Deep Learning for Real-Time Deformation Estimation in Robotic Material Handling Systems

Ciara Cantwell
Department of Computer Science, University of New Hampshire, Durham, NH, USA.
ciaracantwell74@unh.edu

Torgny Bergkvist
Department of Computer Science and Engineering, University of Nevada, Reno, Reno, NV, USA.
torgnyb@unr.edu

Catriona L. MacSweeney
Department of Electrical Engineering and Computer Science, University of Kansas, Lawrence, KS, USA.
catriona.l.macsweeney@ku.edu

Abstract

Robotic material handling increasingly involves limp, fibrous, granular, or viscoelastic workpieces whose geometry changes under grasp, acceleration, and transport. Real-time deformation estimation is therefore a perception and control problem rather than a purely geometric one: the robot must infer a dense displacement field, its strain consequences, and the reliability of that inference within the latency of a closed-loop controller. Existing model-based simulation and vision-only tracking methods face complementary limitations in accuracy, calibration burden, occlusion robustness, and computational cost. This paper develops a methodological framework for deep learning-based deformation estimation that fuses visual, tactile, force-torque, and kinematic observations with continuum-mechanical priors. The estimator maps a short history of multimodal observations to a displacement field and derived strain field, using spatiotemporal encoding to represent contact-induced and inertial deformation. Physics-constrained loss terms penalize kinematic incompatibility and implausible material response, while ensemble and heteroscedastic variance estimates expose model uncertainty. The paper specifies input-output definitions, architecture, training objectives, calibration logic, latency constraints, and control integration, and it analyzes occlusion, contact slip, material heterogeneity, simulation-to-reality shift, and overconfident extrapolation. Rather than presenting a single architecture as universally superior, the analysis derives conditions under which physical priors improve data efficiency and when they introduce model bias. The resulting framework provides a defensible design basis for deformation-aware perception and establishes evaluation criteria beyond average displacement error, including strain accuracy, uncertainty calibration, and worst-case latency.

Keywords

deformable object manipulation; deformation estimation; physics-informed deep learning; tactile sensing; real-time perception; uncertainty calibration

1. Introduction

Robotic material handling has traditionally assumed rigid workpieces whose pose can be estimated from sparse fiducials or geometric registration. That assumption breaks down for textiles, cables, food products, composite plies, foam, and granular-filled containers, where grasp forces and inertial loads produce nonlinear deformation. Surveys of deformable object manipulation show that sensing, modeling, and control are tightly coupled because the object state evolves during contact and cannot be reduced to a static six-degree-of-freedom pose [1]. Soft-robotic and compliant-hand designs further increase the relevance of deformation-aware perception, since the manipulator and the workpiece can both change shape under load [2]. The central difficulty is not merely visual tracking but estimating a physically meaningful deformation state quickly enough to inform grasp adaptation, trajectory correction, and force limiting.

Deep learning offers a route to this problem because deformation estimation requires learning mappings from high-dimensional observations to spatially dense outputs under conditions that are difficult to model analytically. Feedforward and convolutional networks can approximate complex input-output relations without hand-crafted feature pipelines [3]. Temporal recurrence allows the estimator to integrate evidence over time and to retain contact history, which matters when deformation is Viscoelastic or when a transient slip occurs [4]. Attention mechanisms can weight informative regions, sensor channels, or time steps and are attractive when occlusions make some observations unreliable [5]. Encoder-decoder architectures provide a natural structure for dense displacement or strain prediction because they preserve spatial layout while aggregating context [6]. Physics-informed learning adds another ingredient: governing relationships can be imposed through loss terms or architectural constraints, allowing a network to learn from limited data while respecting kinematic and constitutive structure [7].

