EQUI-VOCAL Demonstration: Synthesizing Video Queries from User Interactions

Enhao Zhang
University of Washington
enhaoz@cs.washington.edu

Maureen Daum
University of Washington
mdaum@cs.washington.edu

Dong He
University of Washington
donghe@cs.washington.edu

Manasi Ganti
University of Washington
mganti@uw.edu

Brandon Haynes
Microsoft Gray Systems Lab
brandon.haynes@microsoft.com

Ranjay Krishna
University of Washington
ranjay@cs.washington.edu

Magdalena Balazinska
University of Washington
magda@cs.washington.edu

ABSTRACT

We demonstrate EQUI-VOCAL, a system that synthesizes compositional queries over videos from user feedback. EQUI-VOCAL enables users to query a video database for complex events by providing a few positive and negative examples of what they are looking for and labeling a small number of additional system-selected examples. Using those user inputs, EQUI-VOCAL synthesizes declarative queries that can then retrieve additional instances of the desired events. The demonstration makes two contributions: it introduces EQUI-VOCAL's graphical user interface and enables conference attendees to experiment with EQUI-VOCAL on a variety of queries. Both enable users to gain a better understanding of EQUI-VOCAL's query synthesis approach and to explore the impact of hyperparameters and label noise on system performance.

PVLDB Reference Format:
Enhao Zhang, Maureen Daum, Dong He, Manasi Ganti, Brandon Haynes, Ranjay Krishna, and Magdalena Balazinska. EQUI-VOCAL Demonstration: Synthesizing Video Queries from User Interactions. PVLDB, 16(12): 3978-3981, 2023.
doi:10.14778/3611540.3611600

PVLDB Artifact Availability:
The source code, data, and/or other artifacts have been made available at https://github.com/uwdb/equi-vocal-demo.

1 INTRODUCTION

The increasing availability of inexpensive video storage coupled with advances in machine learning and computer vision has led to a surge in the use of video datasets in many applications. An important capability required by applications is the ability to find complex events, where multiple objects interact in space and time (e.g., a motorcycle passing too closely between pedestrians at an intersection). There are two main challenges in extracting complex events from videos. First, while general-purpose computer vision models are widely available, there is a lack of specialized models that identify user-defined complex events, especially for domain-specific applications. Training such a model, however, requires tedious and time-consuming data labeling. Those models further lack precise semantics and explainability for query results.

The second challenge lies in articulating the query declaratively. Assuming a set of off-the-shelf computer vision models, prior work supports users in expressing queries as compositions of primitive atoms (e.g., objects, relationships, attributes), either as sketches [2] or computer programs [5]. However, these approaches require users to have substantial familiarity with the database language to express such queries, and it is especially challenging when trying to capture real-world events, which can be difficult to articulate accurately.

In recent work [13], we developed EQUI-VOCAL, a system that addresses the above challenges by synthesizing compositional video queries from a small number of labeled video segments. EQUI-VOCAL outputs one or more synthesized queries that can serve to identify matching events in unseen videos. It introduces an expressive data modal and a query language based on spatio-temporal scene graphs [7], which conceptualize the contents of a video as a sequence of graphs and encompass rich information that includes objects, relationships, and attributes. EQUI-VOCAL employs a novel approach that synthesizes queries as a composition of extracted scene graph atoms in a way that limits user effort and computational overhead: EQUI-VOCAL synthesizes queries incrementally in a bottom-up fashion. It uses beam search to limit its exploration of the query space to a small set of most promising branches at each step and active learning [6] to iteratively request labels of carefully-curated video segments to reduce the uncertainty of synthesized queries. It finally employs a set of optimizations to avoid expensive database operations (e.g., recursive joins) that collectively allow it to scale to large video datasets and be resilient to noise.

In this demonstration, we introduce EQUI-VOCAL's graphical user interface (GUI) that facilitates the process of synthesizing compositional video queries. This interface provides the ability for a user to bootstrap query synthesis by specifying a set of positive and negative examples for an event. It also includes a labeling pane that enables users to seamlessly view and label system-selected video segments as query synthesis progresses, and observe the details of each synthesis step. Additionally, the interface shows the current best-scoring query and provides an interpretation of this query in Datalog and as a sequence of scene graphs. For a sample of demonstration queries, we further display the performance of the currently synthesized query on a held-out labeled test set.

