ACGD: Visual Multitask Policy Learning with Asymmetric Critic Guided Distillation

Krishnan Srinivasan[1], Jie Xu[2], Henry Ang[1], Eric Heiden[2], Dieter Fox[2], Jeannette Bohg[1], Animesh Garg[2,3]

***Abstract*— We present Asymmetric Critic Guided Distillation, ACGD, a framework for learning multi-task dexterous manipulation policies that can manipulate articulated objects using images as input. ACGD is a scalable student-teacher distillation approach that utilizes behavior cloning to distill multiple expert policies into a single vision-based, multi-task student policy for dexterous manipulation. The expert policies are trained with traditional RL techniques with access to privileged state information of both the robot and the manipulated object, while the distilled student policy operates under realistic sensory constraints, specifically using only camera images and robot proprioception. During distillation, we use an expert-critic that provides action labels and value estimates to refine the student's action sampling through a dual IL/RL objective. In the multi-task setting, we achieve this through an aggregate critic for different single-task experts. Our approach exhibits strong performance compared to a number of state-of-the-art imitation learning (IL) and reinforcement learning (RL) baselines. We evaluate across a variety of multi-task dexterous manipulation benchmarks including bimanual manipulation, single-hand object articulation tasks, and a tendon-actuated hand and achieves state-of-the-art performance with 10-15% improvement over the baseline algorithms. Visit our website for more details.**

I. Introduction

The dexterous use of tools remains a key challenge in robotics. It requires a multi-fingered robotic hand to make contact with a tool to use it in a deliberate fashion. One issue is that the high number of degrees of freedom of a dexterous multi-fingered hand increases the search space for a control policy. Furthermore, the additional degrees of freedom from the most commonly used tools (e.g. spray bottles or soap dispensers) require the control policy to very deliberately exert forces on the object. Prior work in this space either trained multi-task policies lacking the ability to manipulate tools and articulated objects [1–3], or required to manually collect demonstrations which can be difficult to scale to multiple objects [4, 5]. In this work, we study the problem of learning a multi-task, visuomotor manipulation policy that enables the multi-fingered robot hand to manipulate and control a diverse set of objects, without requiring manually-collected demonstrations.

Our approach to this problem uses a teacher-student distillation approach that can distill a set of single-task manipulation experts that use privileged information into a single, multi-task visuomotor policy network that takes camera observations and robot proprioception as input. The key technical contribution is a novel, critic-informed teacher-student learning framework that is capable of efficiently distilling skills from multiple single-task expert policies. Specifically, our critic-informed distillation leverages additional information from the RL-trained expert critics to form a dual-IL/RL objective [6] to distill multiple tasks at once. Our experiments showcase a set of simulated benchmarks testing multi-task, dexterous manipulation skills learned across a variety of different domains.

Contributions. The key technical contributions of this work are:

1) We present a novel teacher-student learning framework alongside a vector-quantized imitation learning policy architecture that can distill multiple state-based experts for dexterous manipulation into a single multi-task visuo-motor policy.
2) We compile a multi-task learning benchmark for learning dexterous visuo-motor policies from rendered images, consolidating over 15 tasks in 3 environments involving rigid and articulated objects with one or two multi-fingered hands.
3) We share diverse image and proprioception datasets of successful demonstrations generated with task-specific expert policies trained with privileged information.

We compare our approach to several baselines across tasks and benchmark datasets, implemented with the Robomimic [7] library, and observed consistent performance improvements across almost all tasks in three different domains: single-hand Allegro [8], dual-hand Shadow Hand [9], and single musculoskeletal arm environments [1], pictured in Fig. 1. We highlight the advantages of our method in overall task success rate performance, data efficiency, and multi-task scalability compared to several multi-task policy-learning methods such as ACT, BC-RNN, and MT-PPO [9], and DAgger [10] as an alternative distillation approach.

II. Related Work

Previous methods can generally be grouped by the following attributes: i) learn in a multi-task domain, ii) learn pixels to actions from demonstrations, iii) learn dexterous skills with reinforcement learning. Notably, few prior works exist at the intersection of these three domains: learning dexterous skills from pixels in a multi-task setting. Prior work that focuses on multi-task policies typically use grasping and re-orientation tasks such as manipulating blocks or hammers, and have shown to scale the number of objects that can be reposed by a single policy [1, 11]. These tasks demonstrate the ability of a single policy to pick up, translate, and rotate the object into a desired pose, albeit using ground-truth state inputs. Additionally, some of these methods rely on additional human-

[1]Stanford University, [2]NVIDIA, [3]Georgia Institute of Technology
Email: krshna@stanford.edu