The knowledge gap addressed here is the absence of an integrated design account that connects multimodal sensing, spatiotemporal deep learning, physical constraints, uncertainty calibration, and hard latency limits for material handling. Much work treats deformation estimation as an offline computer-vision task or as a simulation problem, while real-time control imposes different priorities. A latency violation can destabilize a grasp even when average displacement error is low; an uncalibrated uncertainty estimate can cause the controller to trust an occluded or out-of-distribution prediction; and a physics penalty can improve extrapolation while also biasing estimates when the assumed constitutive model is wrong. The contribution of this paper is a coherent methodological framework for real-time deformation estimation that specifies the estimator's variables, architecture, training objective, uncertainty treatment, calibration protocol, and control interface, and that analyzes the trade-offs that determine whether the framework is appropriate for a given handling regime. The paper is analytical and design-oriented; it does not report a completed experimental campaign or claim empirical performance. Its purpose is to make the design logic explicit and technically defensible before implementation choices are fixed.

2. Problem Formulation and Physical Constraints

Consider a workpiece occupying a time-varying domain $\Omega(t)$ in the robot's workspace. Let $u(x,t)$ denote the displacement of material point x at time t relative to a reference configuration, and let $\varepsilon(x,t)$ denote the infinitesimal strain tensor used for moderate deformations. The robot has access to a short observation history $S[t-k:t]$, which may include RGB-D frames, tactile images, wrench measurements, joint encoders, and commanded

gripper states. The control context c(t) includes grasp configuration, end-effector velocity, and payload class. A learned estimator with parameters θ produces a dense displacement estimate and, through differentiation, a strain estimate. The fundamental estimation relation is:

$$\hat{u}(x,t) = F\theta(S[t-k{:}\,t], c(t), m)$$

Here û is the predicted displacement field, Fθ is the parameterized estimator, S[t−k:t] is the observation window, c(t) is the control and grasp context, and m is a material or workpiece descriptor available from scheduling, prior inspection, or online adaptation. The relation makes explicit that deformation is not inferred from a single frame but from a temporally extended observation conditioned on the action being taken. This conditioning is essential: the same visual pattern can correspond to different strain states depending on whether the gripper is accelerating, holding, or releasing the object.

For control and safety assessment, displacement alone is insufficient. Grasp stability and material damage depend on local stretch, shear, and compression. Under the moderate-strain assumption, the strain field is obtained from the displacement gradient:

$$\varepsilon(x,t) = \tfrac{1}{2}(\nabla u(x,t) + (\nabla u(x,t))^{T})$$

where ∇u is the displacement gradient and the superscript T denotes transpose. The strain tensor ε captures normal and shear deformation in the reference frame. For large deformations, a finite-strain formulation would replace this linearized expression with a deformation gradient and an appropriate strain measure, but the linearized form is adequate for the methodological argument and for many handling scenarios in which local strain remains bounded. The important point is that the estimator must output a field that can be differentiated consistently; predicting isolated keypoints without spatial coherence is insufficient for strain-aware control.

The physical constraints of material handling introduce several boundary conditions that a purely data-driven estimator can violate. First, rigid-body motion should not be confused with deformation; a translating but undeformed workpiece should yield near-zero strain. Second, contact regions should exhibit boundary conditions consistent with the gripper geometry and applied wrench. Third, deformation should evolve smoothly in time except at slip, impact, or fracture events. Fourth, the estimated field should be compatible with the observed object silhouette and with tactile contact locations. These constraints do not determine the deformation uniquely, because material properties and internal structure are often unknown, but they restrict the plausible solution space and provide regularization signals for learning.

3. Methodological Framework

3.1 Multimodal sensing representation

The estimator should not rely on a single sensing modality. Vision provides global shape and surface texture but is vulnerable to occlusion and illumination change. Tactile sensing provides local contact geometry and normal or shear force information that vision cannot observe directly; high-resolution tactile sensors can estimate contact geometry and force distribution from surface deformation [8]. Force-torque sensing provides aggregate wrench information that constrains the equilibrium of the grasped object. Kinematic sensing supplies the robot's own motion, which is necessary to separate object deformation from camera or end-effector motion.

Data-driven performance modeling for nonuniform linear materials provides relevant precedent for learning heterogeneous material response in robotic fabrication [9]. In the present framework, material descriptors should be treated as conditioning variables rather than fixed labels. A workpiece may be homogeneous at the scale of the gripper but heterogeneous at the scale of the full object, and its effective response may depend on fiber orientation, moisture, temperature, or prior handling. The estimator therefore receives a descriptor vector m that can include class labels, estimated mass, compliance parameters, and contextual features from scheduling. When such descriptors are unavailable, the network can learn a latent material embedding from observed deformation history, but that embedding must be regularized to avoid absorbing transient sensor noise.

