. Creators can add and delete objects in each step, add new objects, and delete objects for the entire video

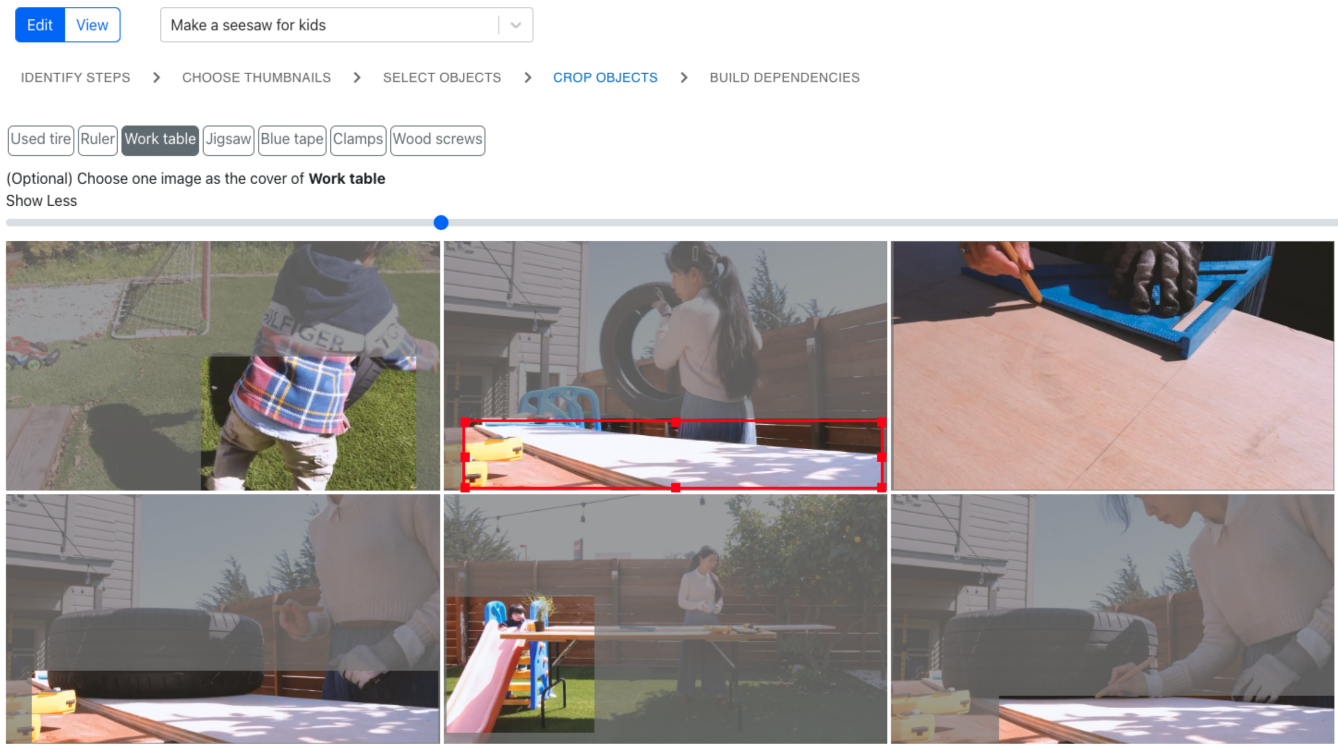


Figure 14: Crop objects. The goal is to provide object images for less common objects. Here, it shows recommended images for the "work table." Once an image is selected, creators can adjust the bounding boxes

Method	Error types	Examples	Video ID
Visual detector	label unavailable	"dough"	4K9h7ojJYkc
Visual detector	missing	"shrimp"	GXnzgRC3sd4
Visual detector	wrong	mistook "pan" for "bowl"	tGaAAI3aAUs
Visual detector	false positive	"necklace"	abfhnSaZFlA
POS Tagging/GPT-3	label unavailable	"wok"	eWBSMD3BiHM
POS Tagging	missing	"chickpea"	R5IAGR2SeaE
POS Tagging	wrong	"soy sauce"=> "soy", "sauce"	ntiGX3X-spA
POS Tagging	false positive	"minutes"	tYg3lQ5aZv8
GPT-3	missing	"water"	jEo9VXYVrxs
GPT-3	wrong	"Cat cat spices"	luDzsPatsGw
GPT-3	false positive	"Clean hands"	7-FatJyHj_g