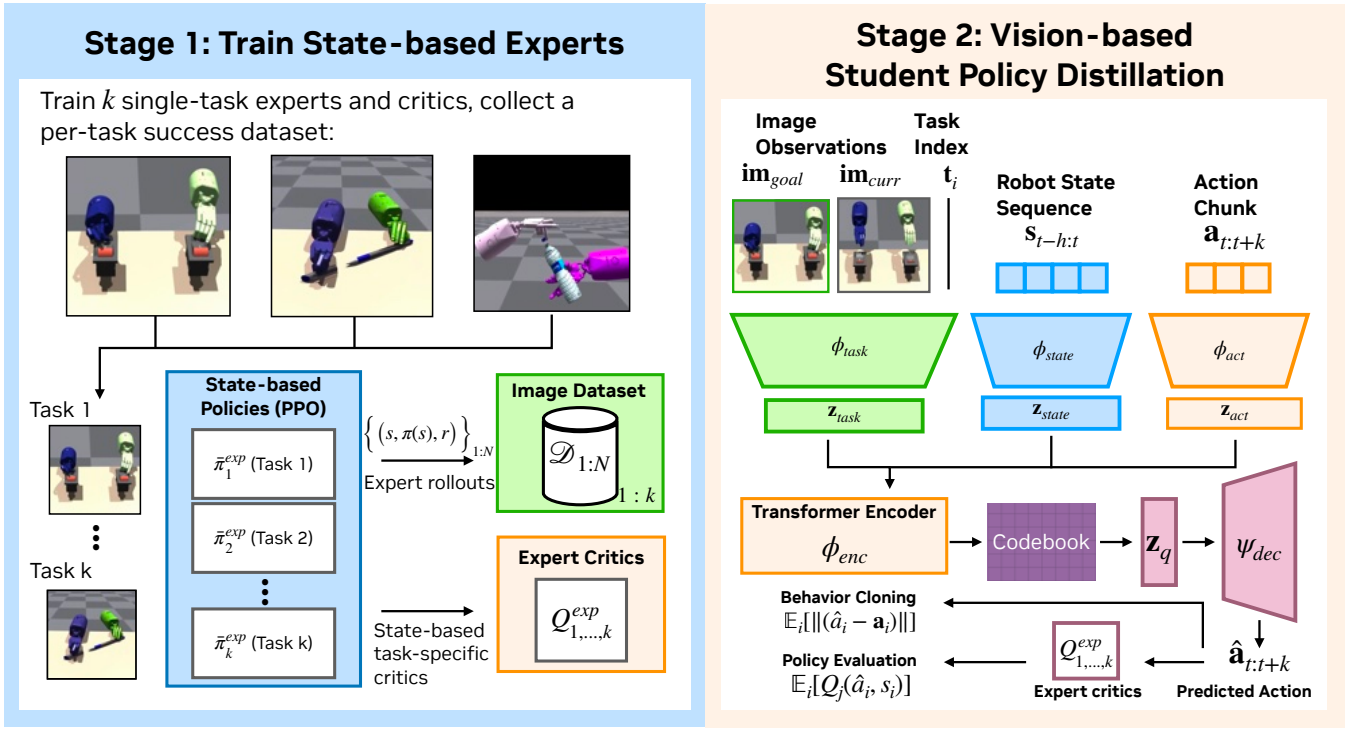

Fig. 1: Overview of ACGD broken into 1) expert training and data collection and 2) student-distillation phases. DistillACT leverages expert rollouts and critics collected in the first phase, and uses expert supervision through expert-critic mechanism for action labeling and value assessments to refine the student's actions during distillation. This is in contrast to DAgger, which also requires sampling of environment rollouts from the state-based expert policy during the second stage of policy distillation.

collected motion capture trajectories as demonstration data for learning. In contrast, our method interacts with objects that are both rigid and articulated, going beyond the reorientation of a single object, and learns from other learned experts using vision.

Another set of approaches can be categorized as motion generation techniques, which attempt to mirror human-generated trajectories of object interaction. While these approaches attempt to solve similar single and dual-hand object manipulation tasks [4, 12, 13], they do not generalize to multiple objects, and require hand-collected demonstration data from a human expert. In contrast, our work aims to automate the trajectory generation by learning single-task teachers with ground-truth state inputs. This avoids the need for human demonstrations and relies on single-task reinforcement learning experts instead to generate demonstrations, similar to methods such as AC-Teach [14] and R-MPO [15]. We also use the expert critics to do distillation instead of online distillation approaches such as DAgger [10, 16].

Some prior works have shown the ability to learn complex high-dimensional continuous control policies from vision, like playing soccer [17], or performing reposing of objects in-hand in mid-air [2, 18], using RL-only methods [3, 19]. However, these methods typically employ extensive multi-stage training pipelines, such as learning 3D rendering of the environment in a NeRF to perform sim-to-real transfer, or multi-stage student policy training to learn a 3D representation from point cloud information. While our work does not attempt to do sim-to-real, we rely on a two-stage training pipeline shown in Fig 1 for distilling our visual student policy by sample-efficient imitation learning, and using pre-trained vision backbones to extract RGB input features. We consider other pure-RL approaches to multi-task dexterous manipulation in simulation experiments include TDMPC2 [20] and MT-PPO [9], which we compare to extensively in Section IV.

We compare ACGD to behavior cloning and distillation methods. ACGD can build on any generic Behavior Cloning Model. In this paper we use the ALOHA ACT architecture as the behavior cloning backbone [21–23] to use with distillation metric that gives it a dual IL and policy-evaluation objective [6]. Additionally, our work adopts a student-teacher distillation approach similar to some past work, but they primarily used RL for both student and teacher policies [24–26]. We discuss the similarities and differences to these methods more extensively in the following section.

III. ACGD: Asymmetric Critic Guided Distillation

We propose ACGD, a method for distilling a multi-task policy from several single-task expert policies. The result is a continuous control policy for dexterous manipulation using image inputs. In our framework, as shown in Figure 1, a set of single-task expert policies are first learned with reinforcement learning using privileged ground-truth state information. Afterwards, a multi-task student policy that takes as input raw images and robot proprioception, is distilled from expert policies. At the core of the distillation process is an efficient imitation learning method leveraging an RL critic-informed distillation loss function.