The observation model is intentionally simple at the design stage:

$$y_t = Hu_t + \eta_t$$

where y_t is the vector of observed features at time t, H is a modality-dependent observation operator that may be nonlinear in practice, u_t is the discretized displacement state, and η_t is observation noise including occlusion, tactile hysteresis, and calibration error. This relation clarifies that the network is solving an inverse problem with partial and noisy observations. It also motivates uncertainty estimation: when H poorly constrains a region because of occlusion, the posterior over u_t should be broad even if the network produces a visually plausible completion.

3.2 Spatiotemporal estimator architecture

A suitable architecture combines a spatial encoder, a temporal integrator, and a dense decoder. The spatial encoder processes each frame or sensor snapshot into a feature map. Convolutional layers are appropriate because deformation fields are spatially correlated and because local contact patterns should be detected regardless of absolute image position. Encoder-decoder skip connections, as used in U-Net-style architectures, help preserve fine spatial detail while aggregating context [6]. The temporal integrator then updates a latent state from the sequence of encoded observations and control context. Recurrent units such as long short-term memory can retain contact history over multiple time steps [4], while attention can selectively focus on time steps or spatial regions that are informative for the current grasp [5].

The latent dynamics can be written in a generic recurrent form:

$$h_t = \varphi(W_h h_{t-1} + W_s s_t + W_c c_t + b_h)$$

where h_t is the latent state, s_t is the encoded observation, c_t is the control context, W_h, W_s, and W_c are learned weight matrices, b_h is a bias, and φ is a nonlinear activation. This equation is not a claim about a specific trained model; it defines the temporal information path that any recurrent or attention-based variant must implement in some form. The latent state must summarize not only current geometry but also recent loading history, because viscoelastic and frictional effects make deformation path-dependent.

The decoder maps the latent state to a dense displacement field on a grid or mesh defined over the workpiece. Two design choices matter. A grid-based decoder is simple and compatible with convolutional training, but it must handle changing object topology and camera viewpoint. A mesh-based or point-based decoder can represent the object more naturally, but it requires reliable correspondence or a canonical template. In either case, the output should include a displacement field, a strain field derived by differentiable differentiation, and a per-region uncertainty estimate. The architecture should also expose

intermediate contact features so that the controller can distinguish a global shape change from a local grasp indentation.

3.3 Physics-constrained training and uncertainty

Training data for deformation estimation are expensive because dense ground-truth displacement is difficult to measure. Synthetic data from finite-element simulation can provide dense labels, but simulation-to-reality shift is substantial when friction, material heterogeneity, and sensor noise are imperfectly modeled. Domain randomization can reduce this shift by varying visual appearance, material parameters, lighting, and contact conditions during training [10]. Real-world data can then be used for fine-tuning with sparse labels from fiducials, tactile contact locations, or force-torque consistency.

A physics-constrained objective combines data fidelity with structural penalties. One generic form is:

$$L(\theta) = \Sigma_i \, ||\hat{u}_i - u_i||^2 + \lambda_p \, L_p + \lambda_u \, L_u$$

where the first term penalizes displacement error on labeled or pseudo-labeled samples, L_p penalizes violations of kinematic or constitutive constraints, L_u penalizes overconfident uncertainty, and λ_p and λ_u are weighting coefficients. The physics term may enforce compatibility of the strain field, equilibrium under known wrench measurements, or smoothness of the deformation gradient except near contact boundaries. The uncertainty term may use heteroscedastic variance, in which the network predicts a mean and variance and is trained with a Gaussian negative log-likelihood, or it may use ensemble disagreement. Bayesian deep-learning treatments distinguish aleatoric uncertainty, which arises from sensor noise and ambiguous observations, from epistemic uncertainty, which arises from model ignorance and distribution shift [11]. Dropout-based approximate Bayesian inference provides one practical way to obtain epistemic uncertainty estimates without a fully Bayesian training procedure [12].