Table 7: Error examples in object detection methods

Error types	Examples	Video ID
ASR error	"...put off the plane" should be "put off the flame"	ikmPrpgWQ5M
object/action unmentioned	"here's an egg put that in there" (didn't mention the "bowl")	TF1iWaX2-DM
video-text discrepancy	talk about animal welfare while chopping a cabbage	Z5bpo2sBsl8

Table 8: Other error types that influence text summary quality

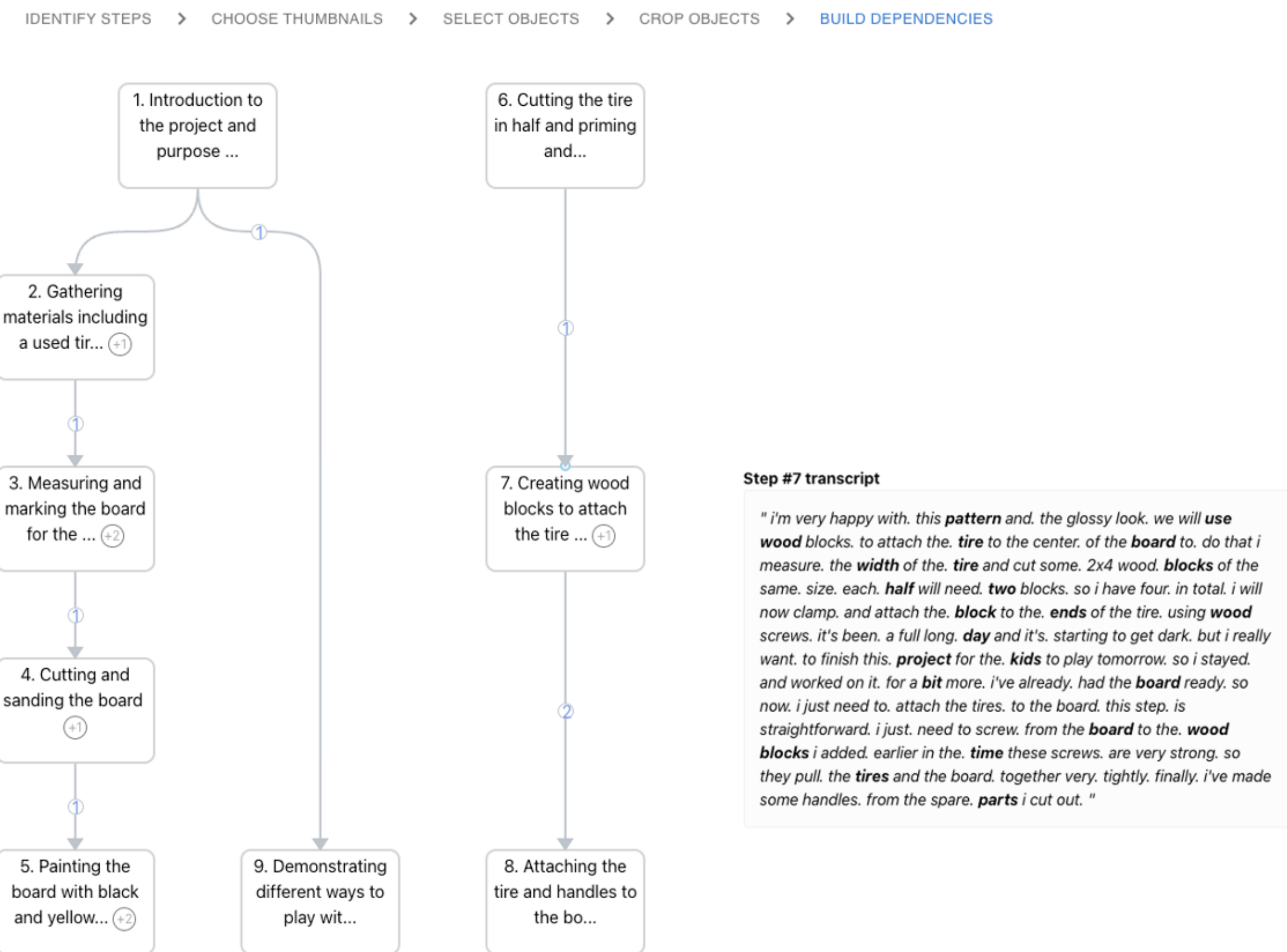


Figure 15: Build dependencies. The goal is to build dependencies between steps so consumers can easily skip and split tasks. To add new dependencies, creators start a new arrow from a step and connect the arrow to another. To delete a dependency, drag the arrow away from a step and release. To help creators recall the content of each step, hovering over a step will display its transcript at the bottom right.