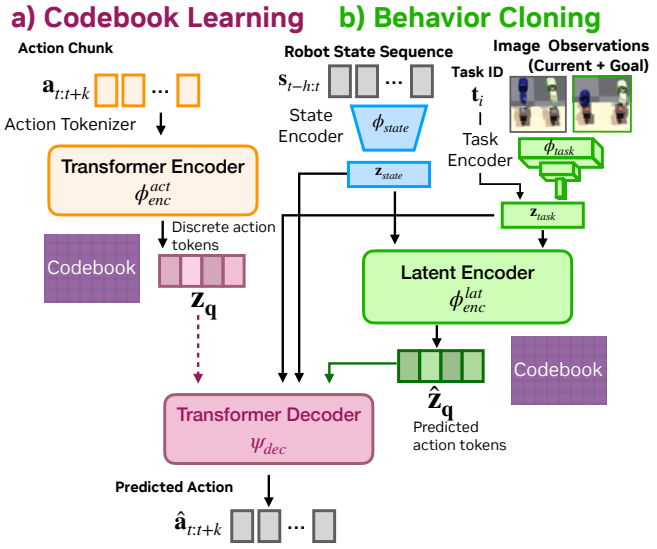

Fig. 2: Multitask Policy Distillation: Diagram showing the different modules contained within the policy VQ-VAE backbone. This includes two stages of training, a) action discretization, during which a codebook of discrete action tokens is learned, and b) behavior cloning, during which observations ($\mathbf{z}_{task}$) are encoded to predict the correct discretizeed and decoded actions.

A. *Behavior Cloning as Sequential Token Prediction*

While critic-guided distillation is compatible in theory with any combination of policy and vision backbone architectures, we chose a simple and effective combination of different imitation learning techniques that have shown to scale effectively for the multi-task student distillation problem. Our goal is to enable learning of task-specific features for actions and observations that allow the policy to coherently output actions for different tasks without confusion or catastrophic forgetting. To this end, we use a vector-quantized variational autoencoder (VQ-VAE) [27] to tokenize action chunks into discrete latent embeddings, and decodes them with a multi-headed attention decoder. This is similar to methods like ACT and VQ-BET [21–23, 28]. Below, we briefly summarize action chunking, temporal ensembling, and discretized action tokenization. For more details, we refer to [29].

Action chunking. – Action chunking groups multiple actions into a batched sequence to reduce the effective horizon of the task. Every k steps, the agent receives an observation and generates the next k actions to execute sequentially.

Temporal ensembling. – To avoid jerky motion from the abrupt change in observation conditioning every k steps, in practice we query the policy at every timestep, causing action chunks to overlap. Multiple predicted actions for a single timestep t from past k chunks (stored as $a_t^{t-k:t}$) are exponentially averaged with weight $w_i = \exp(-\eta i)$, making the ensembled prediction $a_t = \sum_{i=1}^{k} w_i a_t^{t-i}$. Both techniques have been show to be crucial for producing precise and smooth motion, as explored further in [29].

Vector-quantized action generation. – For multi-task imitation learning, we use a vector-quantized variational autoencoder (VQ-VAE) with a transformer encoder and decoder. Instead of predicting actions from latent variables parameterized using a standard normal distribution, ACGD constructs an embedding space for each action chunk $\mathbf{a}_{t:t+k}$ by encoding it as a vector-quantized latent code $\mathbf{z}_q \in \mathbb{R}^{n_q}$. This is done by mapping a continuous action chunk to its latent code encoded by a multi-headed attention model $\mathbf{z}_e = \phi_{\text{enc}}^{\text{act}}(\mathbf{a}_{t:t+k})$. Then for a given codebook C of n_c codes $\mathbf{e}_i$ that are each n_q-dimensional embedding vectors, a latent vector $\mathbf{z}_e$ is quantized by being mapped to its nearest neighbor in the codebook, giving $\mathbf{e}_c$ This yields the vector-quantized latent variable $\mathbf{z}_q(\phi_{\text{enc}}(\mathbf{a}_{t:t+k})) = \mathbf{e}_c$.

In practice, the codebook C is first trained to accurately encode action chunks $a_{t:t+k} \in \mathcal{D}$ with the transformer encoder $\phi_{\text{enc}}^{\text{act}}(\mathbf{a}_{t:t+k}) = \mathbf{z}_e$. The generated $\mathbf{z}_e$ is input to a quantizer [28], which maps $\mathbf{z}_e$ to a one-hot vector to select a code within the codebook C, resulting in latent quantized code $\mathbf{z}_q$. While there are multiple implementations of quantizers, ours treats $\mathbf{z}_e$ as the logits of a softmax function to obtain a probability distribution $\sigma(\mathbf{z}_e)$, which samples c from the resulting multinomial distribution: $c \sim M(\sigma(\mathbf{z}_e))$.

After obtaining the code $\mathbf{z}_q$, the discrete codes input to ψ_{dec} along with additional context for the current state and task, $[\mathbf{z}_{\text{state}}, \mathbf{z}_{\text{task}}]$, to reconstruct action chunks $\hat{\mathbf{a}}_{t:t+k} = \psi_{\text{dec}}(\mathbf{z}_q), \mathbf{z}_{\text{state}, \mathbf{z}_{\text{task}}}$. To train the action encoder and decoder, the reconstruction loss of predicted actions is computed as $\mathcal{L}_{\text{act}} = \frac{1}{k}\sum_{i=t}^{t+k} \|\hat{a}_i - \bar{a}_i\|_1$. The full codebook training objective then combines the action reconstruction and codebook alignment losses to align the latent variables $\sigma(z_e)$ with the sampled codes $\mathbb{1}_c$. Since the quantized code sampling step is non-differentiable, the code alignment loss uses the straight-through-estimator to copy gradients from the $\mathbf{z}_q$ to $\mathbf{z}_e$: $\mathcal{L}_{code} = \|\mathbf{z}_e - \text{SG}[\mathbf{z}_q]\|_1 + \|\text{SG}[\mathbf{z}_e] - \mathbf{z}_q\|_1$, where SG is a stop gradient. This makes the final codebook objective: $\mathcal{L}_{\text{codebook}} = \mathcal{L}_{\text{code}} + \mathcal{L}_{\text{act}}$.