For optimization, adaptive gradient methods such as Adam are common because they handle sparse gradients and heterogeneous loss scales [13]. The training schedule should not be treated as a minor implementation detail: physics penalties often dominate early training and suppress data fitting, while uncertainty terms can collapse to trivial solutions if the variance is unconstrained. A staged schedule that first learns displacement fidelity, then introduces strain and equilibrium penalties, and finally calibrates uncertainty is more defensible than a single fixed weighting. Calibration should be evaluated on held-out conditions using proper scoring rules for predictive distributions, not only mean displacement error.

The predictive variance can be estimated by an ensemble of estimators or by Monte Carlo sampling with dropout. A simple ensemble variance for a region is:

$$\sigma^2(x,t) = \left(\frac{1}{T}\right) \Sigma_j \left(\hat{u}_j(x,t) - \bar{u}(x,t)\right)^2$$

where T is the number of ensemble members or stochastic forward passes, û_j is the prediction from member j, and ū is the ensemble mean. This variance is not a substitute for a full posterior, but it provides a control-relevant signal: regions with high variance should be treated cautiously, and the controller should reduce speed, increase force sensing, or request a new viewpoint when variance exceeds a task-specific threshold.

4. Real-Time Inference and Control Integration

Real-time deformation estimation is a scheduling problem as much as a modeling problem. The controller has a sampling period determined by robot dynamics and safety requirements, and the perception pipeline must complete sensing, preprocessing, inference, postprocessing, and communication within that period. If the estimator runs slower than the controller, the robot acts on stale deformation estimates, which can cause instability during acceleration or grasp release. The design objective should therefore be expressed with an explicit latency constraint rather than as an unconstrained accuracy maximization:

$$min_\theta\ E[||\hat{u} - u||^2]\ s.t.\ E[\tau] \leq \tau_max$$

where τ is the end-to-end inference latency and τ_max is the maximum allowable latency for the control loop. This formulation makes the trade-off transparent: a more accurate but slower model is not admissible if it violates the control deadline. The expectation can be replaced by a high-percentile latency requirement for safety-critical handling, because worst-case latency matters more than average latency when the robot is moving a heavy or fragile object.

Architecture efficiency should be addressed at the design stage. Mobile inverted-bottleneck blocks can reduce computation while preserving representational capacity [14], and pruning, quantization, and weight compression can reduce memory bandwidth and inference time on embedded hardware [15]. However, compression can degrade the fidelity of fine spatial gradients, which are exactly what strain estimation requires. The estimator should therefore be evaluated after compression on strain error and uncertainty calibration, not only on displacement error. A compressed model that predicts smooth displacement but washes out local contact strain may be unacceptable for grasp control.

Control integration should use the estimator's uncertainty and strain outputs directly. A grasp controller might reduce commanded acceleration when predicted strain exceeds a material-specific limit, request additional tactile exploration when uncertainty is high near a contact patch, or switch from position control to impedance control when deformation is poorly observed. The estimator should publish deformation and uncertainty in a robot-centric frame with timestamps and validity flags. The controller should reject predictions whose latency exceeds τ_max or whose uncertainty exceeds a task threshold. This interface prevents the common failure mode in which a visually plausible but uncalibrated completion is treated as a physical measurement.

5. Technical Discussion: Failure Modes, Calibration, and Trade-offs

Several failure modes dominate deformation estimation in material handling. Occlusion is the most obvious: the gripper or workpiece can hide the region where strain is highest. Tactile sensing helps only near contact patches, and force-torque sensing provides an aggregate constraint rather than a spatial field. The estimator can learn to infer hidden deformation from visible boundary motion and contact wrench, but it should not be trusted without uncertainty inflation in occluded regions. A second failure mode is contact slip. Slip changes the boundary conditions abruptly and can produce a deformation transient that resembles material relaxation. Temporal models can detect slip from sudden changes in tactile shear and wrench, but they require training data that include slip events or a physics term that penalizes incompatible tangential motion.