Proceedings of the VLDB Endowment, Vol. 16, No. 12 ISSN 2150-8097.
doi:10.14778/3611540.3611600

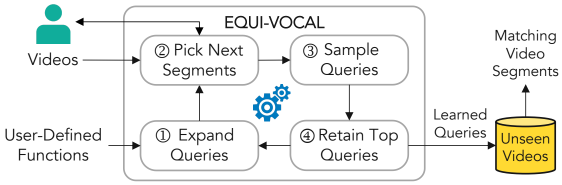


Figure 1: EQUI-VOCAL starts with an empty query and ① expands it by incrementally adding constraints. It ② picks new segments for the user to label that best differentiates the expanded queries. For efficiency, it ③ adopts a beam search strategy to sample a subset of queries to explore at each iteration. Lastly, it ④ updates top-*k* queries after each iteration.

We demonstrate EQUI-VOCAL using the Clevrer dataset [12], three guided query tasks, and one live query task. For the guided query tasks, the demonstration will maximize interactivity by precomputing results using the ground-truth labels. Attendees will then be guided through the synthesis process without needing to figure out what labels to provide and will observe the system's behavior to quickly understand how EQUI-VOCAL iteratively evolves queries. In the live query task, attendees will be able to adjust system configuration parameters, label videos themselves (including purposefully mislabeling some), and investigate EQUI-VOCAL's sensitivity to hyperparameters (e.g., labeling budget, beam width, number of initial examples) and label noise.

2 SYSTEM OVERVIEW

In this section, we briefly review EQUI-VOCAL's data model and various system components (previously described in [13]).

2.1 Data Model

EQUI-VOCAL uses spatio-temporal scene graphs as its underlying data model. Each video is a sequence of N frames $\{f_1, \cdots, f_N\}$. The visual content of each frame is represented by a *scene graph* $g_i = (\mathbf{o}_i, \mathbf{r}_i)$, which consists of the set of all *objects* $\mathbf{o}_i$ in a frame, along with a set of all *relationships* $\mathbf{r}_i$ between those objects. A *region graph* g_{ij} is a subgraph of g_i that contains the necessary information to identify an event, i.e., $g_{ij} \subseteq g_i$. Objects can additionally have *attributes*. Finally, an *event* e is a temporally ordered sequence of region graphs $e = \{g_1, \ldots, g_k\}$.

A query in EQUI-VOCAL returns video segment identifiers and is defined by $q(vid)$:- $g_1, \ldots, g_k, \mathbf{p}, \mathbf{d}, w$, where $g_1, \ldots, g_k$ is a temporally ordered sequence of region graphs specifying that a matching event consists of g_1, followed by g_2, followed by g_3, etc. Each g_i can persist for multiple frames and there can be other frames between g_i and g_{i+1}. $\mathbf{p}$ is a set of predicates that can be applied to objects, relationships, and attributes. $\mathbf{d}$ is a set of duration constraints applied to region graphs and defines the minimum number of contiguous frames that a region graph g_i should be valid before transitioning to the next region graph g_{i+1}. Finally, w is the maximum number of frames that can separate g_1 from g_k.

2.2 Synthesis Process

Figure 1 shows the overall architecture of EQUI-VOCAL. Given a set of videos and user-defined functions that extract semantic information from videos, EQUI-VOCAL starts with an empty query and iteratively explores the search space to refine it. We outline the components that enable EQUI-VOCAL to jointly synthesize high-performance queries and reduce computational and user effort.

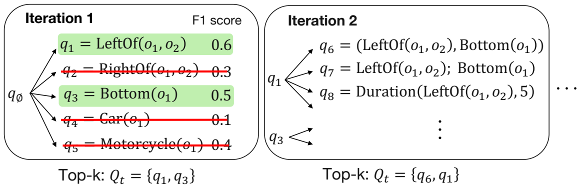

Iteration 1
F1 score
$q_\emptyset$
q_1 = LeftOf(o_1, o_2) 0.6
~~q_2 = RightOf(o_1, o_2) 0.3~~
q_3 = Bottom(o_1) 0.5
~~q_4 = Car(o_1) 0.1~~
~~q_5 = Motorcycle(o_1) 0.4~~
Top-k: $Q_t = \{q_1, q_3\}$
Iteration 2
q_1
q_6 = (LeftOf(o_1, o_2), Bottom(o_1))
q_7 = LeftOf(o_1, o_2); Bottom(o_1)
q_8 = Duration(LeftOf(o_1, o_2), 5)
q_3
Top-k: $Q_t = \{q_6, q_1\}$