Prior work [28] has demonstrated this vector-quantization scheme as being effective at modeling high-resolution multi-modal continuous actions, as they may correspond to different modes and sub-skills found in the demonstration data. We extend this to additionally study it in a multi-task domain, where these modes are inherently reflected in the different tasks.

B. *Improving Imitation with Critic Guidance*

Given a task set containing m tasks, where the action and state spaces $\mathcal{A}$ and $\mathcal{S}$ are shared across tasks, we take m single-task RL experts $\bar{\pi}_i$ and critics Q_i and construct an aggregate critic $Q_{\text{agg}} = [Q_1, \ldots, Q_m]$. Let $\mathbf{t}_i = \mathbb{1}_i$ be the one-hot task index to retrieve the estimated return Q_i corresponding to the i-th task expert critic. Then, given states s, observations o, actions a, and task indices $\mathbf{t}_i$ from a multi-task dataset $\mathcal{D}$ of rollouts generated from each of the m single-task experts $\bar{\pi}_i$, we distill a multi-task visuomotor student policy π_θ using the following distillation loss: $\mathcal{L}_{\text{distill}} = \frac{1}{|D|}\sum_{s,o,\mathbf{t}_i \in \mathcal{D}} [-Q_{\text{agg}}(s, \pi_\theta(s, o, \mathbf{t}_i)) \cdot \mathbf{t}_i]$. The complete critic-informed imitation learning objective for learning a multi-task visuomotor policy becomes:

$$\mathcal{L} = \alpha \mathcal{L}_{BC} + \mathcal{L}_{\text{distill}} \tag{1}$$

combining an arbitrary behavior cloning objective $\mathcal{L}_{BC}$, weighted by α, with the aggregate critic guidance term to maximize the student policy π_θ's expected return for multi-task policy improvement. In prior work, this formulation has been referred to as the dual-IL/RL objective [6], but note that our formulation uses a frozen asymmetric critic for multiple-tasks.

Remark: Another version of this training objective substitutes critic distillation for the DAgger [10] objective. For a given student policy π_θ and expert policy $\bar{\pi}_i$, the DAgger objective is $\mathcal{L}_{\text{DAgger}} = -\mathbb{E}_{s\sim\rho_{\tilde{\pi}}}\|\pi_\theta(s) - \bar{\pi}(s)\|^2$, where $\rho_{\tilde{\pi}}$ is the state distribution induced by following a mixture policy $\tilde{\pi}_\beta$. The mixing term $\beta \in [0, 1]$, mixes both student and teacher policies to sample actions from π with probability β and actions from $\bar{\pi}$ with probability $(1-\beta)$. While this approach is effective, it is computationally more demanding and requires further online interaction with the environment.

C. *Visual Multitask Policy Learning*

Our complete policy architecture π_θ (where $\theta = \{\phi, \psi\}$) consists of a vector-quantized variational auto-encoder (VQ-VAE) with two main components: a BERT-style transformer encoder ϕ_{enc} (as described in Sec. III-A), and an action decoder ψ_{dec}. The different inputs for training and inference necessitate two separate objectives for codebook learning and behavior cloning, as illustrated in Fig. 2 (a) and (b).

In order to achieve visual multi-task imitation learning, our policy must infer which action chunks to generate at test time based on the current robot state, observation, and task. To achieve this, we train the vision-based multi-task policy π_θ (illustrated in Fig. 2 (b)), which is composed of the latent encoder $\phi^{\text{lat}}_{\text{enc}}$ and decoder ψ_{dec}. An MLP state embedding network ϕ_{state} embeds the recent history of input robot states $\mathbf{s}_{t-h:t}$ to give $\mathbf{z}_{\text{state}}$. R3M [30] is used as the pretrained vision backbone ϕ_{task} to encode images and the task index $\mathbf{t}_i$, and receives the final frame o_T of each trajectory as the goal image, as well as the current image o_t from the environment. Together, these are used by the latent encoder $\phi^{\text{lat}}_{\text{enc}}$ to predict the latent code $\hat{\mathbf{z}}_q$. The predicted latent codes are then aligned to match the codes learned in the first codebook training stage by a cross-entropy loss $\mathcal{L}_{\text{lat}} = \text{CE}(\mathbf{z}_q(\phi^{\text{act}}_{\text{enc}}(a_{t:t+k})), \hat{z}_q(\phi^{\text{lat}}_{\text{enc}}(\mathbf{o}_t)))$, with $\mathbf{o}_t$ composed of the list of inputs $[o_t, o_T, \mathbf{t}_i]$ to the policy at inference time. Finally, the VQ-VAE decoder ψ_{dec} is reused to predict the action chunk $\hat{a}_{t:t+k}$, and is trained with the same action reconstruction loss as before. This gives the complete behavior cloning loss for the multi-task student policy as $\mathcal{L}_{\text{BC}} = \mathcal{L}_{\text{lat}} + \mathcal{L}_{\text{act}}$. Combining the codebook, reconstruction, and critic-guidance objectives, we derive the full critic-guided distillation objective from Eq. 1 for ACGD: $\mathcal{L}_{\text{ACGD}} = \alpha(\mathcal{L}_{\text{lat}} + \mathcal{L}_{\text{code}}) + \mathcal{L}_{\text{distill}}$.