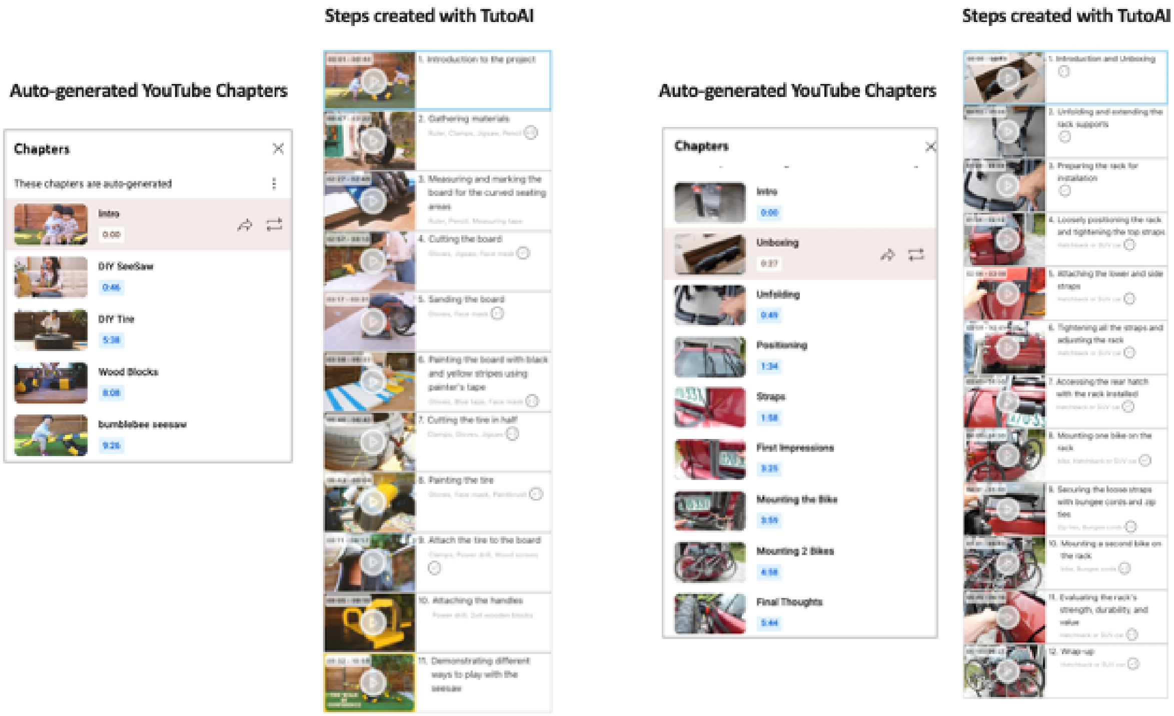
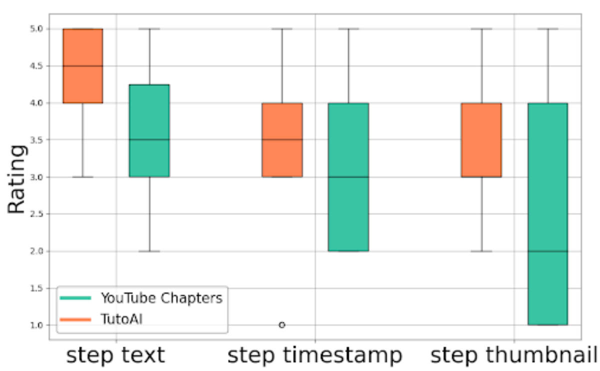
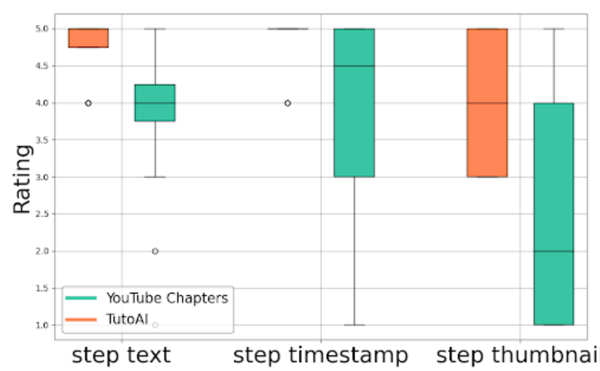


Figure 16: YouTube auto-generated chapters vs. TutoAI steps created by original authors



(a) Before editing: components quality comparison. TutoAI vs. YouTube Chapters, text: 4.4±0.64 vs. 3.6±1.04 (p=0.138); timestamps: 3.3±1.25 vs. 3.0±1.0 (p=1.000); thumbnails: 3.4±0.76 vs. 2.4±1.38 (p=0.138)