Figure 2: EQUI-VOCAL expands queries by adding constraints to existing ones and samples bw most promising branches to explore at each step. At the end of each iteration, it updates the list of top-k queries seen so far. The example sets hyperparameters $bw = 2$ and $k = 2$.

2.2.1 Expanding queries. EQUI-VOCAL considers new queries by adding executable constraints and explores the search space based on the results of executing intermediate queries on its examples (Figure 2). EQUI-VOCAL applies three types of actions when expanding queries: (i) adding a new predicate to an existing region graph, (ii) inserting a new region graph, and (iii) increasing the duration of an existing region graph.

2.2.2 Sampling queries. Since exhaustive exploration is intractable, EQUI-VOCAL leverages a beam-search strategy that limits exploration at each step to only the most promising branches. In our approach, EQUI-VOCAL retains a subset of expanded queries, which will be further expanded in the next iteration, by evaluating their F1 scores on the currently-labeled dataset (Figure 2).

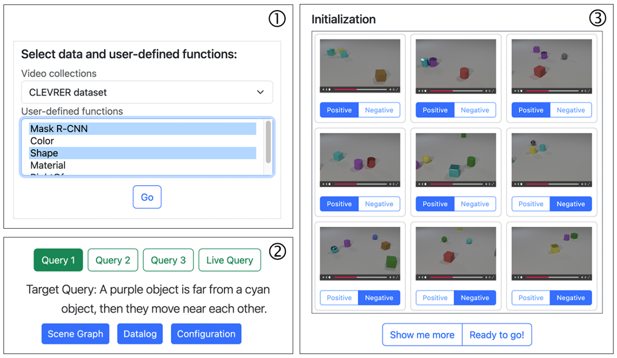
2.2.3 Selecting next segments. Rather than requiring a user to exhaustively provide all examples up front, EQUI-VOCAL starts with a small number of initial user-supplied examples. At each iteration, it adopts a disagreement-based active learning algorithm [6] to pick video segments that best differentiate between the candidate queries obtained during query expansion for a user to label next.

2.2.4 Retaining top queries. At the end of each iteration, EQUI-VOCAL updates its list of best-performing queries seen so far. In our prototype, we measure query performance by computing the F1 score on the currently-labeled dataset. Once terminated, EQUI-VOCAL returns a set of top-k synthesized queries, which may then be executed by a user to obtain matching events on unseen videos.

3 INTERACTIVE INTERFACE

This section describes the general use of EQUI-VOCAL's interface and its various components, as illustrated in Figures 3 to 5. We defer the discussion of the task selection pane to Section 4.

Data preparation. The *data preparation pane* (Figure 3, ①) enables users to select a video collection and set of user-defined functions (UDFs). If the video collection is a long video, EQUI-VOCAL will split it into short, non-overlapping video segments. The selected UDFs are used by EQUI-VOCAL to populate associated relational tables before performing query synthesis. For example, given the Shape UDF, it populates a table with the shape of all objects detected in each video in the collection. Our prototype applies the UDFs ahead of time for simplicity as our contribution focuses on query synthesis. Multiple methods exist to reduce this overhead [9, 11].



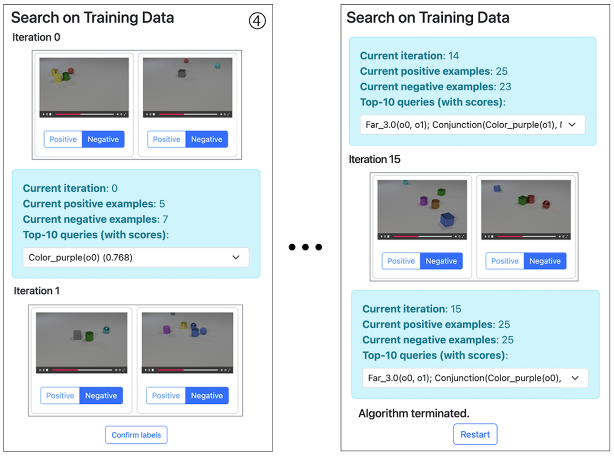
Select data and user-defined functions:

Video collections: CLEVRER dataset

User-defined functions: Mask R-CNN, Color, Shape, Material

Target Query: A purple object is far from a cyan object, then they move near each other.