IV. Experiments

When evaluating ACGD, we set out to address the following questions experimentally:

1) Does critic distillation improve learned multi-task skills compared to multi-task behavior cloning and reinforcement learning alone?
2) How does performance scale both with number of expert demonstrations per task and total number of tasks?
3) For visual dexterous manipulation tasks, what combination of policy architecture, pre-trained image encoder, and input features yields better policies?

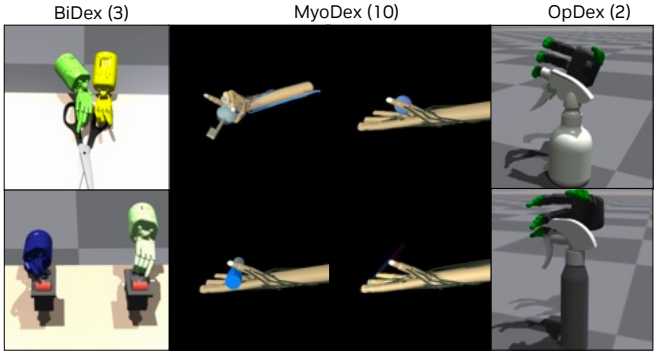

Fig. 3: A snapshot of different task sets and domains used in our benchmarks, showing dexterous manipulation of 22-52 degrees of freedom to manipulate a diverse range of object classes and visual domains.

To evaluate our approach, we consider a range of different difficult dexterous manipulation benchmark task sets (ranging from 2 to 10 skills per set), and compare with multiple policy architectures (BC-RNN, ACT, VQ-VAE, and MT-PPO), and observe the effect of how many demonstrations and tasks are needed for learning a robust multi-task policy. We present a comprehensive comparison of RL and IL baselines to our method to highlight the key improvements over prior work: namely the use of a) the critic distillation loss term and b) vector-quantization. We also evaluated DAgger as an alternate online distillation method to highlight key advantages of our offline critic-guided distillation approach in a reduced number of environment interactions.

Baselines – Comparing our method with existing work, we use BC-RNN [7] as a default behavior cloning strategy with a recurrent policy, and ACT [22] for an transformer-based BC architecture to highlight the effect of adding critic-guided distillation along with vector-quantization for multi-task behavior cloning. We additionally compare our method to multi-task algorithms, MT-PPO and MT-TDMPC2 [1, 20], which are model-free and model-based RL baselines respectively. Note that, we additionally include their single-task state-based variants as expert policies for generating rollouts and critics.

Tasks: Dexterous manipulation – To evaluate our method's ability to learn in a multi-task environment, we use three multi-task domains (Fig. 3): (a) the **MyoDex** benchmark from [1] (a single hand actuated by 39 DoF muscle tendons), b) the **BiDex** benchmark environments from [9] (52 DoF for two Shadow hands), and (c) **OpDex**: a modified AllegroHand environment from [31] with 22 degrees of freedom (16 DoF for hand and 6 DoF for wrist) and operable articulated objects. We include a complete list of tasks, videos, and training details on our website.

A. *Comparisons to BC and RL*

In the `AllegroHandSpray` task set, we compared our method to BC+DAgger, ACT, and MT-PPO on two different

Task	BC-RNN+DAgger [7]	ACT [21]	ACGD (Ours)	MT-PPO	Expert Policy
Observation Type	Image	Image	Image	Image	GT
Agent Type	Multi-task	Multi-task	Multi-task	Multi-task	Single-Task
KeyTurn	0.82	0.82	**1.00**	0.20	1.00
KeyTurnHard	0.08	0.02	**0.12**	0.00	0.20
ObjHold	0.71	0.81	**0.98**	0.00	1.00
ObjHoldHard	0.65	0.72	**0.81**	0.00	0.90
PenTwirl	0.86	0.84	**0.89**	0.12	1.00
PenTwirlHard	0.39	**0.58**	0.36	0.00	0.47
Pose	0.91	0.90	**0.93**	0.00	1.00
PoseHard	0.49	**0.96**	**0.99**	0.00	0.20
Reach	0.89	0.84	**0.99**	0.10	1.00
ReachHard	0.63	**0.76**	0.74	0.00	0.97
Myodex Mean	0.64	0.73	**0.78**	0.04	0.76
Scissors	0.73	0.33	0.84	0.57	**0.89**
Switch	0.89	0.93	**1.00**	0.45	0.65
Bottle	0.84	0.49	**0.93**	0.31	0.58
Bidex Mean	0.82	0.58	**0.92**	0.44	0.74
Spray 1	**1.00**	**1.00**	**1.00**	0.40	1.00
Spray 2	0.58	0.64	**0.89**	0.23	0.97
DexOp Mean	0.79	0.82	**0.95**	0.32	0.98

TABLE I: Comparing performance between BC-RNN+DAgger, ACT, Expert Policy, and ACGD across Myodex, Bidex, and DexOp spraying tasks. ACGD achieves state-of-the-art performance on image-based inputs, surpassing multi-task and single-task expert baselines in most tasks. We note that the ACT baseline performs relatively well on the MyoDex suite, but struggles in the bimanual dexterity task suite, even though there are fewer tasks in that set, indicating our method is well suited to learning higher-DoF multi-task visuo-motor skills.