(b) After editing: components usefulness comparison. TutoAI vs. YouTube Chapters, text: 4.8±0.43 vs. 3.8±1.16 (p=0.063), timestamps: 4.8±0.37 vs. 4.0±1.22 (p=0.192), thumbnails: 4.0±0.91 vs. 2.6±1.50 (p=0.153)

Figure 17: Component quality of group B: strawberry blueberry shortcakes. Group B. Before editing (left), after editing (right)

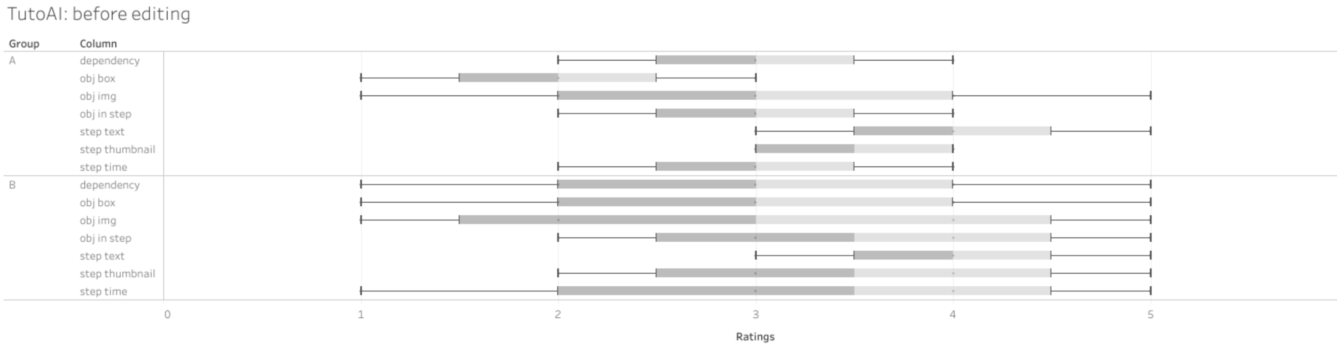


Figure 18: Before editing: TutoAI components quality. Group A: office chair assembly, Group B: strawberry blueberry shortcakes

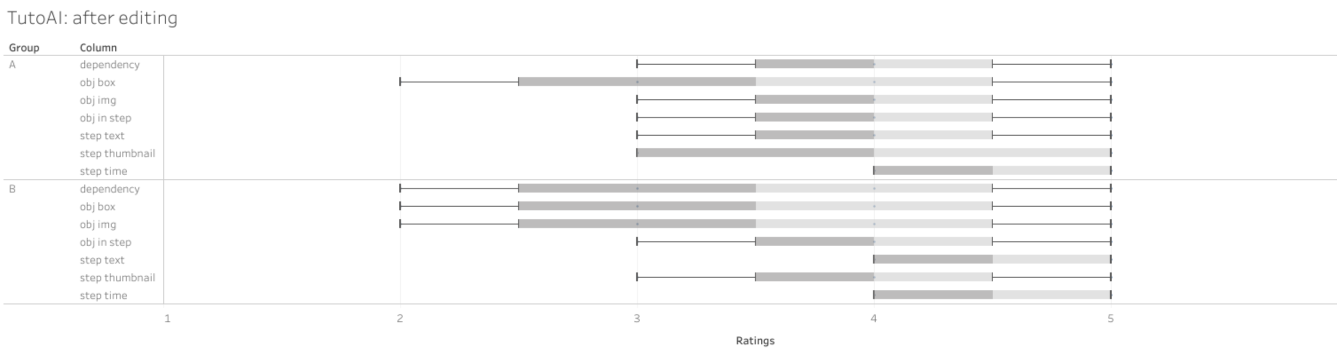


Figure 19: After editing: TutoAI components usefulness. Group A: office chair assembly, Group B: strawberry blueberry shortcakes

Table 9: Steps in mixed-media tutorials (images used with permission)