Material heterogeneity is a deeper challenge. Continuum formulations distinguish reference and current configurations and define strain measures relative to a material model [16], while finite-element discretization provides a numerical route to solving boundary-value problems with spatially varying properties [17]. A learned estimator that assumes a single homogeneous material will misattribute local compliance to global motion or sensor noise. Conditioning on material descriptors and allowing spatially varying latent properties can mitigate this, but it increases the burden on training data and calibration. The physics penalty should therefore be treated as a soft prior, not a hard constraint, unless the constitutive model is known to be reliable for the workpiece class.

Simulation-to-reality shift affects both appearance and physics. Domain randomization can improve visual transfer [10], but it does not guarantee correct frictional or viscoelastic behavior. A practical strategy is to combine randomized simulation with a small amount of real interaction data and to monitor epistemic uncertainty on new workpiece classes. Bayesian filtering offers a complementary route: the learned estimator can provide a measurement model, while a state estimator propagates deformation forward using a simplified dynamic model and updates it with observations [18]. This hybrid structure is attractive because it separates learned perception from recursive estimation and makes uncertainty propagation explicit. The filter need not model the full continuum; a reduced-order state at keypoints or a coarse mesh may be sufficient for control.

Tactile hardware choices also shape the achievable accuracy. Vision-based tactile sensors can provide high-resolution contact geometry and force information [8], and compact low-cost designs have made dense tactile sensing more practical for robot hands [19]. The estimator should exploit the spatial structure of tactile images rather than treating them as unstructured vectors, because contact shape and shear direction are spatially organized. At the same time, tactile sensors have hysteresis, calibration drift, and limited coverage; the network must learn to discount unreliable tactile channels, and the controller must not assume that absence of contact signal means absence of contact.

Finally, end-to-end learning of visuomotor policies provides a useful contrast to the modular estimator proposed here. End-to-end training can learn task-relevant features directly from images and actions [20], but it obscures the deformation state and makes safety verification difficult. A modular deformation estimator exposes displacement, strain, and uncertainty, which can be checked against physical limits and logged for diagnosis. The modular approach may sacrifice some task-specific optimality, but it provides a more defensible basis for material handling, where the consequences of unobserved strain or slip can be material damage or loss of grasp.

The central trade-off is between expressiveness and constraint. A purely data-driven estimator can fit complex sensor patterns but may violate physical consistency and fail under distribution shift. A strongly physics-constrained estimator can extrapolate better but may be biased when the assumed constitutive law is wrong. The appropriate balance depends on the handling regime: for homogeneous, well-characterized materials, strong priors are advantageous; for heterogeneous, poorly characterized materials, weaker priors and stronger uncertainty monitoring are preferable. Real-time constraints add a third axis, because physics penalties and uncertainty ensembles increase computation. The framework therefore recommends a task-specific configuration rather than a universal model.

6. Conclusion

Real-time deformation estimation in robotic material handling requires more than a high-capacity network trained on displacement labels. It requires a representation that separates rigid motion from deformation, a temporal model that captures contact history, a physics-aware training objective that regularizes strain and equilibrium, and an uncertainty treatment that informs control decisions under occlusion and distribution shift. This paper has developed a methodological framework that connects these elements from problem formulation through architecture, loss design, calibration, latency constraints, and control integration. The framework defines displacement and strain fields as the estimation targets, conditions the estimator on multimodal observations and grasp context, and uses physics-constrained objectives with ensemble or heteroscedastic uncertainty. It also makes the real-time constraint explicit, so that accuracy is evaluated only within an admissible latency budget.

The main analytical insight is that physics priors are most valuable when they encode constraints that remain valid across the operating regime, such as kinematic compatibility and smooth temporal evolution, and most dangerous when they impose a constitutive model that the workpiece does not obey. Similarly, uncertainty estimates are useful only if they are calibrated and if the controller acts on them. The proposed design therefore treats deformation estimation as a safety-relevant perception component rather than an isolated vision benchmark. Future implementation should evaluate displacement error, strain error, uncertainty calibration, and worst-case latency on the same held-out conditions, including occlusion, slip, and material variation. That evaluation logic follows directly from the framework and provides a disciplined path from methodological design to deployment in robotic material handling systems.