spray bottle tasks. This task set is challenging for state-based experts because without vision it is difficult to know if and when the hand collides with the object, knocks it over and thereby terminates the episode. As shown in Table I in the OpDex task set, both distillation methods (BC+DAgger and ACGD) outperform MT-PPO, which fails at generalizing to both spray bottles with image-based observations. In contrast, our method (ACGD) significantly outperforms both, the DAgger distillation method and the BC and RL baselines. In addition to tasks with the Allegro Hand, we noticed a similar comparative advantage in the MyoDex and BiDex task sets (Table I). Our method achieves SOTA results and significantly outperform both RL and IL baselines in a majority of the tasks. However, when ablating our model with online DAgger distillation instead of critic-guided distillation (see Table II), we note that while the final task performance is close and comparable to ACGD, DAgger requires additional interactions with the environment, making it an "online" distillation method, compared to critic-guided distillation, which only requires the offline inference compute of Q_{agg}.

B. *Ablation of Model Design Choices*

Next, we perform an extensive empirical analysis of different architectural choices, to determine what key components are needed for high-performing multi-task dexterous visuomotor policies. Specifically, we look at a) the choice of encoder architecture, b) the choice of distillation method, and c) the vision backbone. Our full method outperforms the other combined approaches, but performs comparably to DAgger distillation when combined with the Vanilla and VQ-VAE encoders. However, we note that the key benefit of using critic-guided distillation is that it does not require additional interactions and labels with the environment. On real robot hardware, it would become infeasible query baseline actions from a state-based expert policy.

Additionally, we note that when comparing the vanilla and VQ-VAE architectures directly with no distillation, there are no clear performance differences in this specific task set. As can be seen in Table I, the gap widens in more complex tasks with more DoF (see Sec. IV-A).

C. *Effects of Scaling Demonstrations and Tasks*

A key attribute of multi-task policies is their ability to learn tasks from fewer demonstrations. To test this hypothesis, we varied the number of demonstrations and tasks and then analyse whether our approach catastrophically forgets specific tasks, or fails to generalize in our multi-task benchmarks. To measure the effects of increasing the number of tasks, task difficulty, and demonstration count, we evaluated on three overlapping sets of tasks from the MyoDex benchmark (`Myo5-Easy`, `Myo5-Hard`, and `Myo10`). We compare to TDMPC2 [20], a state-of-the-art, vision and state-based multi-task learning method. Overall, we found that our method outperforms TDMPC2 on both task sets by about 60% per task set (see Table 4).

In the MyoDex task sets, the easy version of a task uses a fixed goal pose, while hard tasks constitute random goals. While our model solves the easy tasks at a higher rate than the hard task set, the best performing model was trained on all 10 tasks when limited to 500 demonstrations per task. This highlights that *model performance is positively correlated with more tasks*, as the total number of trajectories trained on is increased while the number of demonstrations per task remains constant (Myo10 in Fig. 4). This is especially true if learning an easy task encodes relevant skills for the harder version of the task.

Method	Myo5-Easy (%)	Myo5-Hard (%)	Average (%)
ACGD (VQ-VAE), Critic Distillation	**0.96**	**0.60**	0.78
ACGD (VQ-VAE), No Distillation	0.86	0.62	0.74
ACGD (VQ-VAE), DAgger	0.94	0.56	0.75
Vanilla ACT, Critic Distillation	0.76	0.50	0.63
Vanilla ACT, No Distillation	0.88	0.58	0.73
Vanilla ACT, DAgger	0.84	0.60	0.73

TABLE II: Ablating various distillation methods and architectures (Vanilla VAE vs VQ-VAE) on Myo5-Easy, Myo5-Hard task sets. Across both distillation and architecture approaches, we find the VQ-VAE with the critic distillation method is empirically the best combination.

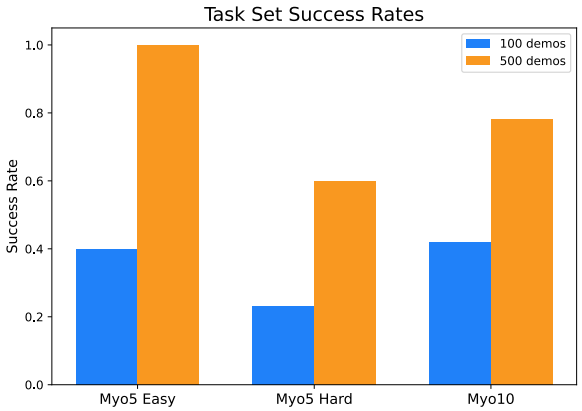

Fig. 4: Success rates as number of demonstrations and tasks scale (for 100 and 500 samples) on Myo5 Easy, Hard, and Myo10 tasks.

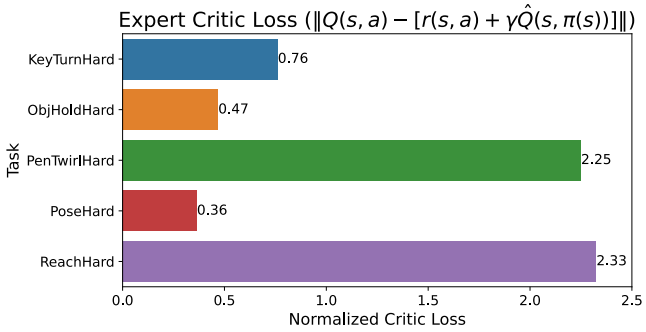

Fig. 5: Comparing the relative critic losses for tasks from the `Myo5-Hard` task set, we see that the critic losses for `PenTwirlHard` and `ReachHard` are key outliers, corresponding to tasks where ACGD struggles due to poor value estimation. In contrast, tasks with low critic prediction error show that ACGD's overall task performance can improve subject to better critic optimization and value estimation.