Figure 3: ① Data preparation. ② Task selection (see Section 4 for details). ③ Query initialization.



Current iteration: 0
Current positive examples: 5
Current negative examples: 7
Top-10 queries (with scores):
Color_purple(o0) (0.768)

Current iteration: 14
Current positive examples: 25
Current negative examples: 23
Top-10 queries (with scores):
Far_3.0(o0, o1); Conjunction(Color_purple(o1), I

Current iteration: 15
Current positive examples: 25
Current negative examples: 25
Top-10 queries (with scores):
Far_3.0(o0, o1); Conjunction(Color_purple(o0),

Algorithm terminated.

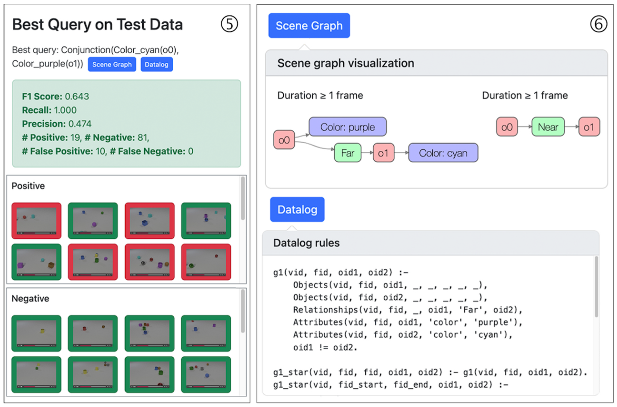
Figure 4: ④ Main labeling pane with history.

EQUI-VOCAL is preloaded with the following UDFs: an object detection model to identify common objects, functions that identify object attributes (`Color`, `Material`, and `Shape`), and rule-based functions based on bounding boxes that extract spatial relationships between objects (`Near`, `Behind`, etc.) as well as spatial locations for individual objects (`Left`, `Right`, `Top`, `Bottom`). Users can also create their own UDFs to be reused across video collections and tasks.

Initialization. The *query initialization pane* (Figure 3, ③) enables users to initialize EQUI-VOCAL with a small number of positive and negative examples. The user scans through video segments and marks each one as either positive (i.e., it contains the event they are searching for) or negative. EQUI-VOCAL automatically populates the pane with candidate videos. As we showed in [13], the number of examples needed for good query synthesis performance depends on the complexity of the target query. The prototype of EQUI-VOCAL populates the initialization pane using random sampling, though more sophisticated strategies are possible. We further discuss how this pane is populated for demonstration purposes to minimize the user effort in finding initial examples in Section 4.

Main labeling pane. Users interact with EQUI-VOCAL's iterative query synthesis algorithm via the *main labeling pane* (Figure 4, ④). EQUI-VOCAL starts with an empty query and gradually expands the complexity of candidate queries. Candidate queries are the partial queries synthesized at a given iteration. At each iteration, EQUI-VOCAL populates the labeling pane with videos selected using active learning (Section 2.2.3). The user then indicates whether or not each video matches their target query. EQUI-VOCAL uses these new labels to update the score of each of its candidate queries. The interface is then updated to show the current top-k queries and their scores on the labeled videos.

The labeling pane is scrollable and maintains results from previous iterations. Users can easily go back to better understand how the query evolved step-by-step. EQUI-VOCAL continues populating the labeling pane until the query synthesis algorithm terminates. Once terminated, EQUI-VOCAL returns the final top-k queries, and users can restart the process if they are not satisfied with the results.