References

1. Sanchez, J., Corrales, J. A., Bouzgarrou, B. C., & Mezouar, Y. (2018). Robotic manipulation and sensing of deformable objects in domestic and industrial applications: A survey. The International Journal of Robotics Research, 37(7), 688–716. https://doi.org/10.1177/0278364918779698

2. Rus, D., & Tolley, M. T. (2015). Design, fabrication and control of soft robots. Nature, 521(7553), 467–475. https://doi.org/10.1038/nature14543

3. Goodfellow, I., Bengio, Y., & Courville, A. (2016). Deep learning. MIT Press.

4. Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. Neural Computation, 9(8), 1735–1780. https://doi.org/10.1162/neco.1997.9.8.1735

5. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., & Polosukhin, I. (2017). Attention is all you need. In Advances in Neural Information Processing Systems (Vol. 30, pp. 5998–6008).

6. Ronneberger, O., Fischer, P., & Brox, T. (2015). U-Net: Convolutional networks for biomedical image segmentation. In Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015 (pp. 234–241). Springer. https://doi.org/10.1007/978-3-319-24574-4_28

7. Raissi, M., Perdikaris, P., & Karniadakis, G. E. (2019). Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving

nonlinear partial differential equations. Journal of Computational Physics, 378, 686–707. https://doi.org/10.1016/j.jcp.2018.10.045

8. Yuan, W., Dong, S., & Adelson, E. H. (2017). GelSight: High-resolution robot tactile sensors for estimating geometry and force. Sensors, 17(12), 2762. https://doi.org/10.3390/s17122762

9. Luo, D., Wang, J., & Xu, W. (2018). Robotic automatic generation of performance model for nonuniform linear material via deep learning. Proceedingof The 23rdConference on Computer-Aided ArchitecturalDesign Research in Asia.

10. Tobin, J., Fong, R., Ray, A., Schneider, J., Zaremba, W., & Abbeel, P. (2017). Domain randomization for transferring deep neural networks from simulation to the real world. In 2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) (pp. 23–30). https://doi.org/10.1109/IROS.2017.8202133

11. Kendall, A., & Gal, Y. (2017). What uncertainties do we need in Bayesian deep learning for computer vision? In Advances in Neural Information Processing Systems (Vol. 30, pp. 5574–5584).

12. Gal, Y., & Ghahramani, Z. (2016). Dropout as a Bayesian approximation: Representing model uncertainty in deep learning. In Proceedings of the 33rd International Conference on Machine Learning (pp. 1050–1059).

13. Kingma, D. P., & Ba, J. (2015). Adam: A method for stochastic optimization. In Proceedings of the 3rd International Conference on Learning Representations.

14. Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., & Chen, L.-C. (2018). MobileNetV2: Inverted residuals and linear bottlenecks. In 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (pp. 4510–4520). https://doi.org/10.1109/CVPR.2018.00474

15. Han, S., Mao, H., & Dally, W. J. (2016). Deep compression: Compressing deep neural networks with pruning, trained quantization and Huffman coding. In Proceedings of the 4th International Conference on Learning Representations.

16. Holzapfel, G. A. (2000). Nonlinear solid mechanics: A continuum approach for engineering. Wiley.

17. Bonet, J., & Wood, R. D. (2008). Nonlinear continuum mechanics for finite element analysis (2nd ed.). Cambridge University Press.

18. Särkkä, S. (2013). Bayesian filtering and smoothing. Cambridge University Press.

19. Lambeta, M., Chou, P.-W., Tian, S., Yang, B., Maloon, B., Most, V. R., Stroud, D., Santos, R., Byagowi, A., Kammerer, G., Jayaraman, D., & Calandra, R. (2020). DIGIT: A novel design for a low-cost compact high-resolution tactile sensor with application to in-hand manipulation. IEEE Robotics and Automation Letters, 5(3), 3838–3845. https://doi.org/10.1109/LRA.2020.2977257

20. Levine, S., Finn, C., Darrell, T., & Abbeel, P. (2016). End-to-end training of deep visuomotor policies. Journal of Machine Learning Research, 17(39), 1–40.