D. *Quality of Critic Estimation*

Next, we take a closer look at tasks where ACGD method underperforms the Vanilla ACT baseline shown in Table I. We plot the expert critic loss for 5 tasks in Fig. 5. If critic error is low, then ACGD outperforms vanilla behavior cloning, regardless of the quality of the expert. Notably, for the `KeyTurnHard` and `PoseHard` tasks, multi-task visuomotor policies outperformed the single-task experts using ground truth states as observations (data source). In contrast, in tasks where critic estimation is poor, regardless of the exert performance, ACGD does not outperform vanilla behavior cloning, since the additional learning signal from critic is not reliable. We see this in `ReachHard` (good expert, high critic loss) and `PenTwirlHard` (poor expert, high critic loss), where ACGD negatively affects the success rate on that task compared to no critic-distillation. This indicate multi-task generalization combined with vision as input and multi-task generalization can lead to further improvement over even single-task state-based RL.

V. Discussion and Conclusion

In this work, we take steps to extend the capabilities of single-task behavior cloning methods to solve multiple high dimensional continuous control tasks using vision. Our method, ACGD, uses critic-guided distillation to learn a multi-task student policy from a combined objective of aggregate Q-functions from multiple single-task experts with a behavior cloning objective using rollouts generated by those same experts. We use a Vector-Quantized Variational Auto-Encoder to predict action chunks from a learned discrete latent action token $\mathbf{z}_q$. Our proposed pipeline (Fig. 2), is evaluated on three different dexterous manipulation task sets: OpDex, BiDex, and MyoDex, first trains a set of single-task experts with RL, collecting a set of task demonstrations from each, and then finally distills the image-based policy with proprioception into a single student policy. In conclusion, this paper provides a strong basis for distilling multiple teacher policies for dexterous manipulation with vision, with a potential of scaling across tasks and demonstrations with more data and experts.

Limitations – Our experiments found that multi-task visuomotor policies trained with ACGD outperformed both IL and RL baselines, as well as DAgger distillation. However, we note that some of the tasks are limited by the expert critic policy evaluation, due to the learned single-task policy being a suboptimal expert. The resulting policy made our method perform worse than a naive behavior cloning model trained on success-only trajectories. Another architectural limitation is using goal-images from expert trajectories when conditioning the policy, as these are not available at test time. On real hardware, this would require each task be shown in a solved state once before evaluating the policy. Some recent works have shown that language embeddings [32, 33] to embed the task goal. This, or the use of a generative model for imagining goals at test-time [5] can remove this goal-image dependence. Lastly, the multi-task training dataset collected by our method is restricted to successes-only due to our chosen imitation learning approach. This remains a key limitation of many IL algorithms, since generated data is discarded. However, critic-distillation only relies on a well-trained critic that evaluates the student policy after being trained on negative samples. This could in theory make including negative rollout samples possible and even benefit the student policy.

References

[1] V. Caggiano, S. Dasari, and V. Kumar, "Myodex: A generalizable prior for dexterous manipulation," 2023.

[2] T. Chen, M. Tippur, S. Wu, V. Kumar, E. Adelson, and P. Agrawal, "Visual dexterity: In-hand reorientation of novel and complex object shapes," *Science Robotics*, vol. 8, no. 84, p. eadc9244, 2023. [Online]. Available: https://www.science.org/doi/abs/10.1126/scirobotics.adc9244

[3] T. Chen, J. Xu, and P. Agrawal, "A system for general in-hand object re-orientation," *Conference on Robot Learning*, 2021.

[4] H. Zhang, S. Christen, Z. Fan, L. Zheng, J. Hwangbo, J. Song, and O. Hilliges, "Artigrasp: Physically plausible synthesis of bi-manual dexterous grasping and articulation," 2024.

[5] Z. Zhou, P. Atreya, A. Lee, H. Walke, O. Mees, and S. Levine, "Autonomous improvement of instruction following skills via foundation models," *arXiv preprint arXiv:407.20635*, 2024.

[6] H. Sikchi, Q. Zheng, A. Zhang, and S. Niekum, "Dual rl: Unification and new methods for reinforcement and imitation learning," 2024.

[7] A. Mandlekar, D. Xu, J. Wong, S. Nasiriany, C. Wang, R. Kulkarni, L. Fei-Fei, S. Savarese, Y. Zhu, and R. Martín-Martín, "What matters in learning from offline human demonstrations for robot manipulation," in *arXiv preprint arXiv:2108.03298*, 2021.

[8] A. Handa, A. Allshire, V. Makoviychuk, A. Petrenko, R. Singh, J. Liu, D. Makoviichuk, K. V. Wyk, A. Zhurkevich, B. Sundaralingam, Y. Narang, J.-F. Lafleche, D. Fox, and G. State, "Dextreme: Transfer of agile in-hand manipulation from simulation to reality," 2024.

[9] Y. Chen, Y. Yang, T. Wu, S. Wang, X. Feng, J. Jiang, Z. Lu, S. M. McAleer, H. Dong, and S.-C. Zhu, "Towards human-level bimanual dexterous manipulation with reinforcement learning," in *Thirty-sixth Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2022. [Online]. Available: https://openreview.net/forum?id=D29JbExncTP

[10] S. Ross, G. J. Gordon, and J. A. Bagnell, "A reduction of imitation learning and structured prediction to no-regret online learning," 2011.