Best query: Conjunction(Color_cyan(o0), Color_purple(o1))

F1 Score: 0.643
Recall: 1.000
Precision: 0.474
# Positive: 19, **# Negative:** 81,
# False Positive: 10, **# False Negative:** 0

```
g1(vid, fid, oid1, oid2) :-
    Objects(vid, fid, oid1, _, _, _, _, _),
    Objects(vid, fid, oid2, _, _, _, _, _),
    Relationships(vid, fid, _, oid1, 'Far', oid2),
    Attributes(vid, fid, oid1, 'color', 'purple'),
    Attributes(vid, fid, oid2, 'color', 'cyan'),
    oid1 != oid2.

g1_star(vid, fid, fid, oid1, oid2) :- g1(vid, fid, oid1, oid2).
g1_star(vid, fid_start, fid_end, oid1, oid2) :-
```

Figure 5: ⑤ Query prediction. ⑥ Alternative query interpretations (scene graph representation and Datalog rules).

Query prediction. While it is unlikely that users will have any labels for their videos, it is easy to reserve a set of videos from the collection for testing. The *query prediction pane* (Figure 5, ⑤) shows the performance of the current best query by displaying its predictions on the held-out test set. Test videos are grouped based on the query's predictions. The border color of the video will only be shown for the demonstration, as described in Section 4. Users can play the videos in this pane to manually verify how well the predictions of the current best query align with their target query.

Query representations. EQUI-VOCAL's interface also provides scene graph and Datalog interpretations (Figure 5, ⑥) of the current best query to enable users to verify whether the query components are reasonable given their event of interest. For example, if the user is searching for a general collision event between two objects but the current best query has a predicate specifying that the color of one object must be red, the user may be interested to see whether later iterations produce queries without this predicate.

4 DEMONSTRATION

We demonstrate EQUI-VOCAL on the Clevrer dataset [12] using four different query tasks. We provide three guided query tasks and one live query task. During the demonstration, we guide the attendee through the following steps (Figures 3 to 5).

Step ① The attendee selects a video collection and a set of UDFs to work with. For the demonstration, we ask the attendee to select the CLEVRER dataset, which comprises synthetic videos of moving objects, and all system-provided UDFs. We precompute the object detection results by executing the ML model on all video frames so that the attendee can instantly engage with the demonstration.

Step ② The attendee could choose from a list of guided query tasks or a live query task. The purpose of the guided query tasks is to demonstrate EQUI-VOCAL's ability to synthesize compositional video queries and facilitate the understanding of how EQUI-VOCAL iteratively evolves queries. We manually pick three target queries with different complexities. As an example, Query 1 looks for video segments where "a purple object is far from a cyan object, then they move near each other". EQUI-VOCAL automatically picks the hyperparameter configuration, precomputes the result using the ground-truth labels, and the attendee will observe the system's behavior. The live query task uses the same target query as Query 1 but allows the attendee to interact with EQUI-VOCAL by picking hyperparameters and labeling videos to investigate how sensitive EQUI-VOCAL is to hyperparameters and label noise. This includes deliberately mislabeling some video segments to observe system robustness, comparing the effect of initial example size on query synthesis results, setting the beam width to 1 to observe the performance degradation due to greedy search, etc.

Step ③ The attendee provides a few positive and negative examples to EQUI-VOCAL. For the guided query tasks, we prepare the initial set of video segments with labels for the attendee, and they can play the videos, observe the labels, and proceed to the next step. For the live query task, the attendee can label the video segments as positive or negative. To facilitate the demonstration and minimize the user effort in finding initial examples, we filter video segments based on their ground-truth labels and show the attendee an expectedly balanced set of videos, but the attendee can still decide to label any video segment as positive or negative.

Step ④ The attendee interacts with EQUI-VOCAL by labeling system-selected video segments and observing the query evolution. For the guided query tasks, the toggle button under each video segment is disabled and EQUI-VOCAL uses the ground-truth label as the user's label. For the live query task, the attendee can label the video segments freely.

Step ⑤ As the attendee labels video segments in step ④, EQUI-VOCAL updates the top-k queries after every iteration ($k = 10$ for guided queries; configurable for the live query). We show query prediction results on a test set containing 100 video segments. For the demonstration, we also have access to the ground-truth label of each video segment. The border color of each video segment indicates whether this prediction is correct. This visualization enables users to easily identify false positives and false negatives (i.e., video segments in the "negative" category but marked with a red border).