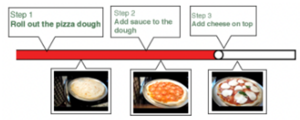
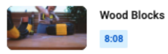
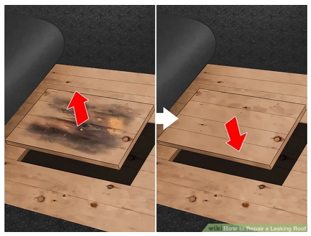
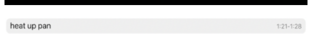
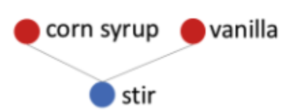
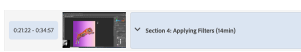
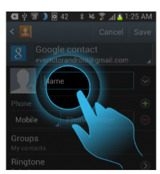
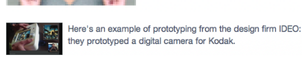
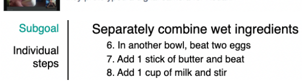
Topic	Source	Format	Human roles
General	ToolScape [36]		input for computational methods
	YouTube chapters [23]		create from scratch or NA
	WikiHow [7]		create from scratch
Cooking	videoWhiz [55]		refine computational results
	Yang et al [83]		create from scratch
	RecipeDeck [13]		refine computational results
Software	Fraser et al [24]		NA
	mixT [15]		NA
	EverTutor [77]		input for computational methods
Makeup	Truong et al [74]		NA
Lecture	Video Digests [59]		refine computational results
	Crowdy [79]		create from scratch

Table 10: Objects in mixed-media tutorials (images used with permission)

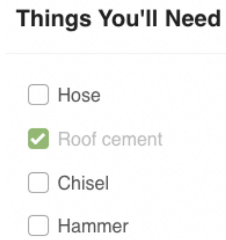
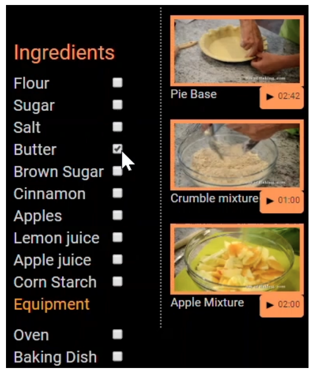
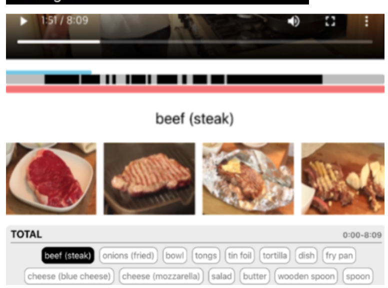
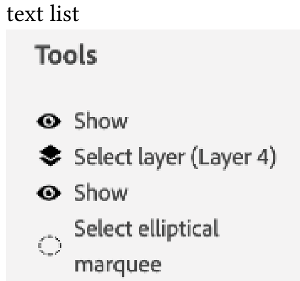
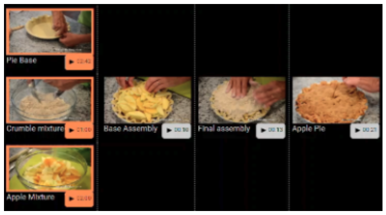
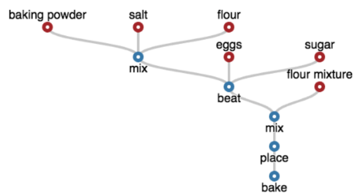
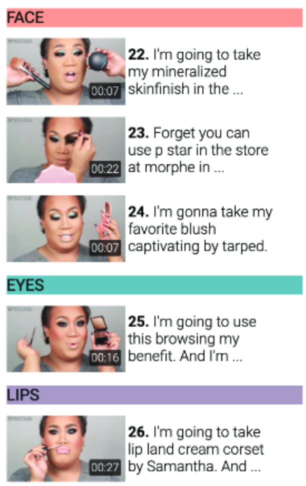
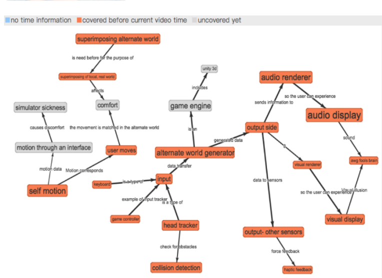
Topic	Source	Format	Human roles
General	WikiHow [80]		create from scratch
Cooking	videoWhiz [55]		refine computational results
	Yang et al. [83]		create from scratch
	RecipeDeck [13]	text list	refine computational results
Software	Fraser et al. [24]		NA
Lecture	ConceptScape [43]	text buttons	create from scratch

Table 11: Dependencies in mixed-media tutorials (images used with permission)

Topic	Source	Format	Relation	Human roles
Cooking	videoWhiz [55]		cooking order	create from scratch
	Yang et al. [83]		spatial relations/cooking order	create from scratch
	RecipeDeck [13]		cooking order	refine/input for computational methods
Makeup	Truong et al [74]		spatial relations	NA
Lecture	ConceptScape [43]		concept prerequisites	create from scratch