[11] S. Dasari, A. Gupta, and V. Kumar, "Learning dexterous manipulation from exemplar object trajectories and pre-grasps," in *2023 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2023, pp. 3889–3896.

[12] Z. Fan, M. Parelli, M. E. Kadoglou, M. Kocabas, X. Chen, M. J. Black, and O. Hilliges, "HOLD: Category-agnostic 3d reconstruction of interacting hands and objects from video," 2024.

[13] K. Zakka, P. Wu, L. Smith, N. Gileadi, T. Howell, X. B. Peng, S. Singh, Y. Tassa, P. Florence, A. Zeng, and P. Abbeel, "Robopianist: Dexterous piano playing with deep reinforcement learning," 2023. [Online]. Available: https://arxiv.org/abs/2304.04150

[14] A. Kurenkov, A. Mandlekar, R. Martin-Martin, S. Savarese, and A. Garg, "Ac-teach: A bayesian actor-critic method for policy learning with an ensemble of suboptimal teachers," *arXiv preprint arXiv:1909.04121*, 2019.

[15] A. X. Lee, C. Devin, J. T. Springenberg, Y. Zhou, T. Lampe, A. Abdolmaleki, and K. Bousmalis, "How to spend your robot time: Bridging kickstarting and offline reinforcement learning for vision-based robotic manipulation," *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 2468–2475, 2022. [Online]. Available: https://api.semanticscholar.org/CorpusID:248562602

[16] A. Galashov, J. S. Merel, and N. Heess, "Data augmentation for efficient learning from parametric experts," *Advances in Neural Information Processing Systems*, vol. 35, pp. 31 484–31 496, 2022.

[17] D. Tirumala, M. Wulfmeier, B. Moran, S. Huang, J. Humplik, G. Lever, T. Haarnoja, L. Hasenclever, A. Byravan, N. Batchelor *et al.*, "Learning robot soccer from egocentric vision with deep reinforcement learning," *arXiv preprint arXiv:2405.02425*, 2024.

[18] A. Agarwal, S. Uppal, K. Shaw, and D. Pathak, "Dexterous functional grasping," 2023.

[19] W. Huang, I. Mordatch, P. Abbeel, and D. Pathak, "Generalization in dexterous manipulation via geometry-aware multi-task learning," *arXiv preprint arXiv:2111.03062*, 2021.

[20] N. Hansen, H. Su, and X. Wang, "Td-mpc2: Scalable, robust world models for continuous control," 2024.

[21] T. Z. Zhao, V. Kumar, S. Levine, and C. Finn, "Learning fine-grained bimanual manipulation with low-cost hardware," *arXiv preprint arXiv:2304.13705*, 2023.

[22] Z. Fu, T. Z. Zhao, and C. Finn, "Mobile aloha: Learning bimanual mobile manipulation with low-cost whole-body teleoperation," in *arXiv*, 2024.

[23] T. Z. Zhao, J. Tompson, D. Driess, P. Florence, S. K. S. Ghasemipour, C. Finn, and A. Wahid, "Aloha unleashed: A simple recipe for robot dexterity," in *8th Annual Conference on Robot Learning*.

[24] Y. Xu, W. Wan, J. Zhang, H. Liu, Z. Shan, H. Shen, R. Wang, H. Geng, Y. Weng, J. Chen *et al.*, "Unidexgrasp: Universal robotic dexterous grasping via learning diverse proposal generation and goal-conditioned policy," *arXiv preprint arXiv:2303.00938*, 2023.

[25] Z. Jia, X. Li, Z. Ling, S. Liu, Y. Wu, and H. Su, "Improving policy optimization with generalist-specialist learning," 2022. [Online]. Available: https://arxiv.org/abs/2206.12984

[26] J. Lee, J. Hwangbo, L. Wellhausen, V. Koltun, and M. Hutter, "Learning quadrupedal locomotion over challenging terrain," *Science Robotics*, vol. 5, no. 47, Oct. 2020. [Online]. Available: http://dx.doi.org/10.1126/scirobotics.abc5986

[27] A. Van Den Oord, O. Vinyals *et al.*, "Neural discrete representation learning," *Advances in neural information processing systems*, vol. 30, 2017.

[28] S. Lee, Y. Wang, H. Etukuru, H. J. Kim, N. M. M. Shafiullah, and L. Pinto, "Behavior generation with latent actions," *arXiv preprint arXiv:2403.03181*, 2024.

[29] S. Haldar, Z. Peng, and L. Pinto, "Baku: An efficient transformer for multi-task policy learning," *arXiv preprint arXiv:2406.07539*, 2024.

[30] S. Nair, A. Rajeswaran, V. Kumar, C. Finn, and A. Gupta, "R3m: A universal visual representation for robot manipulation," 2022.

[31] V. Makoviychuk, L. Wawrzyniak, Y. Guo, M. Lu, K. Storey, M. Macklin, D. Hoeller, N. Rudin, A. Allshire, A. Handa, and G. State, "Isaac gym: High performance gpu-based physics simulation for robot learning," 2021.

[32] H. Bharadhwaj, J. Vakil, M. Sharma, A. Gupta, S. Tulsiani, and V. Kumar, "Roboagent: Generalization and efficiency in robot manipulation via semantic augmentations and action chunking," in *2024 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2024, pp. 4788–4795.

[33] H. Ha, P. Florence, and S. Song, "Scaling up and distilling down: Language-guided robot skill acquisition," 2023. [Online]. Available: https://arxiv.org/abs/2307.14535