Step ⑥ Alternatively, the attendee can view the scene graph representation and the Datalog rules of the best query (step ⑤) and the target query (step ②) by clicking the "Scene Graph" and "Datalog" buttons associated with the query.

5 VOCAL

EQUI-VOCAL is part of the VOCAL [3] project, which envisions an end-to-end video analytics system to support interactive exploration, and compositional query processing over videos. An additional component of VOCAL is VOCALExplore [4], a system designed for interactive video data exploration and domain-specific model building. While EQUI-VOCAL focuses on compositional query synthesis using general-purpose, pretrained computer vision models as UDFs, VOCALExplore targets videos from domains without existing pretrained models. VOCALExplore provides an interface for users to label video segments and view predicted activities. It efficiently and automatically decides how to sample video segments and which feature extractor to apply in order to train high-quality models, all while ensuring fast response times. Models trained by VOCALExplore can be utilized by EQUI-VOCAL as UDFs to further help users find interesting complex events.

6 RELATED WORK

Prior work has investigated compositional queries on videos, but these systems either necessitate users to explicitly articulate the queries [2, 5] or develop specialized models for such queries [1]. In contrast, EQUI-VOCAL adopts a query-by-example approach and refines a query through iterative user feedback. Systems [8] that help users find events based on NLP descriptions can be adopted to find initial examples that EQUI-VOCAL needs as input. EQUI-VOCAL can then generate a precise and explainable query with clear semantics that targets the user intent.

Most query-by-example systems target tabular data. Quivr [10] is most relevant to our work, which also targets video queries, but it only operates on object trajectories rather than entire video scenes and assumes no label noise.

7 CONCLUSION

In this paper, we presented a demonstration of EQUI-VOCAL, a system that synthesizes compositional queries over videos through user feedback. The interface of EQUI-VOCAL enables users to iteratively label video segments and observe the evolution of synthesized queries. We also design important user interface components to help users visualize the query structure and query quality.

ACKNOWLEDGMENTS

This work was supported in part by the NSF through awards CCF-1703051 and IIS-2211133 as well as a grant from CISCO.

REFERENCES

[1] Daren Chao et al. 2020. SVQ++: Querying for Object Interactions in Video Streams. In *SIGMOD*. 2769–2772.

[2] Yueting Chen et al. 2022. Spatial and Temporal Constrained Ranked Retrieval over Videos. *PVLDB* 15, 11 (2022), 3226–3239.

[3] Maureen Daum et al. 2022. VOCAL: Video Organization and Interactive Compositional AnaLytics. In *CIDR*.

[4] Maureen Daum et al. 2023. VOCALExplore: Pay-as-You-Go Video Data Exploration and Model Building [Technical Report]. *arXiv preprint arXiv:2303.04068* (2023).

[5] Daniel Y. Fu et al. 2019. Rekall: Specifying Video Events using Compositions of Spatiotemporal Labels. *arXiv preprint arXiv:1910.02993* (2019).

[6] Mohammad Reza Karimi et al. 2021. Online Active Model Selection for Pre-trained Classifiers. In *AISTATS*, Vol. 130. 307–315.

[7] Ranjay Krishna et al. 2017. Visual Genome: Connecting Language and Vision Using Crowdsourced Dense Image Annotations. *IJCV* 123, 1 (2017), 32–73.

[8] Jie Lei et al. 2021. QVHighlights: Detecting Moments and Highlights in Videos via Natural Language Queries. In *NeurIPS*.

[9] Yao Lu et al. 2018. Accelerating Machine Learning Inference with Probabilistic Predicates. In *SIGMOD*. 1493–1508.

[10] Stephen Mell et al. 2021. Synthesizing Video Trajectory Queries. In *AIPLANS Workshop*.

[11] Oscar R. Moll et al. 2022. ExSample: Efficient Searches on Video Repositories through Adaptive Sampling. In *ICDE*. 3065–3077.

[12] Kexin Yi et al. 2020. CLEVRER: Collision Events for Video Representation and Reasoning. In *ICLR*.

[13] Enhao Zhang et al. 2023. EQUI-VOCAL: Synthesizing Queries for Compositional Video Events from Limited User Interactions. *PVLDB* 16, 12 (2023).